**CHAPTER ONE**

**1.1 INTRODUCTION**

There is an outgrowing need for electronic accessories in a world where there is great necessity for speed, efficiency and perfection of work. Information factors efficiency, access to electronic device and relevant information is usually considered in the setting up of a relational database for a local network intranet. The internet has revolutionized the computer and communications world like nothing before. The invention of the telegraph, telephone radio and computer, set the stage for unprecedented integration of capabilities. The intranet is at once a location-based broadcasting capability, a mechanism for information disseminating and a medium for collaboration and interaction between individuals and their computers without regard for extensive location.

Presently, the internet represents one of the most successful examples of the benefits of sustained investment and commitment to research and development of information infrastructure. Starting with the early research in pocket switching, the government, industry and academic have been partners in evolving and developing this existing new technology. Terms like “http:www.yahoo.com” and “Uktehamum Edu” trip off the lounge of the random person on the street.

The aim of this project is to give an introduction to database and its integration with HTTP server. Before the specifics and theoretical, lets focus on some of the more generic aspects of this study.

**1.2 BACKGROUND OF THE STUDY**

Information handling has gone beyond papers and is now at the discretion of the PC. Thus, this project is a conceivable effort to ameliorate the pressing need for a more web centric environment. Although its primary focus is not on web site design, yet it is inevitable. Since the emergence of the internet and its accompanying protocols, web use has intensified and so developing web applications is a promising way of reaching greater audience. Web use is not limited to the internet, but to virtually any application that can interface and communicate via web protocols. Intranet with the proposed design, staff, students and any authorized person can get all the available information from the polytechnic intranet integrated database. They can get their syllabus, academic calendars and results, register course, to mention just a few.

**1.3 STATEMENT OF THE PROBLEM**

Invariably, the problem that gave rise to this system study is from the continuous use of manual methods of dispensing academic details in the institution. The core is information management. As they say if you are not informed, you are deformed and he who is deformed cannot perform”. It would not be an exaggerated statement if I say that graduates from Nigerian institutions are deformed because they are not being well informed due to inadequate information acquisition facilities. It can be seen that information is not only important in this age but also the method of accessibility. No wonder it is termed as power, without it there is no headway. According to the U.S. Gen. Collins Powell (1995) “Lack of information breeds analysis paralysis” without adequate information on any subject matter, it would be difficult to handle. However, the enatic nature of power supply, especially in third world countries has tremendously affected online information systems and remote access to queries, information down time and data corruption is inevitable.

In Federal Polytechnic, Owerri the method of information processing and sharing is done manually and therefore slows down the speed at which information is stored, processed and shared. To eradicate this to its optimum a robust database and web server should be implemented where the information no longer have to reside in man’s brain or on his desk but on the network and can be accessed by anyone who is privileged.

**1.4 OBJECTIVE OF THE STUDY**

The new system will bring about efficient and effective academic information system. The concise documentation of students and staff details will tremendously improve its timely decision support.

The project would also go a long way in fostering good perception of the information age in our various institutions and individual organization thereby breaking the ugly bone of continuous use of manual and archaic systems. It fosters on the in depth principles, rules and protocols of coding, scripting and hosting of database driven web pages. It will also help in making the processing of information as well as on the sport feedback faster and provide spring board for our young database developer. It also stirs and gives focus to our institutions of higher learning on the facilities obtained in the use of modern system of communication.

In order to produce a more classified knowledge of the subject matter, the researcher intends to describe the tools required to design and implement a functional Apache HTTP web, HTTP web server and a MySQL relational database server for the institution through HTML encoded PHP web interface. The objective of developing the system and procedure for the polytechnic is to ensure that end users are provided with requisite tolls and data that are cost effective and easily accessible.

**1.5 SIGNIFICANCE OF THE STUDY**

These research works critically explores the possible ways of demystifying the mystery behind database and related technologies, web hosting services, protocols and of course design, and encourage young programmers to joint the race without feeling interns to their counterparts in the development parts of the world.

Without shifting this focus to web site design, we might still be in darkness for the next twenty years and by then a new inventions might be made which can sentence us unto another one thousand years of Dark Age. The study will go a long way to cutting down on time wastage, inefficient use of statistical data, manual errors and duplication of efforts by both staff and academic personnel. It will also enhance efficient query and enquiring ensure security of records and assist the polytechnic in copying with the daily work load.

**1.6 SCOPE OF THE STUDY**

Data not withstanding of its size cannot be too large to process. However, a guided approach is adopted to stay within limits for easy comprehension. To this end, the study will only focus on the students together with its accompanying web interface, students’ registration details and academic results.

In its most common form, the wide application of computers in accessing remote files paved way for adopting of more superior ways of making information available to the millions of users. And it also exposes the researcher to the core about the intricacies of database programming for web, an all round technology.

**1.7 LIMITATION OF THE STUDY**

The limitations encountered in this case are mainly based on the fact that the institution, Federal Polytechnic Nekede, Owerri does not have the necessary software and extended network of computers, that is, it lack the projected operational hardware. Nevertheless, the researcher simulated typical sceneries. Now, the last and not the least, the time to develop, test and implement the complete scenario was without unbearable sleepless night and frustration from public power supply.

**1.8** **DEFINITION OF TERMS**

**API** A term used to denote Application Programming Interface. An interface that is defined in terms of a set of functions and procedures, and enables a program to gain access to facilities within an application.

**BYTE** It is a fixed number of bits that can be treated as a unit by the computer hardware. It is a unit of storage capable of holding a single character equal to 8 bits, large amount of memory are referred in terms of kilobytes (1,024 bytes), megabytes (1,048,576 bytes), and gigabytes (1,073,741,824 bytes).

**BANDWIDTH** The amount of data or signal that can be transmitted in a transmission medium or media in a given time. It is measured in bits per second (bps) for digital devices and Hertz (HZ) for analog devices.

**COUNTER**  Keeps track of how many times a web page is visited.

**DOWNLOAD TIME**  The time it takes foe a web page to be completely visible on the computer screen.

**FILE SIZE**  How large the file containing the data is usually measured in kilobytes or megabytes. File size is used when referring to pages, HTML documents and any other type of file.

**FRAMES** Formatting a page so that more than one HTML document is visible on the screen at the same time.

**HOME PAGE**  The main and first page of a website, containing introductory information about the facility that has been accessed, together with links to the actual details of services or information.

**HTML** An abbreviation for Hypertext Mark-Up Language. A form of SGML (standard generalized mark up language) intended for use on the World Wide Web. It is used to create documents for the World Wide Web.

