A Model of Tertiary Institutions' Resource Management Platform Using Cloud Computing (TIRMP)

BY

AMADI, EMMANUEL CHUKWUDI B.ENG
20094738398

A THESIS SUBMITTED TO THE POSTGRADUATE SCHOOL OF FEDERAL UNIVERSITY OF TECHNOLOGY, OWERRI

IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE AWARD OF MASTER OF SCIENCE DEGREE (M.SC) IN INFORMATION TECHNOLOGY

JULY, 2012

CERTIFICATION

This is to certify that Amadi, Emmanuel Chukwudi a postgraduate student with registration number 20094738398, has satisfactorily completed the requirement for course and research work for the Master of Science (M.Sc.) degree in Information Technology, with option in Networking Technology.

Sign _____ _____

Dr. A. M. Uwaleke Date
Head of Department IMT/Supervisor

Sign _____ _____

Prof S. M. Nzotta Date
Dean, SMAT

Sign _____ _____

Engr Prof. (Mrs) K. B. Oyoh Date
Dean PG-School

Sign _____ _____

Engr Prof. H.C. Inyiama Date
External Examiner

DEDICATION

This dissertation is dedicated to God almighty, for his grace that appeared to me from the inception to the completion of this research work.

ABSTRACT

In Cloud Computing, users connect to the 'Cloud', appearing as a single entity as opposed to multiple servers. This work seeks to solve resource managerial problems ranging from server maintenance cost, risk of loss of data, internal security issues, and availability of data for proper decision making. This work presents to tertiary institutions a model for designing a scalable and cost effective platform for institutional resource management on the cloud. The platform is called the Tertiary Institutions Resource Management Platform (TIRMP) using cloud backbone. The real-time evolutionary prototyping methodology was adopted for the TIRMP model because of its continuous modification feature. TIRMP focuses on **Personnel record management.** This model can be adopted to suit any higher institution with little or no adjustments on the design interface. *Force.com* cloud computing platform was used as the cloud service provider. The work involves the design of an improved Campus Area Network (CAN) for optimised internet access, the design of a database structure using ERD diagram; capturing all the personnel parameters on the manual system, and then the transfer of the database structure to the cloud platform using objects, relationships, rollup summaries and access privileges. With cloud computing initial cost of setting up a management platform is greatly minimized. The TIRMP is a model that can be adopted by any institution for management of personnel records. It provides a robust, scalable, cost effective and zero downtime platform for resource management on the cloud.

ACKNOWLEDGEMENT

I would like to express my profound gratitude to my supervisor Dr. Amaobi Uwaleke, for the new concept on “cloud computing” he has introduced me to. Thank you for your guidance, encouragement and support throughout this project. Over the past three years our relationship has grown to be like father and son. To my mummy Dr. Mrs Eze, you have been a real help to me academically I appreciate your motherly council and advice. God bless you richly.

Many thanks to my family. I am indebted to my parents for standing by me throughout these years. To my sisters, your prayers made it a worthwhile. May God bless you all. No words are suitable to express my gratitude to my sweetheart Amaka. With you by my side I can see that the sky is just my starting point in my life career.

Each individual in 2009/2010 M.Sc. class - no names mentioned and no one forgotten - being friends and colleague are mentioned, for your kind support and encouragement. You guys have done great.

I would not fail to acknowledge my bosom friends Ifeanyi (Anyiwata), Ifeanyi Nduka, Etus, Cosmas, and Engr Areh Innocent. Your friendships were and are still valuable to me.

I would also like to acknowledge the financial support in form of scholarship provided by the Federal University of Technology, Owerri, Nigeria. Your funding of this research is well appreciated.

I appreciate and acknowledge the spiritual support from the Full Gospel Business Men Fellowship International Obinze Chapter, and the Assemblies of God Church Obinze1. Indeed the prayers of the righteous ones avail much. Thanks to you all.

My special thanks goes to God Almighty, my Father, Lord, Saviour, portion, inheritance and strength. Your grace has brought me this far. You are my all in all.

TABLE OF CONTENT

CHAPTER TWO

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE

1.0 INTRODUCTION

> *We are at the very beginning of time for the human race. It is not unreasonable that we grapple with problems. But there are tens of thousands of years in the future. Our responsibility is to do what we can, learn what we can, improve the solutions, and pass them on (Richard, 1993).*

Cloud Computing is a new paradigm in Information Technology (IT). There are several definitions of the Cloud, ranging from very broad as almost everything on the Internet to the very narrow only concerning Virtualization on servers only (Rittinghouse & Ransome, 2010). The basic idea is that anything that traditionally is possible in computing can be shifted to the cloud; Communication, scientific computing, word processing etc. In their research Vaquero and other researchers, proposed the following definition:

> Clouds are a large pool of easily usable and accessible virtualized resources (such as hardware, development platforms and/or services). These resources can be dynamically reconfigured to adjust to a variable load (scale), allowing also for an optimum resource utilization (Vaquero et al., 2009).

A closer look at the above definition provides us with the concept of virtualization of resource. Looking at the work cloud, and already existing definitions the researcher propose the following definition:

> A cloud is first a platform born from the internetworking concept. It provides easily usable resources to its subscribers on a pay as you use basis or on a user agreement level in a situation where organizations come together to put up the cloud platform for their joint benefit. A cloud as the name imply can span over small to large geographical areas and even the entire globe as being seen today. Thus in summary **a cloud is backbone**

that trunks resources to its users on an agreement level, as to usability and accessibility of available resources.

Today is the age of information technology. And according to kambil in the journal of business strategy, the facets of work and personal life are moving towards the concept of availability of everything online and understanding this trend, the big and giant web based companies like Google, Amazon, Salesforce.com came with a model named "Cloud Computing" the sharing of web infrastructure to deal with the internet data storage, scalability and computation (Kambil, 2009). A closer look at the concept of cloud computing brings us to a point of ease and availability of resources without the need of necessary software platform or compilers to go about application development

Current technological progress, particularly, in the past few decades, has demonstrated the need for performing more and more complex computations. To meet this requirement, large and complicated distributed systems have become essential. In Conventional IT environments, clients connect to multiple servers located on company premises. Clients need to connect to each of the servers separately. In Cloud Computing clients connect to the Cloud platform. The Cloud contains all of the applications and infrastructure and appears as a single entity to the user. Cloud Computing allows more efficient use of the resources by dynamically configuring resources to cater for changes in the demand for load.

1.1 The Cloud vs the Internet

The term 'cloud' is analogical to 'internet. Cloud computing is an internet based computing technology where virtual servers provide infrastructure, platform, software, devices and other resources and hosting to customers on a pay-as-you-use basis (Luit infotech, 2011). Cloud computing customers do not own the physical infrastructure rather they rent the usage form a third-party cloud provider who owns the infrastructure.

The researcher sees the cloud as a concept that is similar to the internet in operation but not same as the internet in principle and usage. The cloud platform runs on the internet platform in that the distributed servers that host application are connected to the internet. In operation, the cloud makes use of its distributed database servers for application utilization and access using the internet as a route.

Basically, the internet is made up of servers located as various points in the globe. This servers range in their capabilities and availability and has some level of limitations compared to a cloud distributed server nature. Taking a bearing from the popular www.facebook.com platform that runs on the cloud and a typical webpage of a government agency in the Nigerian environment; the facebook platform has the ability to grant access at the same time to millions of users while the government agency website may not have such capability. This difference is as a result of the fact that the cloud is a more robust platform for interconnection as is runs on a distributed server concept whereas the website runs on a single server system and might not be able stand millions of hits at the same time from users.