**HTTP**  Hypertext transfer protocol. An application level protocol with the lightness and speed, necessary for distributed collaborative hypermedia information systems. It transfers hypertext requests and information between servers and browsers.

**INTERNET** The global information network that now links a very substantial fraction of the world’s computer networks.

**INTRANET** A private network setup by an organization or company that resembles the World Wide Web, but which is inaccessible by external users.

**JAVA** A language for object-oriented programming on the internet, especially applicable to the World Wide Web.

**JAVA SCRIPT**  A scripting language designed to add features to web pages.

**OPERATING SYSTEM** The set of software products that jointly controls the system resources and the processes using these resources on a computer system.

**CHAPTER TWO**

**2.1 INTRODUCTION**

**A** relational database management system (RDBMS) is a collection of programs and capabilities that enable IT teams and others to create, update, administer and otherwise interact with a relational database. RDBMSes store data in the form of tables, with most commercial relational database management systems using Structured Query Language (SQL) to access the database. However, since SQL was invented after the initial development of the relational model, it is not necessary for RDBMS use.

The RDBMS is the most popular database system among organizations across the world. It provides a dependable method of storing and retrieving large amounts of data while offering a combination of system performance and ease of implementation.

**2.2 REVIEW OF RELATED LITERATURE**

In the primitive and barbarian days before computers the amount of information shepherd by a group of people could be collected in the wisdom and the stores of its older members. In this world, storytellers’ magicians and grandparents were considered a great and honored storehouse for all that was known.

Apparently, and according to vast archaeological data, complies were used (like command line middle ware) by the younger members of the community to access the information stored in the minds of the elders using APL’s such as “public string tellUsAbout. The time when (strings).” And then of course, like a sweeping and rapidly encompassing viral infection, came agriculture, over production of food stuff, and the origins of modern day commerce. Dealing with vast storehouses of wheat, rice and maize became quite a chore for the monarchs and emperors that developed along with the new economy. There was simply too much data to be managed in the minds of the elders. And so, in order to store all the new information, humanity invented the technology of writing. And though great scholars like Aristotle warned that the invention of the alphabet would lead to the sensibility of humanity; data began to be stored in voluminous data repositories called books. As we know, eventually books copulated with great speed and soon, whole communities of books migrated to the first real “databases” libraries. Unlike previous versions of data warehouses (people and books), that might be considered the australopithecine of the database lineage, libraries crossed over into the modern day species, though they were incredibly primitive of course over into libraries introduced standards by which data could be stored ad retrieved.

After all, without standards for accessing data, libraries would be like closet, endless and engulfing swarms of chaos. Books and the data within books had to be quickly accessible by anyone if they were to be useful. In fact, the usefulness of a library, or any base of data, is proportional to its data storage and retrieval efficiency. This one corollary would drive the evolution of database over the next 2000 years to its current state.

Thus, early libraries defined standardized filing and retrieval protocols. Perhaps, if you could visit the polytechnic library, you will see its cute little indexing system (card catalog) and pointers. And more complex and simple libraries grew and grew, along with associated storage/retrieval technologies such as the filing cabinet, colored tabs and three ring binders.

The day computer was born, almost instantly; the computer was applied to the old age problem of information storage and retrieval. After all, by information storage and retrieval. After all, by World War II, information was already accumulating at rates beyond the space available in publicly supported libraries.

Thus, the first attempts at information storage and retrieval followed traditional lines and metaphors. The first systems were based on discrete files in a virtual library. In this file oriented system, a bunch of files would be stored on a computer and could be accessed by a computer operator. Files of archived data were called “tables” because they looked like tables used in traditional file keeping. Rows in the table were called “records” and columns were called “fields”.

**Consider the following:**

|  |  |  |  |
| --- | --- | --- | --- |
| First name | Last name | E-mail | Phone number |
| Sharon | Stone | [sharons@eff.org](mailto:sharons@eff.org) | 08021346509 |
| Uche | Uzokwe | [uche@outgun.com](mailto:uche@outgun.com) | 08023912196 |
| Hedd | Ismaila | [heed@mun.ca](mailto:heed@mun.ca) | 0817779876 |
| Emmanuel | Mark | [nuelss@botton.ac.uk](mailto:nuelss@botton.ac.uk) | 07033561236 |

The “flat” system was a start. However, it was seriously inefficient. Essentially, in order to find a record, someone would have to read through the entire file and hope it was not the last record. With a hundred thousand records, you can imagine the dilemma. What was needed, computer scientists though (using existing metaphors again) was a card catalog, a means to achieve random access processing, that is, the ability to efficiently access a single record without searching the entire file to find it. The result was the indexed file oriented system in which a single index file, stored “key” words and pointers to records that were stored elsewhere. This made retrieval much more efficient. It worked just like a card catalog in a library. To find data, one needed only search for keys rather than reading entire records.

However, even with the benefit of indexing, the file oriented system still suffered from problems including:

|  |  |
| --- | --- |
| Data redundancy | The same data might be stored in different places |
| Poor data control | Redundant data might be slightly different such as when one changes he/her name to another, and the change is only reflected in some of the files containing her data. |
| Inability to easily manipulated data | It was a tedious and error prone activity to modify fillies by hand. |
| Cryptic work flows | Accessing the data could take excessive programming effort and was too difficult for real users (as opposed to programmers) |

Consider how troublesome the following data files would be, to maintain:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Address | Course | Grade |
| Mr. C.N. David | 123 Howell | Compiler | BC |
| Mr. C.N. David | 44 Brass Street. | Sociology 104 | A |
| Mr. C.N. David | 123 Howell | Data structures | B |
| Mr. C.N. David | 123 Howell | English 101 | A |
| Ms. Nkoli A | 100 capita Ln. | Psychology 102 | AB |
| Mr. Nkoli A | 88 West 1st Str. | Human cultures | A |
| Ms. Nkoli A | 88 West 1st Str. | European Govt. | A |
| Patricia Amadike | 89 Poly Rd. | Operations Research | B |

What was needed was a truly unique way to deal with the old age problem, a way that reflected the medium of the computer rather than the tools and metaphors it was replacing. The succeeding chapters would provide added clarifications.

As with the preceding paragraphs, simple put, a database is a computerized record keeping system. It is a system involving data, the users who turn the data into information the hardware physically stores that data, the software stores the data and provide a standardized method for retrieving or changing the data. Databases, since its conception in the 60’s were created to solve the problems with file oriented systems in that they were compact, fast, easy to use, current, accurate, allowed the easy sharing of data between multiple users and were secured.

A database might be as complex and demanding as an account tracking system used by a bank to manage the constantly changing accounts of thousands of bank customers, or it could be as simple as a collection of student registration number with matching departments. The important thing is that, a database allows you to store data and get it and modify it when you need to regardless of the amount of data being manipulated. Traditionally, databases run on large, powerful mainframes for business applications. However, with the advent of small, powerful personal computers, databases have become readily usable by the average computer users. MySQL Open source, Microsoft’s Access and Borland’s Dbase proprietary are a few popular PC-based engines around.

More importantly for our focus, database as we know them today were in their infancy. Around 1970 a researcher called (Ted Codd had developed the “relational data model” which was to become the foundation stone of modern database technology. In the mid 70s however, computer database were not a common thing. These days when you talk about databases in the wild, you are primarily talking about two types: Analytical databases and operational databases. Operational databases (a.k.a OLTP-On-Line Transactions Processing), on the other hand, are used to manage more dynamic bits of data. These types of database allow you to modify, add, change or delete data.

These types of databases are usually used to track real time information. For example, a financial company might have an operational database used to track cash transactions as customers make payments from an online store, an operational database can be used to keep track of how much cash is left and if the customer draws below his credit line.

Analytical database (a.k.a OLAP-Online Analytical Processing) is primarily static read only databases which store archived, historical data used for analysis. For example, a manufacturing company might store sales records over the last ten years in an analytic database and use that database to analysis marketing strategies in relation to demographics. On the web, you will often see analytic database in the form of inventory catalogs such as those of Amazon and eBay.

An inventory catalog analysis databases usually holds descriptive information about all available products in the inventory. Web pages are generated dynamically by querying the list of available products in the inventory against some search parameters. The dynamically generated page will display the information about each item (such as title, author and so on) that is stored in the database.

**What is a data model?**

It is an abstract model of some real world situation or domain of interest about which information is to be held in a database and which the logical scheme for that database encodes. Data models are abstractions, oftentimes in mathematical algorithms and concepts. You cannot really touch a data model. But nevertheless, they are very useful. The analysis and design of data models has been the cornerstone of the evolution of database efficiency. Before the 1980’s, the two most commonly used database models were the hierarchical and network systems.

As its name implies, the Hierarchical database model defines hierarchically arranged data. Perhaps, the most intuitive way to visualize this type of relationship is by visualizing an upside down tree of data. In this tree, a single table acts as the “root” of the database from which other tables “branches” out. You will be instantly familiar with this relationship because that is how all window based directory management systems like windows explorer works. Relationships in such a system are thought of in terms of children and parents such that a child may only have one parents but a parent can have multiple children.

Parents and children are tied together by links called “pointers” (perhaps physical addresses inside the file system). A parent will have a list of pointers to each of their children.

|  |  |  |  |
| --- | --- | --- | --- |
| Course ID | Course Name | Department | Professor |

Phone

Address

Name

Assign

Student Grade

ID Name Address Phone

Hierarchical model

This child/parent rule assures that data is systematically accessible. To get to a low level table, you start at the root and work your way down through the tree unity you reach your target. The hierarchical model however, is much more efficient than the flat file model we discussed earlier because, there is no much need for redundant data. If a change in the data is necessary, the change might only need to be processed once.

Consider the student flat file database example from our discussion of what database are:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Address | Course | Grade |
| Mr. Chike David | 123 Howell | Compiler | BC |
| Mr. Chike David | 44 Nekede Rd. | Sociology | A |
| Mr. Chike David | 123 Howell | Data structure | B |
| Ms. Nkoli A | 88 West 1st Str. | Psychology 101 | A |
| Ms. Nkoli A | 100 capitol Ln. | Psychology 102 | AB |

As we mentioned before, this flat file database would store an excessive amount of redundant data. If we implement this in a hierarchical database model, we would get much less redundant data.

**2.3 DATA AND INFORMATION**

Data and information are terms people use interchangeably in everyday speech, but they mean different things. Data are facts, such as name, a number and so on while information is organized processed data. A data item (e.g. the data 7/15/99) means little. When you associate the data items, such as a deadline and a subject, you concrete information. For example, the deadline for your next project might be 7/15/99, you store data in a database, and you retrieve information from the database. One cornerstone of data design and data normalization is that data organization for storage differs from the information most people want to see. For example, a manager of a sporting goods company might want to see who the customer is, the destination of the order, the billing address, the contact phone number, the placement time of the order the orders shipping destinations, when and how delivery occurred, what articles the order included differs from the particular information the manager wants.

However, as you can imagine, the hierarchical database model has some serious problem. For one, you cannot add a record to a child table until it has already been incorporate into the parent table. This might be troublesome if, for example, you wanted to add a student who had not yet signed up for any course. Worse, yet, the hierarchical database model still create repetition of data within the database. You might imagine in the database system shown above, there may be a higher level that includes multiple courses. In this case, there could be redundancy because students would be enrolled in several courses and thus each “course free” would have redundant student information. Redundancy would occur because hierarchical database handles one to many relationships well but do not handle many relationships well. This is because a child may only have one parent. However in many cases you will want to have the child is related to more than one parent. For instance, the relationship between students and class is a “many to many” not only can a student take many subjects and many students may also take a subject. How would you model this relationship simple and efficiently using a hierarchical database? The answer is that you wouldn’t.

Though this problem can be solved with multiple databases creating logical links between children, the fix is very awkward. Faced with this serious problem, the network model was concealed. The network database model was designed to solve some of the more serious problems with the hierarchical database model. Especially the network model solves the problem of data redundancy by representing relationships in terms of sets rather than hierarchy. The model had its origins in the conference on data systems languages (CODASYL), which had its origins in the database task group to explore and design a method to replace the hierarchical model. The network model is very similar to the hierarchical model. The hierarchical model is a subset of the network model. However, instead of using a single parent tree hierarchy, the network model uses set theory to provide a free like hierarch with the exception that child tables were allowed to have more than one parent. This allowed the network model to support many to many relationships.

Visually, a network database looks like a hierarchical database in that you can see it as a type of tree. However, the look is several trees, which share branches. Thus, children can have multiple parents, parents can have multiple children.

647 105366

801 10533

556 100,000

900 55

Hodges Sidehill Brooklyn

Shiver North Bronx

Lowery Maple Queens

Nevertheless, though it was a dramatic improvement, the network model was far from perfect. Most profoundly, the model was difficult to implement and maintain. Most implementations of the work model were used by computer programmers rather than real users. What was needed was a simple model that could be used by real end uses to solve real problems.

Of course in the 80’s the “relational database model” became the rage. The relational model developed out of the work done by Dr. E.F. Codd at IBM in the late 1960s to find ways to solve the problems with the existing models.