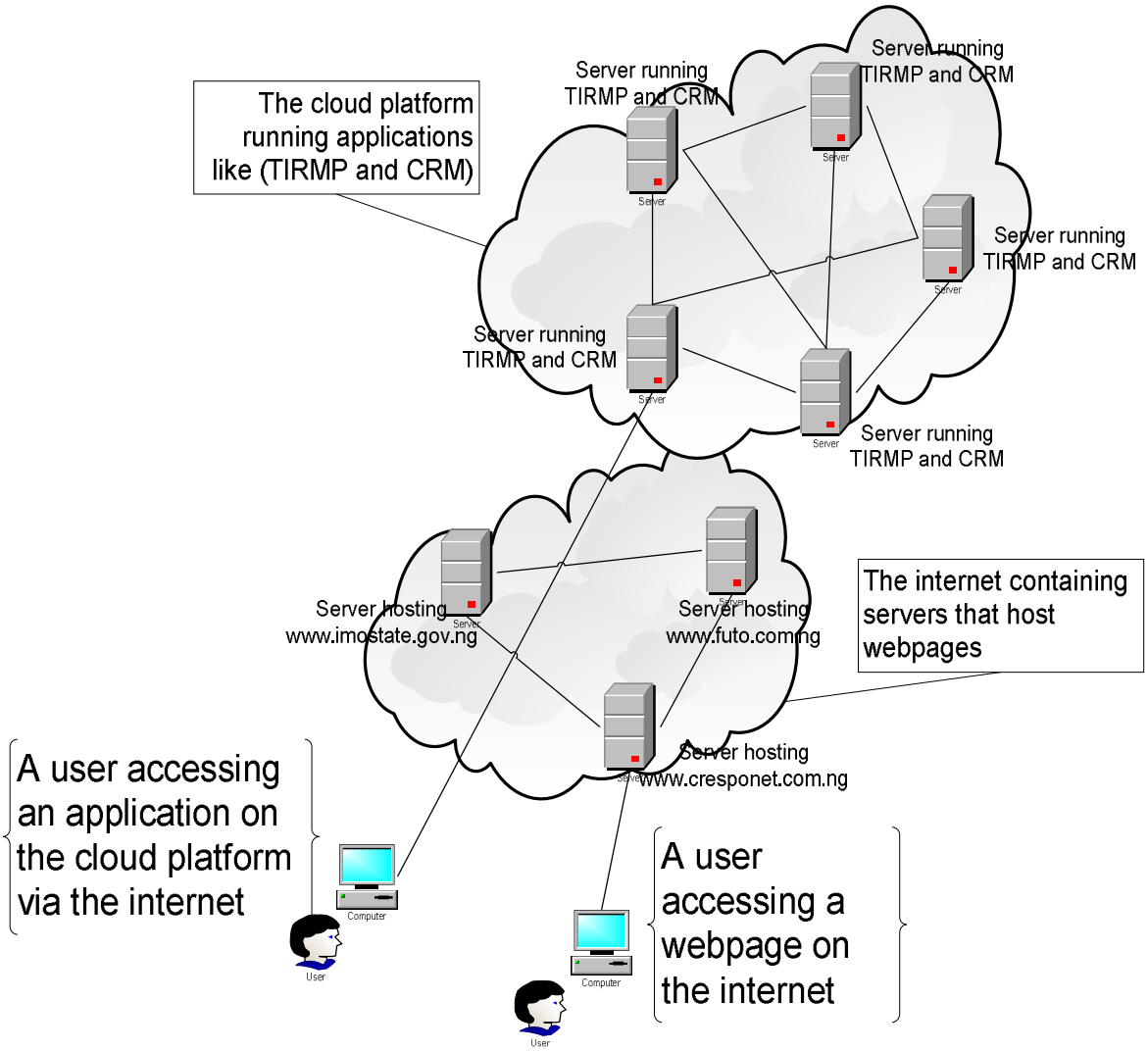


Figure 1.1: The Internet and the Cloud Platform

1.2 Cloud Computing Overview

1.2.1 Cloud Architecture

Cloud computing architecture consist of two components "**the front end**" and "**the back end**". The front end of the cloud computing system comprises the client's device (or it may be a computer network) and some applications are needed for accessing the cloud computing system. The back end refers to the cloud itself which may encompass various computer machines, data storage systems and servers. A group of these clouds make a whole

cloud computing system. The whole system is administered via a central server that is also used for monitoring clients demand and traffic, ensuring the smooth functioning of the system. A special type of software called “middleware” is used to allow computers that are connected on the network to communicate with each other. Cloud computing systems also must have a copy of all its clients’ data to restore the service which may arise due to a device breakdown; making a copy of data is called redundancy (Kambil, 2009).

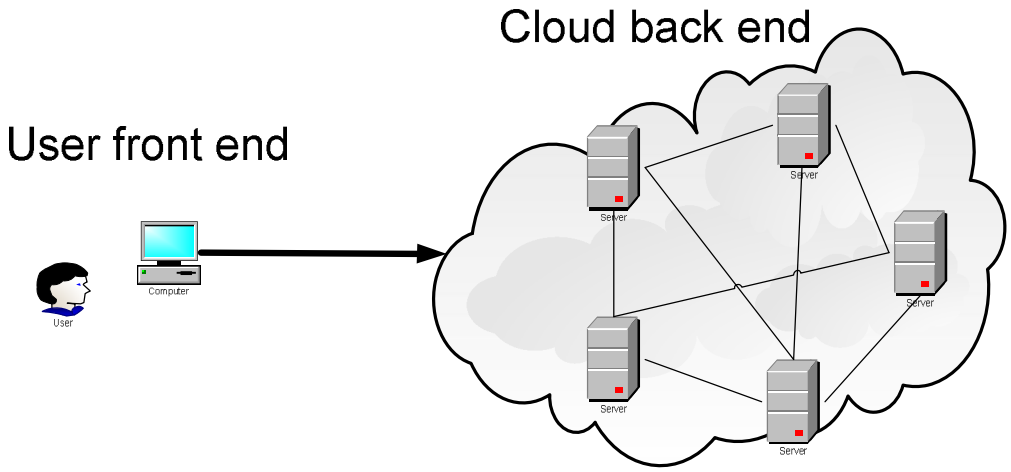


Figure 1.2: Diagram Illustrating the Cloud Architecture

1.2.2 Cloud Deployment Models

Deploying cloud computing can differ depending on requirements, the following four deployment models proposed by the Dialogic Corporation have been identified, each with specific characteristics that support the needs of the services and users of the clouds in a particular ways. These models include Private cloud, Community cloud, Public cloud, and Hybrid cloud (Dialogic Corporation, 2011).

a. **Private Cloud** — The cloud infrastructure has been deployed, and is maintained and operated for a specific organization. The operation may be in-house or with a third party on the premises. The private cloud is also referred to as internal cloud or on-premise cloud, a private cloud intentionally limits access to its resources to service

consumers that belong to the same organization that owns the cloud. In other words, the infrastructure is managed and operated for one organization only, primarily to maintain a consistent level of control over security, privacy and governance. Essential characteristics of a private cloud typically include:

- Heterogeneous infrastructure
- Customized and tailored policies
- Dedicated resources
- In-house infrastructure (capital expenditure cost model)
- End-to-end control.

b. **Community Cloud** — The cloud infrastructure is shared among a number of organizations with similar interests and requirements. In this case, organizations come together to setup a cloud for their common interest. Each organization has access to the platform based on their contributions to the overall setup of the cloud system. This may help limit the capital expenditure costs for its establishment as the costs are shared among the organizations. The operation may be in-house or with a third party on the premises. This deployment model typically refers to special-purpose cloud computing environments shared and managed by a number of related organizations participating in a common domain or vertical market.

c. **Public Cloud** — The cloud infrastructure is available to the public on a commercial basis by a cloud service provider or in some cases the government. This enables a consumer to develop and deploy a service in the cloud with very little financial outlay compared to the capital expenditure requirements normally associated with other deployment options. It is also known as external cloud or multi-tenant cloud, this model essentially represents a cloud environment that is openly accessible. It generally provides an IT infrastructure in a third-party physical data centre that can be utilized to

deliver services without having to be concerned with the underlying technical complexities.

Essential characteristics of a public cloud typically include:

- Homogeneous infrastructure
- Common policies
- Shared resources and multi-tenant
- Leased or rented infrastructure; operational expenditure cost model
- Economies of scale

Note that public clouds can host individual services or collections of services, allow for the deployment of service compositions and even entire service inventories.

d. **Hybrid Cloud** — The cloud infrastructure consists of a number of clouds of any type, but the clouds have the ability through their interfaces to allow data and/or applications to be moved from one cloud to another. This can be a combination of private and public clouds that support the requirement to retain some data in an organization, and also the need to offer services in the cloud.

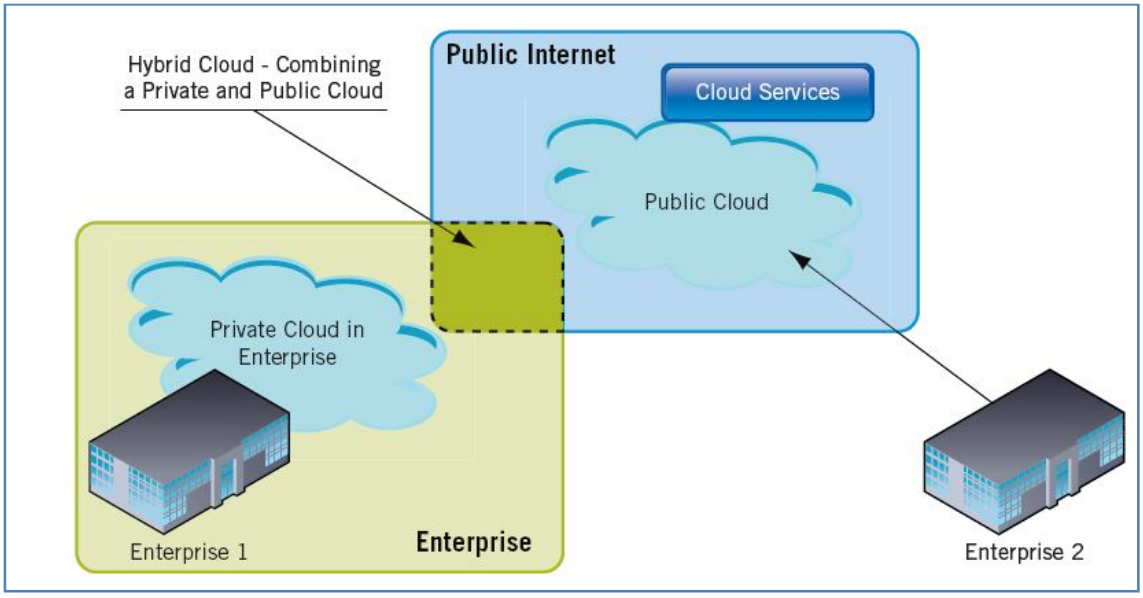


Figure 1.3: Public, Private, and Hybrid Cloud Deployment Example [6]

1.2.3 Cloud Computing Services

Many different types of services that can be delivered in the various cloud deployment environments exist (Thomas ERL, 2010). Essentially, any IT resource or function can eventually be made available as a service. Although cloud-based ecosystems allow for a wide range of service delivery models, three have become most prominent:

a. Infrastructure-as-a-Service (IaaS)

This service delivery model represents a modern form of utility computing and outsourced managed hosting. IaaS environments manage and provision fundamental computing resources (networking, storage, virtualized servers, etc.). This allows consumers to deploy and manage assets on leased or rented server instances, while the service providers own and govern the underlying infrastructure.

b. Platform-as-a-Service (PaaS)

The PaaS model refers to an environment that provisions application platform resources to enable direct deployment of application-level assets (code, data, configurations, policies, etc.). This type of service generally operates at a higher abstraction level so that users manage and control the assets they deploy into these environments. With this arrangement, service providers maintain and govern the application environments, server instances, as well as the underlying infrastructure.

c. Software-as-a-Service (SaaS)

Hosted software applications or multi-tenant application services that end-users consume directly correspond to the SaaS delivery model. Consumers typically only have control over how they use the cloud-based service, while service providers maintain and govern the software, data, and underlying infrastructure.

d. Other Delivery Models

Cloud computing is not limited to aforementioned delivery models. Security, governance, business process management, integration, complex event processing, information and data repository processing, collaborative processes - all can be exposed as services and consumed and utilized to create other services.