At the core of the relationship model is the concept of a table (also called a relation) in which all data is store. Each table is made up of record (horizontal rows also known as tuples) and fields (vertical columns also known as attributes). It is important to note that how or where the tables of data are stored make no difference. Each table can be identified by a unique name and that name can be used by the database to find the table behind the scenes. As a user, all you need to know is the tables name in order to use it. You do not need to worry about the compelxitis of how the data is stored on the hard drive. This is quite a bit different from the hierarchical and network models in which the user had to have an understanding of how the data were structured within the database in order to retrieve, insert, update or delete records from the database.

So how does one find data in a relational database, if there is no map to follow? Well in the relational model, operations that manipulate data do so, on the basis of the data values themselves. Thus, if one wish to retrieve a row from a table for example, one could do so by comparing the value stored within a particular column for that row of some search criteria. For example, “give me all the rows from the “STUDENTS” tables which have “Patricia” in the “FIRST NAME” column. The database might return a list which looks essentially like this:

|  |  |  |  |
| --- | --- | --- | --- |
| Patricia | Amadike | SID-001 | 2135 |
| Patricia | Gale | SID-268 | 818-934-5069 |
| Patricia | Monica | SID-991 | 310-234-6475 |

One could also use the data from a retrieved row to query another table. For example, you want to know what grade Patricia Amadike received in operations research 421. In this case, the student ID number from the previous query is used as a keyword in the next query. Thus, the query could be “I want the row in the operations research 421” course table where student ID equals “SID-001”.

This data access methodology makes the relational model a lot different and better than the earlier database models because it is a much simple model to understand. This is probably the main reason for the population of relational database systems today.

Another benefit of the relational system is that it provides extremely useful tools for database administration. Essentially table cannot only store actual data but they can also be used as the means for generating Meta data about the table and field names which form the database structure, access rights to the database, integrity and data validation rules etc.

**2.4 DATABASE MODEL**

Thus everything within the model can be stored tables. This means that many relational systems can use operations recursively in order to provide information about the database. In other, words a user can query information concerning table names, access right, or some data and the results of these queries would then be presented to the user in the form of a table.

However, there are many types of databases and all of them will be useful for web applications. In particular, it will be the client server database rather than the stand alone packages that will be adopted for the web.

A client server database works like this: a database server is left running 24 hours everyday. Thus, the server can handle database requests at any hour. A database request comes in from “clients” who access the database through its command line interface or by connecting to a database socket. Requests are handled as they come in and multiple requests can be handled at a time. For network applications, that must be available for worldwide time zone usage. It is essential to build a client server database which can run at all time.

**2.5 DATA RELATION AND KEYS**

A relation is defined as a set of tuples. By definition all elements of a set distinct, hence all tuples in a relation must also be distinct. This means that no two tuples can have the same combinations of values for all their attributes. Any set of attributes of a relation scheme is called super key the set of all its attributes. A key is a minimal super key i.e. a super key from which we cannot remove any attribute and still have the uniqueness constraint hold.

In general, a relation scheme may have more than one key. In this case, each of the keys is called a candidate key. It is common to designate one of the candidate keys as the primary key of the relation. A foreign key is a key in a relation R but it is not a key, rather just an attribute in other relation R of the same schema.

**2.6 ENTITY AND INTEGRITY CONSTRAINTS**

The notion of entity integrity arises from the choice of a primary key while referential integrity arises from the choice of foreign keys. In a relational database, a primary key is a set of attributes designed by the user. Each tuple in a relation is uniquely identified by the primary key values. In addition, the primary key must be a minimal set of attributes for which this uniqueness property holds. As quoted from codd’s definition of entity, integrity (2021). “No component of a primary key is allowed to have missing value of any type”. This is because the primary key value is used to identify individual tuples in a relation hawing NULL values for the primary key implies that some tuples cannot be identified. Thus, referential integrity constraints are used to maintain the consistency among tuples of relations. The referential integrity constraints states that a tuple in one relation that refers to another relation must refer to an existing tuples in that relation. Ion the other hand, a prime attribute of a relation R is an attribute of a relation schema R, if it is a member of any key of the relation R. Consequently, a non prime is an attribute, if it is not a member of any candidate key.

**2.7 DATABASE NORMALIZATION**

According to Elmasiri and Navathe (1994), the normalization process as first proposed by Codd (1972), takes a relation schema through a series of test to certify whether or not it belongs to a certain normal form (NF). Initially, Codd proposed three normal forms, which he called 1st, 2nd and 3rd normal forms. A stronger definition of 3NF was proposed later by Boyce and Codd and is known as Boyce Codd normal form (BCNF). All these normal forms are based on the functional dependencies among the attributes of a relation, later a fourth normal form (4NF) and a fifth normal form (5NF) were proposed, based on the concept or multi-valued dependencies and join dependencies respectively. Normalization of data can be looked on as a process of organizing data in a database. Or processes during which unsatisfactory relation schemas are decomposed by breaking up their attributes into smaller relation schemas that posses desirable properties. One objective of the normalization process is to improve flexibility and to ensure that redundancy and inconsistent abnormal do not occur.

**Normal forms provide database designers with:**

* A formal framework for analyzing relation schemas based on their keys and on the functional dependencies among their attributes.
* A series of test that can be carried out on individual relation schema, so that the relational database can be normalized to any degree. When a test is carried out, the relation individually meets the normalization test as outlined below.

**FIRST NORMAL FORM**

A relation is in first normal form (INF) if and only if all underlying simply domains contain atomic values only. Atomic data is a form of minimalism for data items. Thus, INF tends to:

* Eliminate repeating groups in individual tables
* Create a separate table for each set of related data
* Identify each set of relation data with a primary key.

**SECOND NORMAL FORM**

A relation is in second normal form (2NF) if and only if it is in 1NF and every non key attribute is fully dependent on the primary key. Where the INF deals with redundancy of data across a horizontal row, 2NF deals with redundancy of data in vertical columns. Thus, 2NF tends to:

* Create separate table for set values that apply to multiple records
* Relate these tables with a foreign key.

**THIRD NORMAL FORM**

A relation is in third normal form (3NF) if and only if it is 2NF and every non-key attribute is non-transitively dependent on the primary key. It tends to:

Eliminate fields that do not depend on the key.

**BOYCE-CODD NORMAL FORM**

A relation is in Boyce Codd normal form (BCNF) if and only if every determinant is a candidate key. A determinate is any attribute on which some attribute is (full) functionally dependent.

**FOURTH NORMAL FORM**

A relation R is fourth normal form (4NF) if and only if wherever there exist a multiplied dependency in the R. Thus this form prohibits independent multi-valued components of the key. For example, if an employee can have many skills and many dependent, you would move the skills and dependents to separate tables as they are not related in any way.

**FIFTH NORMAL FORM**

A relation R is in fifth normal form (5NF) or projection join normal form (PJNF), if and only if every join dependency in R is a consequence of the candidate keys of R. it advocates that you continue splitting the structure down until either of the two states exists or further splitting would be travail.

**DOMAIN KEY NORMAL FORM**

This defines a stricter form that takes into account additional types of dependencies and constraints. The idea behind domain key normal form (DKNF) is to specify, (theoretically at least) the ultimate, normal form” that takes into account all constraints and dependencies that should hold on the relation can be enforced by enforcing the domain constraints and key constraints specified on the relation.

Furthermore, normal forms, when considered in isolation from other factors, do not guarantee a good database design. It is generally not sufficient to check separately each relational schema. Rather the process of normalization through decomposition must also confirm the existence of additional properties that the relational schemas, taken together, should posses. Two of these properties are:

* The loss/less join or non additives join property, which guarantees that the spurious tuple problem does not occur.
* The dependency preservation property, which ensures that all functional dependency is represented in some of the individual resulting relations.

It is however, important to point out here that these normal forms: BCNE, 4NF, 5NF and DKNF do exist but are rarely considered in practical design. Disregarding these rules should not affect functional as originally intended.

**THE OPEN SOURCE MODEL**

This is a generalized concept for free software development and acquisition. It is often confusing to learn that an open source company may give its products away for free or for a minimal cost. How then do open source companies make up for the cost? While it is true that an open source business may not make money directly from its products, it is untrue that open source companies do not generate stable and scalable revenue streams.

In actual, in the 21st century, web technology market, it is the open source company that has the greater long strategic advantage. Companies such as LINUX, Apache, MySQL and Netscape, a host of web specific technologies such as Java have demonstrated this.

**2.8 LOCAL NETWORKS AND THE INTERNET**

As Nigerian born scientist, Philip Emeagwali (1999) puts it, “the internet is the greatest of all networks, the network of several networks (usually local network0 in its pool. The internet was not invented in 1993 by a single individual, as is widely believed. The internet is product of a succession of inventions that occurred in the 1970s and 80s.

The dream behind the web is of a common information space in which we communicate by sharing information. It universality is essential the fact that a hypertext link can point to any thing be at personal, local or global, be it draft or highly polished. There was a second part of the dream too, dependent on the web being so generally used that it became a realistic mirror (or in fact the primary embodiment) of the ways in which we work and play and socialize. That was once the state of our interaction, was on line we could then use computers to help us analysis it make sense of what we are doing, how we better work together.

With the dramatic flood of rich material of all kinds onto the web in the 1990s, the first part of the dream is largely realized, although still very few people in practice, have access to intuitive hypertext creation tools. The second part has yet to happen but there are signs and plans, which make us confident. The great need for information about information, to help us categorize, sort, pay for, own information is driving the design of languages for the web design for processing by machines, rather than people. The web of human readable document is being merged with a web of machine understandable data. The potential of the mixture of humans and machines working together and communicating through web could be immensed.

According to Owo Abidemi E. (2021), “there has been lot of improvement on the web programming concepts. We had the top down and bottom up the procedural and structured, the object oriented and event driven programming methods of software application and information generation to meet users requirements:. A major initial motivation for both the early networks. ARPANET and internet was resource sharing, connecting the two together was far more economical than duplicating these very expensive computers. However, while file database transfer and remote login (Telnet) were very important applications, electronic mail has probably had the most significant impact of the innovations from that era. Email provided a new model of how people could communicate with each other and changed the nature of collaboration. A key concept of the internet is that it was not designed for just one application, but as a general infrastructure on which new application could be conceived, as illustrated later by the emergence of the World Wide Web. It is web evolution will bring us new applications. Internet telephone and slightly further out, internet television. It is changing to accommodate yet another generation of underlying network technologies with different characteristics and requirements, from broadband residential access to satellites.

The intent of these applications is generally to promote a product or service or actually sell a product or service over the network, be it local or global (Bob-B1,M 2018). Thus, attracting and keeping a target audience in an important accept of web programming.

**CHAPTER THREE**

**Detailed Analysis of the existing system**

**3.1 Introduction**

The project used a Relational Database Management System (RDBMS) design concept. It satisfies the ACID (atomicity, consistency, integrity, and durability) properties required for a database design (Waqas et al., 2015). The stages of our design are as follows. In first stage, we developed database tables and identified the various tables and attributes related to each table. We gathered the required information from student bio data, student semester registrations, courses registered by the students in each session, a student information handbook and institutional rules and regulations guiding undergraduate studies. This information was divided into tables (entities) with attributes as columns while each row in a table represents a recording unit of information. We populated these tables with available information and carefully avoided repetitions. In the second stage of development, we have defined the primary key as attributes or set of attributes that uniquely identify a tuple (row) within each table. Using foreign keys, we established relationships among the various tables in the database (such as one-to- one, one-to-many, and many-to-many) to enforce entity and referential integrity constraints (Harrington, 2016; Olivera, 2019).

**3.2. Detailed Analysis of the existing system.**

**INPUT PROCESS:** The required data entry will involve HTML forms with adequate validation for onward processing. Forms contain information unto themselves (the names of buttons, the labels for fields, and the values of check boxes and radio buttons). Forms also accept information, as in type in boxes. Forms should be designed to make its own information accessible, make it easy to manipulate.

**REGISTER**

Sign up as a new user

Note: This sign up form is for registered students only.

You must use a valid E-mail address to receive

Your username and password via E-mail.

First Name:

(Maximum 10 characters)

Last Name:

E-mail:

Your E-mail will be used as your username

Password:

Sign up

**Course Registration**

Submit

Department:

Level:

Course 1:

Course 2:

Course 3:

Course 4:

Course 5:

Course 6:

Course 7:

Course 8:

Course 9:

Course 10:

Course 11:

Course 12:

Fig 3.2 Simple HTML form

HTML forms are a means of collecting information. Please fill in a form and/or select something. Then they click a button. Setting up a form is one thing but processing its contents is another. This project design was built using the famous method “get” and method = “post” attributes. Method = “GET” is used if you want to sent information somewhere via a browser URL.

e.g.:http://fedpoly.edu/transcript.php?stud\_id=586&deptcode=013

In the above URL, the part after the question mark is information sent to transcript. Php. This method is not secure since they are visible to user method = “POST” is the most common method used to send information from a form to an information processing program or function. This is the method used when sending form information to PHP script functions. Data can also be sent directly by employing the basic database manipulation command: INSERT which adds a new record to a specified table. For example, the above input form can alternatively be added using:

INSERT INTO POLY USERS

(FIRST NAME, LASTNAME, E-MAIL, PASSWORD)

VALUES

(‘Uche’, Uzokwe, [Uche@outgun.com’,password](mailto:Uche@outgun.com',password)’);

However, most PHP programs are written to accept information with Post method, some to accept only the GET method.