An Analogy

An on-premise infrastructure is like having your own car. You have complete control over when and where you want to drive it, but you are also responsible for its operation and maintenance. IaaS is like using a car rental service. You still have control over when and where you want to go, but you don't need to be concerned with the vehicle's maintenance. PaaS is more comparable to public transportation. It is easier to use as you don't need to know how to operate it and it costs less. However, you don't have control over its operation, schedule, or routes.

1.2.4 Benefits of Cloud Computing

Cloud computing provides a wide range of benefits which range form the reduction in the upfront capital expenditure on hardware and software development to the pay-as-you use billing system like phone bills. This enable user terminate contract at anytime.

Some of the possible benefits cloud computing offers it users include but not limited to the following:

- **Cost Saving:** companies can reduce their capital expenditure and use operational expenditures for increasing their computation capabilities. This is a lower barrier to entry and also requires fewer in-house IT resources to provide system support.
- **Scalability/Flexibility:** companies can start with a small deployment and grow to a large deployment fairly rapidly, and then scale back if necessary. Also, the flexibility of

cloud computing allows companies to use extra resources at peak times, enabling them to satisfy consumers demand.

- **Reliability:** services using multiple redundant sites can support business continuity and disaster recovery.
- **Maintenance:** Cloud service providers do the system maintenance, and access is through APIs that do not require application installations onto PCs, thus further reducing maintenance requirements.
- **Mobile Accessible:** mobile workers have increased productivity dir to systems accessible in an infrastructure available from anywhere.

1.2.5 Service Level Agreement (SLAs) on the Cloud

Users of the cloud can terminate the contract at any time and are often covered by Service Level Agreement (SLA) with financial penalties (Kambil, 2009). This reduces risk and uncertainty and ensures return on investment (ROI)

As consumers move towards adopting such a Service-Oriented Architecture as cloud computing, the quality and reliability of the services becomes a major bone of contention (Patel et al., 2009). It is not possible to fulfil all consumer expectations from the service provider perspective and hence a balance needs to be made via a negotiation process. At the end of the negotiation process, provider and consumer commit to an agreement. In Service-oriented architecture (SOA) terms, this agreement is referred to as a Service level agreement (SLA). This SLA serves as the foundation for the expected level of service between the consumer and the provider. The Quality of service (QoS) attributes that are generally part of a SLA (such as response time and throughput) however changes constantly and to enforce the agreement, these parameters need to be closely monitored. A more detailed insight on SLA in cloud computing is presented in chapter two.

1.2.6 Challenges Facing Cloud Computing

The following are some of the notable challenges associated with cloud computing, and although some of these may cause a slow down when delivering more services in the cloud, most also can provide opportunities, if resolved with due care and attention in the planning stages (Dialogic Corporation, 2011).

- **Security and Privacy:** Perhaps two of the more "hot button" issues surrounding cloud computing relate to storing and securing data, and monitoring the use of the cloud by the service providers. These issues are generally attributed to slowing the deployment of cloud services. These challenges can be addressed, for example, by storing the information internal to the organization, but allowing it to be used in the cloud. For this to be achieved, the security mechanisms between organization and the cloud need to be robust and a Hybrid cloud could support such a deployment.
- **Lack of Standards** — Clouds have documented interfaces; however, no standards are associated with these, and thus it is unlikely that most clouds will be interoperable. The Open Grid Forum is developing an Open Cloud Computing Interface to resolve this issue and the Open Cloud Consortium is working on cloud computing standards and practices. The findings of these groups will need to mature, but it is not known whether they will address the needs of the people deploying the services and the specific interfaces these services need. However, keeping up to date on the latest standards as they evolve will allow them to be leveraged, if applicable.
- **Continuously Evolving** — User requirements are continuously evolving, as are the requirements for interfaces, networking, and storage. This means that a "cloud," especially a public one, does not remain static and is also continuously evolving.
- **Compliance Concerns** — The Sarbanes-Oxley Act (SOX) in the US and Data Protection directives in the EU are just two among many compliance issues affecting

cloud computing, based on the type of data and application for which the cloud is being used. The EU has a legislative backing for data protection across all member states, but in the US data protection is different and can vary from state to state. As with security and privacy mentioned previously, these typically result in Hybrid cloud deployment with one cloud storing the data internal to the organization.

1.2.7 A Brief History of Cloud Computing

The underlying concept of cloud computing dates back to the 1960s when John McCarthy opined that, "computation may someday be organized as a public utility." (Wikipedia, 2011).

On the issue of the how cloud computing started, Jasmine Antonick provided a back ground as follows:

> We cycled between periods when computing was more centralized (and seemed more remote and less accessible to users) and other periods when computing were right on user desktops. No one was ever satisfied. Centralized computing failed to give users enough control and was too inflexible. Distributed computing made every user his own system administrator and was very inefficient. In the last few years, as the cost of a unit of computing power has continued to decrease – but the cost of humans with the skills to implement and manage computer systems has not – the vision of centralized computing has returned. It has taken several turns. Some computer scientists have suggested (and experimented with) a vast Grid of computers, attached via the Internet, whose power can be combined for large-scale tasks when needed. In some cases, very large computing systems can be part of these grids for specialized tasks. Others have suggested a computing Utility which would provide just as much computing power as an organization needed, on an on demand basis, much like

electricity. Eventually, as large web users such as Google and Amazon built out enormous data centres for their own purposes, they realized that they could permit others to access these “clouds” of computing power at relatively attractive prices. The Cloud computing era began (Jasmine, 2009).

Today, many companies are setting up large data centres, sometimes, these data centres are an extension of the companies need while on the other hand they are for customers to use at attractive prices. At the beginning of the era of cloud computing, the idea was that these clouds computing infrastructure would offer high processing power and storage. As the idea became more popular and several companies gaining ground in the technology, additional services like the provision application have been introduced.

1.3 Problem Formulation

At the apex of this thesis lies the underlying challenge of institutions needing an ever increasing level of data storage, manipulation and retrieval. The current climate within which academic institutions are operating is particularly challenging as the manual system of data storage, retrieval and manipulation within institutions cannot cater for the teaming population currently being experience. The problems currently being experienced in Resource management within higher institutions especially in developing economies can be summed up as follow:

- The problem of loss of data stored in file cabinets or on single computers/servers, arising due to fire outbreak, mismanagement or equipment failure.
- High cost of replacing obsolete equipment due to constant change in technology and increased traffic on the existing system
- High cost of maintaining local servers due to inadequate power supply

- Increased awareness on the availability of improved computing platforms giving rise to a constant increase in the demand for convenience in resource management.
- The gap created between staffs as a result of poor information flow across the institutional setting.
- The need for a more cost efficient platform for information resource management.

With the growth of IT over the last few years the solution to this problem is not far from our reach. With the advent of cloud computing, the problems experience in the management of data storage, retrieval and management is gradually becoming the thing of the past. As being experience in the social networking platforms like “facebook” and “tweeter”, cloud computing can be conveniently adopted for the management of resources within in academic institutions.

1.4 Objective of the Design

Tertiary institutions across the world are rapidly growing both in size and, student population and staff strength. Tertiary institutions include: Universities, Polytechnics, Colleges, training centres and lots more. The tertiary institution is one of the key sectors in any country and plays a vital role in developing professional labour force. As institutions grow there is thus need to improve resource management to be able to meet up with demands at various levels. This will help to reduce the lags in information dissemination and record collation.

Historically, the administration of institutions has used spreadsheet and document related packages like Microsoft excel and work to manage resources both human and material. However with the growth of institutions and the growing need for better data storage and management platforms, it has become evident that unless the current process of resource management is replaced by one that is more collaborative, reliable, and scalable, institutions will not be able to satisfy its populaces.

Tertiary institutions need a centralized application that can aggregate all of its staff and student record together.

The approach to solve this problem as presented in this paper is to leverage a salesforce account to build a resource management application on the force.com platform. In other words, institution is being introduced to the world of cloud computing.