**OUTPUT FORM ANALYSIS**

The output is somehow closely tied to the input design. Invariably, HTML FORMS would also be adopted for reports and message prompts. Some output will be in formatted HTML pages, which are printable if needed.

Department:…………………………………………

Level:………………………………………………..

Score Grade

Course 1:………………………………. ……………. ……………

Course 2:………………………………. ……………. ……………

Course 3:………………………………. ……………. ……………

Course 4:………………………………. ……………. ……………

Course 5:………………………………. ……………. ……………

Course 6:………………………………. ……………. ……………

Course 7:………………………………. ……………. ……………

Course 8:………………………………. ……………. ……………

Course 9:………………………………. ……………. ……………

Course 10:………………………………. ……………. ……………

Course 11:………………………………. ……………. ……………

Course 12:………………………………. ……………. ……………

Print

Fig. 3.2.0: Simple HTML Output form

**3.2.1**. **Problems of the existing system.**

As individuals, co-operate bodies or even a nation; we are confronted with a lot of problems everyday such as problems relating to education, technology, physiological and psychological aspects of life. In other to solve these problems, we have to make strong decisions as to methods and steps of solving the various problems. To be able to make headway, we need to conduct research. According to mini-chambers dictionary, research is a careful and scientific study or investigation into a system, to try to find out new ways of doing things that will replace, the old ways. Therefore, research is considered as the process of arriving at a dependable solution to given problem through the systematic collection, analysis and interpretation of data.

In Federal Polytechnic Nekede, students’ academic information are manually kept. This process is slow and prone to errors and data redundancy. The following problems are experienced:

1. Manual course registration tends to be slow and inconsistent.
2. Many hours are spent on grading students results, compiling and storage, this leads to late release of result.
3. Scheduling and releasing academic calendar pose a challenge as well as a lot of inconsistencies and clashes are always experienced.
4. The system hampers fast decision making due to the slow manual nature of processing.
5. The system is cumbersome and cost of labour is high.

**3.3 RESEARCH METHODOLOGY**

In the design of the system, modular programming was adopted. This structural approach was TOP-DOWN methodology. Here, the system id designed in levels, consisting of one or more modules. In practice, a module can be a “Routine” or “Procedure” or “Page” in this case consisting of a sequence of “Calls” or “Links” to another “page” or “Routine” having only one “Entry and “Exit” point. Each data page or routine contains a link and unique identify tag as a provenance.

The design will focus on objective, model, constraints, actual programming and installation. In addition, routine testing and report documentation would be maintained. At this stage, it is important to have the system specification that explains what each module performs in relation to the problem.

**PRIMARY DATA**

These are data collected directly from respondents relevant to the subject under investigation. The primary data used in this case is interview.

**THE INTERVIEW APPROACH**

The interview approach was used to gather facts and ideas for this research work. Interviews were conducted in the exams and records of federal polytechnic Nekede Owerri which involved about 5 (five) persons, old and new students were also interviewed.

**SECONDARY DATA**

These are already documented materials collected from a second hand source. In the case of this project, most of the data collected are published documents on the internet as well as reference to published and unpublished materials relevant to subject under investigation.

**3.4. OBJECTIVES OF THE NEW SYSTEM.**

In this study, we aim to design an efficient, scalable and performance driven database system for processing student academic records in Federal Polytechnic Nekede Owerri. We used an efficient database design technique to achieve the following goals. To correct salient errors in computerized results. Salient errors are the errors which are easily discovered from processed results such as, wrong computation of Grade Point Averages (GPAs), errors due to omissions of courses and grades, and errors due to irrelevant repetitions.

1. To automate the discovery and prevention of inconspicuous errors in the processed results.
2. Inconspicuous errors are, in most cases, not easily discovered, especially at the time of results presentation.
3. To provide checks in the required minimum or maximum total course units that a student is allowed to register in a semester.
4. To determine if students satisfy the minimum requirements at each level of study. Also, this feature is used to determine if a student has met the required minimum criteria to graduate from the study.
5. To prevent students from registering non-prescribed courses. Such courses, if passed, will lead to an increase in the total number of units students pass, and creates wrong perceptions that such students have passed all the required courses.

**3.5 FEASIBILITY STUDY**

With the proposed design, staff, students and any authorized persons can get all the available information from the polytechnic intranet integrated database. They can get their syllabus, academic calendars and results, register course, to mention just a few.

**3.6. NEW SYSTEM STRUCTURE (Program structure)**

**FILE DESIGN**

Invariably, this serves as a storage facility: the core of the database structure and it also defines the various files that will house the new system and how data will flow within this system. Basically, the HTML and related MYSQL forms and table based configuration is adopted. This will also incorporate customized PHP scripts for an efficient menu driven and data processing system.

The code you write

Database Driver

DBD Oracle Module

DBD Oracle Module

Database Driver

Database Driver

DBD MYSOQ Module

DBI Module

Web PHP Scripts

Web Servers

Web Browsers

Fig. Data flow layout.

When designing a database, it is not advisable to model the structure after the physical hierarchy of the department or organization. This is to avoid issues of unfamiliarly since your audience might involve both local and remote users. It is important to always view the structure of the organization as if you have no inside knowledge. In addition, a database site designed based on the content of the pages is a better choice.

In this respect, the method used for navigation and data capture/display needs to be intuitive enough that someone not familiar with the site or organization will be able to easily find the information they desire. They should be able to extract data and link to any page. In the same vein, all link names must make sense without the context of the rest of the site. When it comes to choosing a method of navigation, it is useful to consider all levels of web experience from the novice to the expert user, it is never a good idea to leave graphical buttons on a web page without offering some type of textual explanation as to where they link. However, cautions should be exercised, as anything graphical will impede on the download speed.