This study intends to develop a cloud computing platform for Information Resource Management for academic institutions in Nigeria. This platform is called the "Tertiary Institution Resource Management Platform" (TIRMP). The major objectives of this work are as follows:

1. Development of a database structure for the four key resources within the academic institutions in Nigeria namely: Staff, Student, Books/publication, and equipment. This structure can adopt to any platform for development
2. Virtualization of the management of both human and material resources within tertiary institutions using cloud computing. This will provide for easy access, retrieval and manipulation. The force.com cloud platform will be used in this work.
3. Implement security roles, privileges and authentication within the cloud platform to ensure data security and integrity.
4. This work also provides a proposal for developing a robust, cost effective and efficient internet infrastructure design with a scalable bandwidth adjustment table. This design can be adopted by any institution in Nigeria.

This project seeks to automate the management of resources within tertiary institutions with the following specific objectives:

1. Track all the personnel records in the institution both academic and non academic
2. Track staff publications, books, research works at various levels.

3. Track teaching and professional experience of staffs and their growth in the institution for the purpose of appraisal for promotion.
4. Allow for the appraisal of staff members at various levels
5. Track available vacancy within the institution and provide a list of possible candidates to fill those vacancies
6. Keep track of employment status of staff members provide such information to the staff members in real time.
7. Inform staff members of necessary steps to take as regards their growth within the institution.
8. Automate the posting of vacancies to the institutions website.

1.5 Scope of the Design

This work focuses on one of the major modules of the TIRMP platform; the personnel management module which is refereed to as the Tertiary Institutions Personnel Management Platform (TIPMP). The TIPMP platform is a platform that runs on the cloud and automates the management of personnel records within higher institutions. The major aspects the TIPMP handles includes; personnel records (for academic and non academic personnel), personnel appraisal and promotion, capturing of staff qualification/publication and conferences attended, and staff chatting.

The TIPMP requires a robust database structure and an efficient campus area network for internet access. All of these will be included the development of the TIPMP of the TIRMP platform.

1.6 Relevance of the Design

As the battles between PC vs. Mac and IE vs. Firefox rage on, there is another battle being waged that is going relatively unnoticed: legacy computing vs. cloud computing (CRM,

2011). Legacy computing involves programs that are installed on a computer's hard drive or a local server; cloud computing refers to software services and platforms that are offered through the Internet. An easy-to-understand example is e-mail; some people access e-mail from a program on their computer while others use e-mail services online such as Gmail or Yahoo mail. Cloud computing services, while not as new as they may seem, are quickly becoming the standard of choice for businesses everywhere.

Cloud computing is an efficient way to store and maintain databases, and is an especially helpful tool storage, manipulation and retrieval of data. Using a platform on the cloud for data management, helps everyone who uses it by streamlining data and procedures into one central location. It's an easy way to organize information in different departments while still allowing for institution-wide collaboration. Services offered by cloud computing are actually software, but the software is never installed on a computer; this is what's known as software-as-a-service, or SaaS. The software is accessed through the Internet, and provides storage, database creation, information management and many other business-related services.

Cloud computing solutions are often less expensive than their software counterparts, another reason why they are a becoming a popular choice. Pricing is often offered on a per-user basis, so organizations pay a flat fee based on the number of people who use it. It also saves time and money when there is a need for upgrade; cloud services are updated by the provider, so everyone is always working on the latest platform. Cloud computing users have access to their information through the Web, so it's accessible from anywhere they need it. It's also generally less expensive than other solutions; a simple return on investment (ROI) calculation can help determine just how cost-effective it can be for an organization. A big benefit that is often touted by the companies who use cloud services is that the fear of losing data, or having unsecured data, is no longer a factor. When changes are made on the cloud, they happen in real time. There's no need to save constantly in case a computer crashes. There's no in-house server

required needing daily backup, either. All of the IT worries are “virtually” gone. Security is still maintained, because information is password accessible.

What does this mean for an organization? It means that employees will have the capability to do things faster and with much more ease; since everyone is using the same system, everyone is able to collaborate and get the information they need. Many cloud based services offer customizable options to fit virtually every business need, and the option to customize it even further by writing your own programs and add-ons, or consulting to have these customizations done for you.

1.7 Organization of the Study

The structure of this report is divided into five main chapters, the introduction, literature review, design methodology and analysis, system implementation, and finally the conclusions and recommendations. The introduction presents Cloud computing, what it is, what can be done with it and the current status of Cloud computing. The literature review will present general literature that is relevant for achieving the purpose of the thesis while the Cloud computing literature review will present cloud computing specific literature. Further more the literature review chapter will elaborate on the salesforce platform to be used as the cloud service provider and the reasons behind the choice of this platform. The design methodology and analysis chapter will present how the design will be conducted, the database structure format to be used and the design method that will be adopted for the design of the platform on the cloud. The system implementation chapter will focus on the results obtain from the platform and screen shots of the platform with test data. This chapter will also provide system specification for the use of the platform. In the conclusion and recommendation chapter the empirical findings will be presented and recommendations made on the requirements to develop and run a successful cloud platform for resource management within academic institutions in Nigeria.

CHAPTER TWO

2.0 LITERATURE REVIEW

A cloud is first a platform born from the internetworking concept. It provides easily usable resources to its subscribers on a pay as you use basis or on a user agreement level in a situation where organizations come together to put up the cloud platform for their joint benefit. A cloud as the name imply can span over small to large geographical areas and even the entire globe as being seen today.

The objective of this chapter is the elaboration and the response to the scientific concepts within the upcoming chapters. At first it will be necessary to get a fundamental overview of the development of computing, its first occurrence and how it evolved. It is also inevitable to clearly clarify what cloud computing is, which concepts it involves and how it distinguishes from all the other concepts that will be explained.

Secondly, database models are presented. The major database models are explained in detail in a way to provide a basis to the framework development and the choice of the model to be used for this work.

Thirdly, information resource management is reviewed, as it evolved from the use of file systems of storage to the use of distributed database system seen in recent times.

Fourthly, various network topologies and internet solution are reviewed with their comparative advantages over each other.

Finally this chapter reviewed the concept of system development.

2.1 Computing Overview

Computing Curricula defined "computing" in the following statements:

> "In a general way, we can define computing to mean any goal-oriented activity requiring, benefiting from, or creating computers. Thus, computing includes

> designing and building hardware and software systems for a wide range of purposes; processing, structuring, and managing various kinds of information; doing scientific studies using computers; making computer systems behave intelligently; creating and using communications and entertainment media; finding and gathering information relevant to any particular purpose, and so on. The list is virtually endless, and the possibilities are vast." (Computing Curricula, 2005)

Computing includes designing and building hardware and software systems for a wide range of purposes; processing, structuring, and managing various kinds of information; doing scientific studies using computers; making computer systems behave intelligently; creating and using communications and entertainment media; finding and gathering information relevant to any particular purpose, and so on.

The term "computing" is also synonymous with counting and calculating. In earlier times, it was used in reference to mechanical computing machines (Wikipedia, “Computing” 2011).

Cloud computing which is a computing platform, is a new paradigm in Information Technology (IT). There are several definitions of the Cloud, ranging from very broad as almost everything on the Internet to very narrow only concerning Virtualization on servers (Rittinghouse & Ransome, 2010). The basic idea is that anything that traditionally is possible in computing can be shifted to the cloud; Communication, scientific computing, word processing etc Vaquero, propose the following definition:

> Clouds are a large pool of easily usable and accessible virtualized resources (such as hardware, development platforms and/or services). These resources can be dynamically reconfigured to adjust to a variable load (scale), allowing also for an optimum resource utilization (Vaquero et al., 2009).

This section of this chapter will look at the history of computing which is directly related to history of computer and the evolution of several computing platforms.

2.1.1 History of Computing

The history of computing is longer than the history of computing hardware and modern computing technology and includes the history of methods intended for pen and paper or for chalk and slate, with or without the aid of tables (Wikipedia, "History of Computing", 2011).

The idea of computing is tied to the hardware and software that are used for computations. The key hardware device is the computer which is driven by programme lines known as software. Thus computing involves the analysis, storage and manipulation of information using the associated hardware and software that is related to the particular kind of computation. Computation cuts across all aspect of science and the growing need for better and faster computation platforms cannot be over emphasised. The computer being the major computing platform in recent times has a rich history, with recent advancements that have seen its deployment both as high performance stand alone platforms and as distributed platform for computation.

As presented in Wikipedia, the earliest known tool for use in computation was the abacus, and it was thought to have been invented in Babylon circa 2400 BC (Wikipedia, "Abacus"., 2011). Although the Abacus was known from early times, it was not until about 1623-1624 that a mechanical calculator capable of carrying out basic arithmetic operations of addition, subtraction, multiplication and division was built (Encyclopaedia Americana, 2007).

In 1642, Bliase Pascal began to develop a mechanical calculator for addition and subtraction. This innovation was followed by the work of Gottfried Wilhelm Von Leibniz, who appended to Pascal's machine a device for automatically doing multiplication and division.

This new machine was exhibited in the Royal Society in London in 1673 (Encyclopaedia Americana, 2007).

On computing, Leibniz said:

> "Also the astronomers surely will not have to exercise the patience which is required for computation. It is this that deters them form computing or correcting tables…. For it is unworthy of excellent men to loose hours like slaves in the labour of calculation which could safely be relegated to anyone else if machines were used." (Encyclopaedia Americana, 2007).

The machine of Liebniz was in a sense a fore runner of the mechanical desk calculator invented by charles X. de Colmar in 1820. In 1887, Leon Bollee built into a calculator a multiplication table and obviated the need for the repeated addition of Liebniz and Colmer.

The 19th century saw great breakthroughs in the development of the computer system. Charles Babbage automated the calculation of mathematical tables to aid astronomers and then Heiman Hollerith designed a machine for the U.S. Census office towards the end of the 19th century. Another important mile was gained when Lord Kelvin developed an analog machine. Howard H. Aiken of Harvard University conceived the first large scale automatic digital computer in 1937 which was eventually built by IBM corporation (Encyclopaedia Americana, 2007)..

In the spring of 1943 work on the first all electronic computer started. The machine called ENIAC (Electronic Numerical Integrator and Computer) was completed in the winter of 1944-1940. ENIAC had a computing speed of about 5000 additions per second and about 300 multiplications per second. This great speed factor is what produces the computer revolution. Other innovations at the time ENIAC was being developed included the Colossus which was being developed in England, EDVAC, EDSAC, UNIVAC and the IAS machine developed in

1946 by Von Neumann and H.H. Goldstine. The IAS machine was completed in 1952 and became the prototype for modern electronic computer (Encyclopaedia Americana, 2007)..

In the 20th century, the key technology was information gathering, processing, and distribution. Among other developments, was the installation of worldwide telephone networks, the invention of radio and television, the birth and unprecedented growth of the computer industry, and the launching of communication satellites (Andrew, 2007).

As a result of rapid technological progress, collection, transporting and processing of information is gradually decentralizing. Organizations with hundreds of offices spread across wide geographical areas need to monitor their various offices with just a push of a button. As our ability of gather, process and distribute information grows, the demand for more sophisticated information processing grow even faster. The old model of a single computer serving all of the organization's computational needs has been replaced by one in which a large number of separate but interconnected computers do the job. These systems are called computer networks (Andrew, 2007). The advance of computer networks and distributed system brought about the development of the internet and subsequently the new paradigm called cloud computing.

2.2 Utility Computing

Utility computing is a service provisioning model in which a service provider makes computing resources and infrastructure management available to the customer as needed, and charges them for specific usage rather than a flat rate (Anette, 2011). This utility model seeks to maximize the efficient use of resources and/or minimize associated costs. The word utility is used to make an analogy to other services, such as electrical power, that seek to meet fluctuating customer needs, and charge for the resources based on usage rather than on a flat-

rate basis. This approach, sometimes known as pay-per-use or metered services is becoming increasingly common in enterprise computing (Anette, 2011).

Utility computing is the packaging of computing resources, such as computation, storage and services, as a metered service similar to a traditional public utility (such as electricity, water, natural gas, or telephone network). This model has the advantage of a low or no initial cost to acquire computer resources; instead, computational resources are essentially rented - turning what was previously a need to purchase products (hardware, software and network bandwidth) into a service.

Utility Computing is a concept established by John McCarthy, who predicted already in the late 1960s that "computation may someday be organized as a public utility", such as it happened with electricity. This statement was a ground-breaking step but not further followed by that, as hardware and software capabilities could not meet this vision. As soon as the technology was not an issue anymore to fulfil the idea of providing computing resources as a service, utility computing has found its way to be realized in outsourcing providers and also cloud computing (Foster et al., 2008)

2.3 Distributed Computing

Distributed computing refers to the very idea of using distributed systems that are generally multiple computers connected to each other via computer networks to collaboratively process a common goal. Those computers communication can be homogeneous or heterogeneous, distributed globally or locally. According to the characteristics of localization or equality, distributed systems have different subsets, such as supercomputers, grids, clusters, web 2.0 and clouds (Foster et al., 2008).

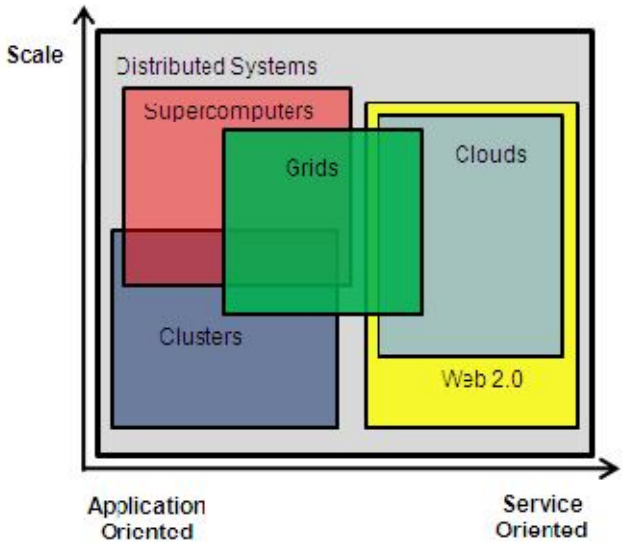


Figure 2.1: Distributed Computing and its Subsets (Foster et al., 2008).

As illustrated in figure 2-1 above, several distributed computing platforms exist which include supercomputers, grid computing, cluster computing, web 2.0 and cloud computing.

2.3.1. Cluster Computing

A cluster is a type of parallel or distributed computer system, which consists of a collection of interconnected stand-alone computers working together as a single integrated computing resource (Rajkaumar, 1999) (Gregory, 1998). Clustering these stand-alone computers together has resulted in high-performance, high-availability, and high-throughput processing on a network of computers at a much lower cost than traditional supercomputing systems, thus resulting in cluster computing being a more viable choice as a supercomputing solution (Chee et al., 2009).

The growing popularity of the Internet and the availability of high performance computers and high-speed networks as low-cost commodity components are changing the way we do computing. These technological developments have led to the possibility of using networks of computers as a single, unified computing resource, known as cluster computing

[(Rajkaumar, 1999) (Gregory, 1998). Clusters appear in various forms: high-performance clusters, high-availability clusters, dedicated clusters, non-dedicated clusters, and so on. In addition, computer scientists in the mid-1990s, inspired by the electrical power grid's pervasiveness and reliability, began exploring the design and development of a new infrastructure, computational power grids for sharing computational resources such as clusters distributed across different organisations (Ian & Carl, 2003). In the business world, cluster architecture-based large-scale computing systems, called data centres, offering high-performance and high-available hosting services are widely used. The reliable and low-cost availability of data centre services has encouraged many businesses to outsource their computing needs; thus heralding a new utility computing model (Chee et al., 2009).

Characteristics of clusters are that the computers being linked to each other are normally distributed locally, and have the same kind of hardware and operating system. Therefore cluster work stations are connected together and can possibly be used as a super computer. Currently, service-oriented Grid technologies are employed to enable utility computing environments where the majority of Grid resources are clusters. Grid schedulers such as brokers and workflow engines can then discover suitable Grid resources and submit jobs to them on the behalf of the users (Chee et al., 2009).

2.3.2. Supercomputers

Supercomputers can be easily compared to clusters, because it follows the same concept, except the fact that it is merged into one box already and is not locally interconnected with other machines. (Chee et al., 2009).

A supercomputer is a computer at the frontline of current processing capacity, particularly speed of calculation (Ian & Carl, 2003). Supercomputers are used for highly calculation-intensive tasks such as problems including quantum physics, weather forecasting, climate research, molecular modelling (computing the structures and properties of chemical

compounds, biological macromolecules, polymers, and crystals), and physical simulations (such as simulation of airplanes in wind tunnels, simulation of the detonation of nuclear weapons, and research into nuclear fusion). (Wikipedia “Supercomputers”, 2011).

A typical supercomputer consumes large amounts of electrical power, almost all of which is converted into heat, requiring cooling. For example, Tianhe-1A consumes 4.04 Megawatts of electricity (Nvidia, 2010). The cost to power and cool the system can be significant, e.g. 4MW at $0.10/KWh is $400 an hour or about $3.5 million per year (Nvidia, 2010). Thus we can say that the two main disadvantages of supercomputers are that they are usually expensive and have the necessity of a huge amount of energy.

As of October 2010 the fastest supercomputer in the world is the K computer which has over 68,000 8-core processors, while Tianhe-1A system at National University of Defence Technology comes at second number with more than 14,000 multi-core processors (Wikipedia “Supercomputers”, 2011).

In February 2009, IBM also announced work on "Sequoia," which appears to be a 20 petaflops supercomputer. This will be equivalent to 2 million laptops (whereas Roadrunner is comparable to a mere 100,000 laptops (Wikipedia “Supercomputers”, 2011). It is slated for deployment in late 2011 (Tom, 2009). The Sequoia will be powered by 1.6 million cores (specific 45-nanometer chips in development) and 1.6 petabytes of memory. It will be housed in 96 refrigerators spanning roughly 3,000 square feet (280 meter square (Petaflop Sequoia Supercomputer, 2009).

2.3.3 Grid Computing

When defining grid computing it is necessary to differ it from clusters. While clusters are distributed locally and obliged to use the same hardware and OS, grids involve heterogeneous computers that are connected to each other and distributed globally. The OS and

hardware that run on those machines can also be different from each other (Chee, 2009). The computers that are interconnected over the internet can come from anywhere while there is usually no obligation to pay. For this reason already it is obvious that grids being connected are not nearly as expensive as the supercomputers that are offered from IBM and other technology companies.

The aim of Grid computing is to enable coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations (Ian et al., 2001). An infinite number of computing devices ranging from high performance systems such as supercomputers and clusters, to specialized systems such as visualization devices, storage systems, and scientific instruments, are logically coupled together in a Grid and presented as a single unified resource to the user (Rajkumar et al., 2001). Figure 2-2, shows that a Grid user can easily use these globally distributed Grid resources by interacting with a Grid resource broker. Basically, a Grid user perceives the Grid as a single huge virtual computer that provides immense computing capabilities, identical to an Internet user who views the World Wide Web as a unified source of content.