**DATABASE (LAYOUT)**

The table structure(s) below where confined within a single database and several associated links implemented to give a relation schema. This resultant structure was compact, offered very fast B-trees indexing and full text search mechanism that can support data sizes of up to eight (8) billion bytes. Undoubtedly, the MYSQL benchmark test proved satisfactory with the proposed data structure.

|  |  |  |  |
| --- | --- | --- | --- |
| **Table** | **Records** | **Type** | **Size** |
| Country | 25 | MYISAM | 54.2KB |
| Course | 23 | MYISAM | 24.1KB |
| Department | 21 | MYISAM | 32.1KB |
| Faculty | 17 | MYISAM | 14.1KB |
| Programme | 4 | MYISAM | 2.1KB |
| Province | 46 | MYISAM | 42.1KB |
| Staff | 16 | MYISAM | 34.0KB |
| State | 30 | MYISAM | 22.1KB |
| Transcript | 1 | MYISAM | 4.0KB |
| Tuition | 9 | MYISAM | 7.0KB |
| **11 Table(s)** | **209** | **……………….** | **299.9KB** |

Fig……. Database implementation table(s)

**3.7. SYSTEM DESIGN/MENU SPECIFICATIONS**

To create and maintain a computer database, you need a database program often called a database management system or DBMS. Just as databases ranges from simple single table lists to complex multi-table systems, database programs, too range in complexity.

Some, such as the database component of Microsoft works, are designed purely to manage single file databases, with such a product you cannot build a multi table databases. You can certainly create numerous tables for storing different types of information, but there’s no way to link information from one table to another. Such programs are sometimes called flat file databases or list managers. Other database programs called relational database programs or RDBMS are designed to handle multi-file databases. This project adopted the MYSQL that produces a full featured relational database management available either as a serve program or as a client suits.

* + 1. **SOFTWARE REQUIREMENTS**

The software deployed for the implementation is PhPDev v5 (4.2.3) an open source package containing MYSQL v4.0.0m, Apache V1.3.2, Perl v5.6.1, Zend engine v1.2.0, and optimizer v1.3.1. Other versions of these applications are available but PhPDev 5 was adopted because of the bundled easy deployment process, free licensing, stability and low hardware requirements at an impressive process or utilization. The apache version is from the mod\_perl binary distribution for Win32 of Perl 5.6.1 built with VC++ 6.0 (5p5) under windows NT/XP. It also includes an associated Apache 13.27 binary. The Apache binary also includes mod\_ssl (2.8.11.3.27) based on open SSL (0.9.6a) and Perl sources of build 633 provided by Active state. The MYSQL driver was compiled with the Win32 binary (version 4.0: x-nt) package.

To successfully run PHP scripts, the web sever expects these scripts to be in a known place ­­- the & DOCUMENT\_ROOT, Only files in or below the $ DOCUMENT\_ROOT can be passed by apache to PHP, Parsed, and then displayed on any browser. Min Phpdev5- the $DOCUMENT\_ROOT is C:/Phpdev/www, so all Php/html scripts was placed in there. It is worth white to note that any other item required for the website need also to be in this location.

* + 1. **HARDWARE REQUIREMENTS**

The new system would easily be integrated into the existing polytechnic network, and can subsequently be extended to include other sections and divisions. Technically the following hardware is required:

* Intel Pentium 80386 or better workstations with configurations based on availability of funds.
* Display monitor with a minimum of 640 x 480 resolutions and at least 256- colour depth.
* Microsoft windows 98, windows NTA 0, windows XP/2000 with support for TCP/IP suite of protocols.
* 8MB of minimum memory or a higher RAM module.
* An active local network connection using the existing 24-port 10/100mbps inter-network packet.

**CHAPTER FOUR**

**IMPLEMENTATION**

* + 1. **CODING –** see Appendix
  1. **PROGRAM TESTING AND DEBUGGING**

Debugging involves the identification and removal of localized implementation errors or bugs from a program or system. Programs are typically integrated in a top down incremental fashion for ease of testing. In this regard, websites should be adequately evaluated for accessibility preferably, with real people. Consultants and lab technicians can also be hired to perform user testing depending on affordability.

It is also important to develop a strategy for maintaining a website once it’s up and running, periodically debugs the site for broken links and out of data information. it is also a good idea to idea to keep a list of which pages containing time sensitive data and how of ten they need to be updated. For example, course syllabi and exam schedule need to be updated once in a semester, while a page listing upcoming events probably should be updated weekly. You should come up with a calendar of when different pages need to be updated and make sure that someone is in charging of updating that information.

The second evaluation, issue that should be recognized is that the current site is always considered the prototype for its replacement. What this means is that we should always be thinking about ways to improve on the site. It is a good idea to write all ideas down and implement several changes all at once rather than implement small changes periodically. An important aspect of improving on a site once its up, is to play attention to user input in case if the viewers takes time to send an email message commenting on your site, then, the necessity of considering what they say is very vital whether good or bad. The inputs of these viewers will help and elaborate the parts of the site that are most useful and well visited, because visitors will let you know what aspects of the page prove to be incompatible with their computer systems and what material is just hard to see or laid out in a confusing manner. Viewers input are perhaps the most important feedback one will get in regards to the database and website.

**MENUS AND SUBSITES**

In this project, the data site is assumed small and will probably need a number of submenu pages that users enter, from a general category listing on the home page. In complex sites with multiple topic areas, it is not practical to burden the home pages with dozens of links the page grows too long to load in a timely manner, and its sheer complexity may be off putting to many users. Providing a submenu page for each category will create mini-home pages for each section of the site.

For specialized detailed submenus, it is recommended that frequent users link there directly. In this way the submenus will become alternate home pages in “subsites” oriented to a specific audience.

1. Academic result
2. Tuition and fees confirmation
3. Admission, course and student registration

Major polytechnic units to be effected will include:

1. Schools and its departmental units as concerns individual students.
2. Academic registry to ameliorate student records processes
3. Exams and records to ease transcript preparation
4. Bursary division for efficient tuition management
5. Library and consultancy services.

Profile mgt.

Result checking

Course registration

Fees confirmation

Result posting

Student management

Student

Login

Student Registration

Admin. Login

Main menu

Pin generation

**Admin login:** Panel that enables the site administer to login into the portals control panel.

**Pin generation:** This facility enables the admin to generate pin codes that will be used for registration by the students as well as to make fees payment online.

**Student management:** Enables the admin to manage students’ records and profiles in the control panel.

**Result posting:** Enables the site admin to input students result into the system.

**Student registration:** Panel that enables the students to register with the portal

**Students’ login:** Enables the registered students to access their personalized area of the portal.

**Fees confirmation:** Enables the students to confirm the payment of their fees online.

**Course registration:** Enables the students register their semester courses on the portal.

**Result checking:** Enables the students check result posted by the admin.

**Profile management:** This enables the students to update their personal profile.

* 1. **CHANGEOVER PROCEDURE**

Typically, it is advisable to run the old information system alongside the new one at the same time until management decides that the old system can be turned off. This parallel changeover procedure would ensure that cove institutional activities are not. Interrupted however, an incremental (phased) change form the old system to this new information system, starting with one or a few functional components and then gradually extending the procedure to cover the entire new system. Other several maintenance and hand over strategy must also be observed. The maintenance strategy is really something that needs to be thought about before the database; web page and image begin to populate the site. The most important thing to remember when creating a data web page for a department or organization is that one will not be maintaining a site forever. Eventually it will become someone else’s job. It is up to the design whether the handover will be smoother or full of confusion and duplicate work.