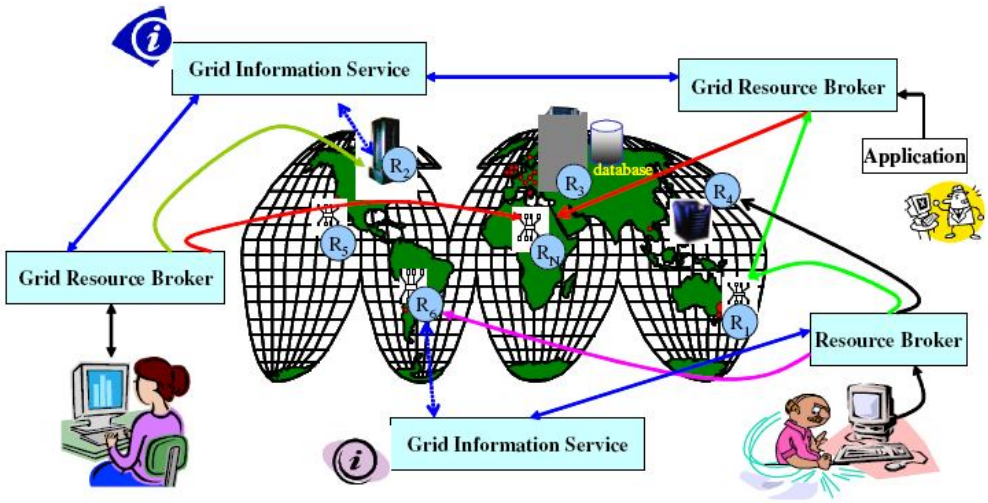


Figure 2.2: A Generic View of a Global Grid (Chee, 2009).

A diverse range of applications are currently or soon to be employed on Grids, some of which include: aircraft engine diagnostics, earthquake engineering, virtual observatory, bioinformatics, drug discovery, digital image analysis, high energy physics, astrophysics, and multi-player gaming (Ian & Carl, 2003). Grids can be primarily classified into the following types, depending on the nature of their emphasis as depicted in Figure 2-3 (Klaus et al., 2002):

- Computational Grid: Aggregates the computational power of globally distributed computers [e.g. TeraGrid (TeraGrid, 2006), ChinaGrid (ChinaGrid, 2006), and APACGrid (APACGrid, 2006)].
- Data Grid: Emphasizes on a global-scale management of data to provide data access, integration, and processing through distributed data repositories [e.g. LHCGrid (LHCGrid, 2006) and GriPhyN (GriPhyN, 2006)].
- Application Service Provisioning (ASP) Grid: Focuses on providing access to remote applications, modules, and libraries hosted on data centres or Computational Grids [e.g. NetSolve/GridSolve (NetSolve/GridSolve, 2006)].
- Interaction Grid: Focuses on interaction and collaborative visualization between participants [e.g. AccessGrid (AccessGrid, 2006)].
- Knowledge Grid: Aims towards knowledge acquisition, processing, management, and provide business analytics services driven by integrated data mining services [e.g., Italian KnowledgeGrid (Mario & Domenico, 2003) and EU Data Mining Grid (EU Data Mining Grid, 2006)].
- Utility Grid: Focuses on providing all the Grid services including compute power, data, and services to end-users as IT utilities on a subscription basis and the infrastructure necessary for negotiation of required Quality of Service (QoS), establishment and management of contracts, and allocation of resources to meet competing demands from

multiple users and applications [e.g. Gridbus (Rajkumar & Srikumar, 2004) and Utility Data Centre (Sven et al., 2003)].

These various types of Grids follow a layered design, with the Computational Grid as the bottom-most layer and the Utility Grid as the top-most layer. A Grid on a higher layer utilizes the services of Grids that operate at lower layers in the design. For example, a Data Grid utilizes the services of Computational Grid for data processing and hence builds on it. In addition, lower-layer Grids focus heavily on infrastructural aspects, whereas higher-layer ones focus on users and QoS delivery. Accordingly, Grids are proposed as the emerging cyber infrastructure to power utility computing applications (Chee, 2009).

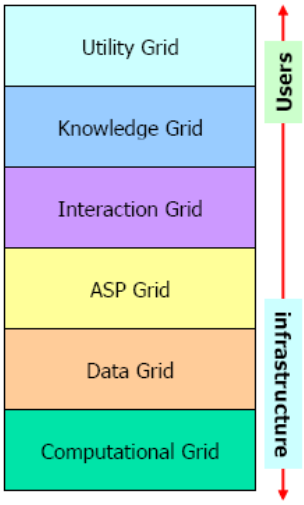


Figure 2.3: Types of Grid and their Focus (Chee, 2009).

2.4. Cloud Computing

Cloud Computing is a new paradigm in Information Technology (IT). There are several definitions of the Cloud, ranging from very broad as almost everything on the Internet to very narrow only concerning Virtualization on servers (Rattinghouse & Ransome, 2010). The basic idea is that anything that traditionally is possible in computing can be shifted to the cloud; Communication, scientific computing, word processing etc.

Clouds are a large pool of easily usable and accessible virtualized resources (such as hardware, development platforms and/or services). These resources can be dynamically reconfigured to adjust to a variable load (scale), allowing also for an optimum resource utilization (Vaquero et al., 2009).

Clouds can also be defined as computers that are networked anywhere in the world with the availability of paying the used clouds in a pay-per-use way, meaning that just the resources that are being used will be paid (Ambrust et al., 2009). Cloud computing makes sense for a large organisation with many sites and many servers as standardisation and maintenance / control of the ICT infrastructure can be simplified by pooling resources and forming a cloud based environment. Smaller companies could also benefit from joining the cloud although the success or failure of this integration basically comes down to the capabilities of your broadband connection or indeed fibre connection, if you are fortunate enough to live in an area covered by the fibre providers.

2.4.1. Cloud Models

Deploying cloud computing can differ depending on requirements, according to a Solution White paper of Dialogic Corporation the following four deployment models have been identified, each with specific characteristics that support the needs of the services and users of the clouds in a particular ways. These models include Private cloud, Community cloud, Public cloud, and Hybrid cloud (Dialogic Corporation, 2010).

a. Private Cloud

Private clouds are normally data centres that are used in a private network and can therefore restrict the unwanted public to access the data that is used by the company. It is obvious that this way has a more secure background than the traditional public clouds. However, managers still have to worry about the purchase, building and maintenance of the system (Ambrust et al., 2009). The cloud infrastructure has been

deployed, and is maintained and operated for a specific organization. The operation may be in-house or with a third party on the premises. The private cloud is also referred to as internal cloud or on-premise cloud, a private cloud intentionally limits access to its resources to service consumers that belong to the same organization that owns the cloud. In other words, the infrastructure is managed and operated for one organization only, primarily to maintain a consistent level of control over security, privacy and governance. Essential characteristics of a private cloud typically include:

- Heterogeneous infrastructure
- Customized and tailored policies
- Dedicated resources
- In-house infrastructure (capital expenditure cost model)
- End-to-end control.

b. Community Cloud

The cloud infrastructure is shared among a number of organizations with similar interests and requirements. This may help limit the capital expenditure costs for its establishment as the costs are shared among the organizations. The operation may be in-house or with a third party on the premises. This deployment model typically refers to special-purpose cloud computing environments shared and managed by a number of related organizations participating in a common domain or vertical market.

c. Public Cloud

A public cloud encompasses the traditional concept of cloud computing, having the opportunity to use computing resources from anywhere in the world. The clouds can be used in a so-called pay-per-use manner, meaning that just the resources that are being used will be paid by transaction fees (Ambrust et al., 2009). The cloud infrastructure is available to the public on a commercial basis by a cloud service provider. This enables

a consumer to develop and deploy a service in the cloud with very little financial outlay compared to the capital expenditure requirements normally associated with other deployment options. It is also known as external cloud or multi-tenant cloud, this model essentially represents a cloud environment that is openly accessible. It generally provides an IT infrastructure in a third-party physical data centre that can be utilized to deliver services without having to be concerned with the underlying technical complexities.

Essential characteristics of a public cloud typically include:

- Homogeneous infrastructure
- Common policies
- Shared resources and multi-tenant
- Leased or rented infrastructure; operational expenditure cost model
- Economies of scale

Note that public clouds can host individual services or collections of services, allow for the deployment of service compositions and even entire service inventories.

d. Hybrid Cloud

As the name already reveals, a hybrid cloud is a mixture of both a private and public cloud. This can involve work load being processed by an enterprise data centre while other activities are provided by the public cloud (Sam, 2009). The cloud infrastructure consists of a number of clouds of any type, but the clouds have the ability through their interfaces to allow data and/or applications to be moved from one cloud to another. This can be a combination of private and public clouds that support the requirement to retain some data in an organization, and also the need to offer services in the cloud.

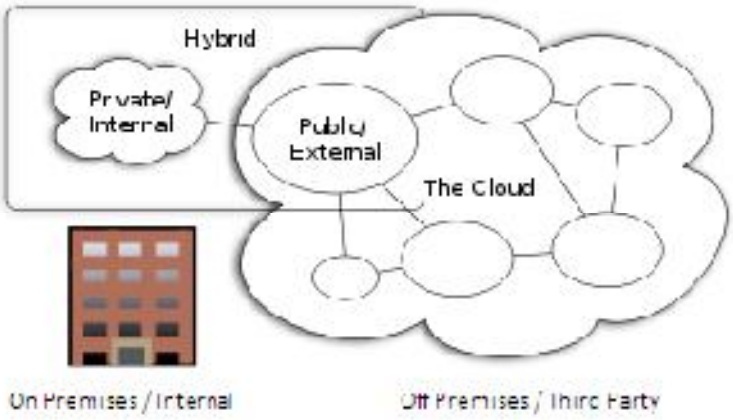


Figure2.4: Cloud Computing Types (Sam, 2009).

2.4.2 Cloud Service Providers

Listed below are some available clouds computing platforms (TechTarget, 2011) (Washington Technology, 2011):

a. **Google Apps for Business:** Google is the household name, and it has all needed to offer a reliable and secure online office tools – even before the cloud is as popular as today – Google Docs, Google Calendar, Gmail, etc. – you can access all of them in a secure and private environment with 24/7 support. The "personal" version is free, while the business version is offered at $50 per user per year.

b. **Skype:** Forget the latest historic downtime – Skype is one of the most trusted and reliable cloud-based companies offering free Internet call, with Pay As You Go and subscription-based plans to call on any phones to meet your small business needs.

c. **SalesForce:** A household name in cloud computing for customer relationship management (CRM.) Also one of the front-runners in the cloud, SalesForce is growing its arsenal of cloud apps: Sales Cloud, Force.com, Service Cloud, and the latest, Database.com. SalesForce can help your small business to manage everything related to

your sales-generating activities in a centralised "dashboard." It can also be adopted for use within system that make use of robust databases and data access levels.

d. **Basecamp:** One of the leaders in online collaboration and project management. It can help your stakeholders to discuss, update, upload/download, share – anything you can think of – in one single web account, in real time. Basecamp claims that there are over 5 million people worldwide who are using it.

e. **Quickbase:** This online database software can help your small business tech team to create online database application from scratch or use more than 200 templates. Don't have someone to build database app, yet? Intuit's Quickbase can recommend you one of 160 partners to help you out.

f. **Box.net:** A cloud storage service provider allowing you to share, manage and access files and folders online in a secured and private environment. You can also collaborate to update documents on the fly with your business team members or clients.

g. **Outright:** Outright business accounting by allowing them to track income/expenses, tax obligations, and profits/losses in real time, online. No more hiring/firing bookkeeper for business finances.

h. **Evernote:** I call Evernote a "cloud reminder" or "cloud scratchpad" if you will, simply because of what it does best – it helps you to store your ideas, notes, reminders, schedules, to-dos, audios, images, videos, etc. for you to recall and review later on.

i. **Mozy:** The industry leader for online backup of any kind of data and information – images, documents, audios, etc. You can use Mozy cloud app to back up your entire business, regardless of your business' size. With Mozy, you are not location-constrained – you can backup and access your backup remotely.

j. **SiteCloud:** Cloud hosting is great to host your small business website because it's on-demand and scalable – in real time. You are guaranteed service availability due to the

nature of the cloud – no issues regarding a sudden surge of web traffic to your site. You can switch plan anytime you want, without service interruption.

The diagram below shows three top cloud computing providers

Figure 2.5: Technology Capabilities for Cloud Computing (Naarasimhan, 2009).

2.4.3 Cloud Services

Cloud Computing encompasses different types of services. There are 3 classes of technology capabilities that are being offered as a service which include platform as a service (PaaS), software as a service (SaaS) and infrastructure as a service (IaaS) (Naarasimhan, 2009). According to the 'The Prentice Hall Service-Oriented Computing Series' from Thomas ERL, many different types of services that can be delivered in the various cloud deployment environments exists. Essentially, any IT resource or function can eventually be made available as a service (Thomas ERL, 2011). Although cloud-based ecosystems allow for a wide range of service delivery models, three have become most prominent:

a. Infrastructure-as-a-Service (IaaS)

This service delivery model represents a modern form of utility computing and outsourced managed hosting. IaaS environments manage and provision fundamental

computing resources (networking, storage, virtualized servers, etc.). This allows consumers to deploy and manage assets on leased or rented server instances, while the service providers own and govern the underlying infrastructure.

Statistically proven figures show that 80% of the computing power is not efficiently used, neither is 65% of the storage of servers. Hence there is a huge potential to share resources in order to use them in a cost efficient way rather than underutilizing them (Ambrust et al., 2009). Instead of investing in their own corporate server or network infrastructure, companies are able to purchase those resources on a rental basis and use it on demand rather than having their own resources locally. The providers are taking care of the servers, storage and network settings, while the client has virtual instances of that (Ambrust et al., 2009) (Buyya, et al., 2009)

Amazon Web Services is one example of that, where infrastructure is available on a pay-per-use self service basis and get servers, storage, network configuration, set all that up and run it, while not having to worry about co-location, rental or data enters (Amazone, 2009).

b. Platform-as-a-Service (PaaS)

The PaaS model refers to an environment that provisions application platform resources to enable direct deployment of application-level assets (code, data, configurations, policies, etc.). This type of service generally operates at a higher abstraction level so that users manage and control the assets they deploy into these environments. With this arrangement, service providers maintain and govern the application environments, server instances, as well as the underlying infrastructure. Platform as a service provides the facility to support the development lifecycle from design, implementation, debugging, testing, deployment, operation and support of rich internet applications (RIA) and online services. Here mostly the internet browser will be used for the

development (Ambrust et al., 2009) (Buyya, et al., 2009). With platform as a service an entire software environment can run at a service provider while not worrying about the technology underneath it. Just for the database and the application environment it needs to be taken care of (Ambrust et al., 2009).

Force.com is an example of platform as a service offered from Salesforce providing a development platform that makes it very easy for developers to build multi-tenant applications. The applications run on the data centres of Salesforce, so there is no necessity to take care of maintenance, security and back-ups (Phil & Chris, 2010).

c. Software-as-a-Service (SaaS)

Hosted software applications or multi-tenant application services that end-users consume directly correspond to the SaaS delivery model. Consumers typically only have control over how they use the cloud-based service, while service providers maintain and govern the software, data, and underlying infrastructure.

Companies can use software that is made available online on a rental or usage basis rather than buying the whole software package locally without being sure whether or not the investment will pay off on a long-term basis. No maintenance or updates are involved; this will all be handled by the software provider (Ambrust et al., 2009).

As an example the so-called Google Apps offer software for business or private entities online that can do the fundamental business action that a usual on-premise office suite can provide. Google Apps involve document collaboration within text documents, presentation and spreadsheets as much as calendars and e-mail services (Caroline, 2010).

d. Other Delivery Models

Cloud computing is not limited to aforementioned delivery models. Security, governance, business process management, integration, complex event processing,

information and data repository processing, collaborative processes - all can be exposed as services and consumed and utilized to create other services.

An Analogy

An on-premise infrastructure is like having your own car. You have complete control over when and where you want to drive it, but you are also responsible for its operation and maintenance. IaaS is like using a car rental service. You still have control over when and where you want to go, but you don't need to be concerned with the vehicle's maintenance. PaaS is more comparable to public transportation. It is easier to use as you don't need to know how to operate it and it costs less. However, you don't have control over its operation, schedule, or routes.

2.5 Force.com Cloud Computing Platform

As users of the Internet, we're all familiar with the fascinating, innovative, creative, and sometimes silly ways in which it has changed how we work and play. From social networking sites to wikis to blogs, and more, it’s exciting to watch the innovations taking place that are changing the ways we communicate and collaborate. While these changes have certainly impacted how we work with content, a similar set of Internet-driven ideas and technologies is changing how we build and work with applications. While yesterday's business applications required thousands, if not millions, of naira and sometimes years of professional services help to set up and customize, the technologies offered by the Internet today make it much easier to create, configure, and use applications of all kinds. Indeed, the power of the Internet has given us the ability to solve new kinds of business problems that, because of complexity or cost, had previously remained out of reach.

Just as the changes that moved publishing technology from paper to bits made it possible for us to have information about anything in the whole world right at our fingertips, the changes in application technology make it similarly possible to imagine a robust,

enterprise-class application for almost any business need. These new ways of building and running applications are enabled by the world of cloud computing, where you access applications, or apps, over the Internet as utilities, rather than as pieces of software running on your desktop or in the server room. This model is already quite common for consumer apps like email and photo sharing, and for certain business applications, like customer relationship management (CRM).

Because almost all apps these days are delivered via a Web browser, it's increasingly hard to tell which applications are “traditional software,” and which are run in the cloud. As with the Internet, applications that run in the cloud have grown so ubiquitous that almost every business user interacts with at least one, whether it's an email service, a Web conferencing application, or a sales system (Phil & Chris, 2010).

This section introduces us to the Force.com platform, salesforce.com's platform for building and running applications in the cloud.