The biggest problems that can occur when handling, over maintenance of a site is that:

* The new webmaster will not know where any of the file(s) or image(s) are located and
* People will not understand the method or design process one has gone through to create the database page(s).

The problem of not being able to find the files can be greatly relieved with a well designed directory structure. Every organization with a website should have a subdirectory where they store all their images and another subdirectory to store any scripts or programs etc. Other information may warrant holding related data.

At the beginning of this project, the question was asked: “How does a webmaster create a site that can be competitive such an over populated environment? The answer to that is simple plan your site carefully, make it clear and easy to access, keep the information updated and finally and most importantly, giving attention or any attention to the audience because it is around the audience that the success of the whole site revolves.

* 1. **DOCUMENTATION**

Documentation is undoubtedly necessary in every new system development cycle. It is the detailed information about a systems design specifications, its internal work is and its functionality. It generally involves written and other visual information about an application system, how it works and how to use it.

The author agrees that documentation tools should seek to better extract knowledge from core resources. These resources include the systems source code, test code and changes to both. Resulting technologies could then help reduce the effort for documentation maintenance something that is shown to rarely occur.

In this project, Apache documentation is included at C:/apache/htdoc/manual, while that of mod\_perl and mod\_ssl is at C:/apache/htdocs/manual/mod/mod\_ssl. Respectively. Links to these are found by running the web server. The standard perl documentation, as well as that for all available locally installed modules, is available on uncer/perl/html.

On a final note, any level of a functional database together with its website should be maintained, therefore each aspect needs a documentation mechanism in place consequently, and some vital comments on the database program module, php and html page have been included for ease of accountability and future modifications. Traditionally an update of the system logs ensures each database, script, webpage and the overall website lifecycle is adequately recorded. In addition, the internet service provider (ISP) systems blueprint (if consulted) must be well understood to avoid design incompatibility issues.

**CHAPTER FIVE**

1. **RECOMMENDATION**

Designing a database driven website for an educational institution entails more of data collection than traditional programming. Nevertheless the outcomes and benefits of a well planned website cannot be over emphasized. Although many factors contribute to an effective website, yet there are so many that one could not but think of them all. The internet with all its functionality has indeed changed the way people live and interact politically, socially, economically and otherwise. The internet as with database combines audio, video and textual contents while eliminating time and space of traditional media.

This has enabled website, an interactive media when integrated with a database making it a cost effective one on one educational tool. Thus, if federal polytechnic Nekede and its contemporaries should utilize this honey comb, literacy in African as a whole would wear a glorious crown.

To this end, the researcher suggests the following recommendations:

* Implementing a database local website for federal polytechnic Nekede will not only enhance management communication but also accomplish effective resource utilization in terms of information sharing and decision making. It is therefore recommended that educational institution, federal polytechnic Nekede inclusive adopt an interactive system. This proposal is 70% achievable considering the already existing equipment, which only requires little manpower.
* This implementation will also allow for staff, student and management interaction through live survey, chats, forms and emails, while helping to improve the knowledge and understanding of the institutional needs and interests.
* Undoubtedly, a well incorporated webbed database for federal polytechnic Nekede, owerri would improve the customer support service by saving money and time while expanding on its educational distribution.
  + 1. **AREAS OF FURTHER IMPROVEMENT**

1. The polytechnic should provide an efficient internet network by incorporating its existing computers into an enterprise local network. Staff and students should be sensitized to leverage its resources such as chart and voice messaging, classroom conferencing, catalogues, library and other academic information ranging from registration, lectures to result enquires.

2. In addition to the above network infrastructural enhancement, it is suggested that the polytechnic acquires its own VSAT (satellite communication). This will ensure internet connection with its counterparts in the World Wide Web.

3. A major drawback to these proposals would be unreliable power supply system. Therefore, the institution is urged to complement this with industrial uninterruptible power supply systems (UPS) that can ensure round the clock power supply for the network.

4. Develop an intensive in house training for academic and non academic staff. This will help reduce the dependence on manual processing and also increase staff competence on world class office automation.

Once, the database is fully integrated with www. Federal polytechnic Nekede will not only register the institution’s presence, but also boast its academic as a citadel of technological excellence students and individuals of academic interest will include those with mobility constraints, will find the universal access very comforting. It will also create avenue for online research and development, which in turn will create more revenue for the site.

* 1. **CONCLUSION**

The need for available, secure and reliable information solution is heightened by the increasing dependence on web like systems and database technologies to provide core educational services, develop efficient academic products, administer daily activities and perform both short, long term management functions. Using federal polytechnic Nekede, owerri as a case study this project has elaborated on the technical and operational requirement of developing an academic information system. Issue like web and database concept and technologies: analysis and design tools, security and content maintenance were adequately explained.

Initially, HTML was used to convey contents like text, graphic images, video, audio and hyperlinks in static form. But today, database and dynamic web page scripting are the communication means to live information content. These technologies have come to stay no doubt that every organization, institution and government parastatal wants to join the e-race. Designing a functional academic information website for federal polytechnic Nekede is indeed a giant step in the right direction.

**REFERENCES**

**Brendan, P.K. (1992):** Zen and the art of the internet. A beginners guide: Prentice hall publishing.

**Burstein, C.D. (1997):** Accessible site design elements: (htt://www/anybrower.org/campaign/abdesign2.html)

**Conger, S. & Richard, M. (1998):** Planning and designing effective websites Cambridge, MA: International Thompson Publishing Co.

**Date, C.J: (1990):** An introduction to database system: M.A. Addison\_Wesley publishing company 5th ed. Volume 1. reading.

**Elmasri, R. & Navathe, S. (1994):** Fundamental of database systems: The Benjamin/cummings publishing Co: Redwood city, CA: Zend, pp 143-144, 401, 407-409, 435, 438, 440, 442-443.

**Internet Society (ISOC) (2003):** A brief history of the internet (<http://www.isoc.org/internet/history/briefshtml>).

**Jeffre, V. (2001):** The art and science of web design New riders publishing.

**Jennifer, N. (2002):** Web design in a nutshell .O’Reilly books, 2nd edition

**Joe, C. (2000):** Building accessible websites. David Dwyer publishers- ISBN 0-7357-115-x.

**Robert, L. (1998):** Multiple computers networks and inter computer communication: ACM Gatlinburg conference.

2