2.5.1 The Benefits of the Force.com Platform

Two huge benefits are offered by the Force.com platform apps in general: the platform is data-centric and collaborative.

a. Data-Centric Applications

Because the platform is centred on a database, it allows you to write apps that are data-centric. A data-centric app is an application that is based on structured, consistent information such as you might find in a database or an XML file. We can find these data-centric apps everywhere, in small desktop databases like Microsoft Access or FileMaker, all the way to the huge systems running on database management systems like Oracle or MySQL. Unlike applications that are built around unstructured data, like plain text documents or HTML files, data-centric apps make it easy to control, access, and manage data.

b. Collaborative

Because the platform can be accessed by multiple users at the same time, it also allows you to write apps that are collaborative. A collaborative app is an application with data and services that are shared by multiple users in different locations. Unlike more traditional forms of software that are installed on a single machine and are hard to access from a distance, collaborative apps on the platform can be accessed from anywhere in the world with only a Web browser. This makes it easy for teams to work together on activities or post documents at the same time and share resources in real time.

In addition to easy access over a Web browser, a number of built-in platform features also facilitate productive group collaboration:

- The platform's security and sharing model allows you to finely control a user's access to different data.
- Workflow rules allow you to automatically assign tasks, update data, or send email alerts when certain business events occur, such as the creation of a new record or a change in the value of a record field
- Approval processes allow you to set up a sequence of steps necessary for a record to be approved, including who must approve it at each step
- Collectively, these features provide a framework for sharing apps across groups, divisions, and entire corporations without relinquishing administrative control over sensitive data.

2.5.2 The Basics of an Application on the Salesforce Platform

The basics application interfaces on a salesforce platform includes the tabs, forms and links as illustrated in figure 2-4.

a. Tabs

As you can see when you start clicking around, there are a few key elements that form the foundation of the Sales Automation app and of most applications created with the platform. First, across the top of the app is a set of tabs that segment the app into different parts. Each tab corresponds to a type of object, such as an account or contact, and within a tab you can perform actions on particular records of that tab's type. You can also edit existing accounts, or use a list view to filter lists of accounts by certain criteria. Most app development work revolves around creating tabs and defining the data and behaviours that support them.

b. Forms

A second key element is the form that is displayed as part of a tab. As in any business app, forms are the primary means of entering and viewing information in the system. Forms allow you to view and edit the data associated with a particular record on a tab. When developing a new app you can define what information appears in each form, and how it is organized. For example, the form for a contact record includes fields such as Last Name, Home Phone, Mailing City, Title, Birth date, Reports To, and Account. In a Force.com platform app, the form used to enter information is referred to as an edit page and the read-only view of that information is referred to as a detail page (Phil & Chris, 2010).

c. Links

Finally, because Force.com platform apps are delivered via a Web browser, they use links to provide navigation to related data. For example, on an account detail page, there are links to related records, such as the contacts that belong to the account and the sales user who manages the account. Other links take you to recently visited

records and to areas of the app where users can set personal preferences. These links provide navigation both within an app and out into the Web.

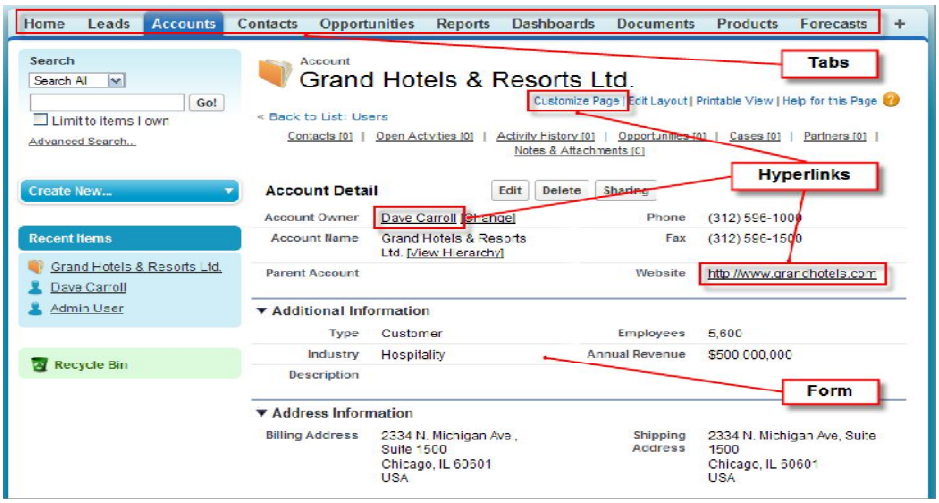


Figure 2.6: Sample Salesforce Application Platform Showing Tabs, Forms and Links.

2.5.3 The Technologies Behind the Force.com Platform Apps

It is important to review some of the technologies behind the platform itself. These technologies have a big impact on what the platform supports, and what it's like to develop on it.

Table 2.1: key Technologies Behind the Force.com Platform (Phil & Chris, 2010).

Technology	Description
Multitenant architecture	An application model in which all users and apps share a single, common infrastructure and code base
Metadata-driven development model	An app development model that allows apps to be defined as declarative "blueprints," with no code required. Data models, objects, forms, workflows, and more are defined by metadata.
Web services API	An application programming interface that defines a Web service that provides direct access to all data stored in the Force.com platform from virtually any programming language and platform
Apex	The world's first on-demand programming language, which runs in the cloud on the Force.com platform servers
Visualforce	A framework for creating feature-rich user interfaces for apps in the cloud
Sites	Public websites and applications that are directly integrated with your Salesforce.com organization—without requiring users to log in with a username and password
AppExchange directory	A Web directory where hundreds of Force.com apps are available to Salesforce.com customers to review, demo, comment upon, and/or install. Developers can submit their apps for listing on the AppExchange directory if they want to share them with the community.

a. A Multitenant Architecture

The platform's multitenant architecture means that all users share the same physical instance and version of any application that runs on it. In contrast to their single-tenant counterparts, such as client-server enterprise applications or email servers, multitenant applications are designed so that any upgrades to the platform or the apps it supports happen automatically for all users at once. Consequently, no one has to worry about buying and maintaining their own physical stack of hardware and software, or making sure that their applications always have the latest patch installed. Besides the Force.com platform, several popular, consumer-based applications also use a multitenant

architecture, including eBay, My Yahoo!, and Google Mail. Multitenant architecture allows these applications to be low cost, quick to deploy, and open to rapid innovation—exactly the qualities for which salesforce.com have also become known.

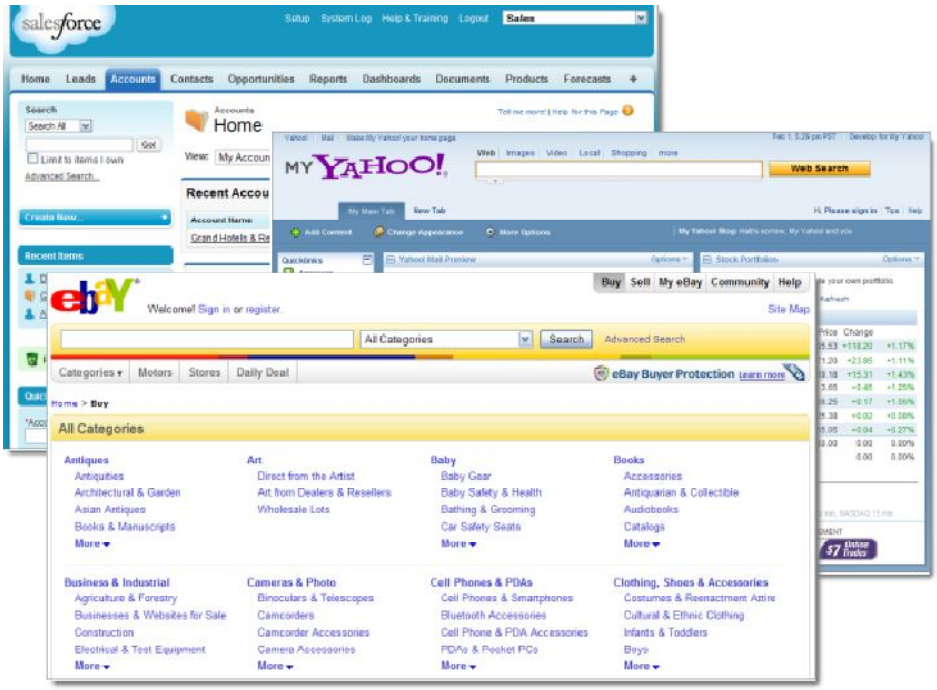


Figure 2.7: On Demand, Multitenant Applications that run on the Cloud (Phil & Chris, 2010)..

The platform's multitenant architecture also impacts how developers use the platform to create new applications. Specifically, it defines a clear boundary between the platform and the applications that run on it. A boundary is important because it allows applications to define their own components without jeopardizing the functionality of the core platform or the data stored by other users.

b. A Metadata-Driven Development Model

The Force.com platform also uses a metadata-driven development model to help app developers become more productive in putting together apps. It means that the basic

functionality of an app—that is, the tabs, forms, and links—are defined as metadata in a database rather than being hard-coded in a programming language. When a user accesses an app through the Force.com platform, it renders the app's metadata into the interface the user experiences. As a result of metadata-driven development, the Force.com platform app developers work at a much higher level of abstraction than if they developed applications using Java or C#, and are shielded from having to worry about low-level system details that the platform handles automatically. At the same time, Force.com platform developers can also leverage advanced features that the platform provides by default. The Force.com platform vastly simplifies the work of building an app and increases a developer's overall productivity. And, like Web pages that use JavaScript or Flash to add functionality to HTML pages, the Force.com platform also provides ways for more advanced developers to add custom functionality to the apps you build.

c. We Service API

The platform's metadata-driven development model allows app developers to quickly build a lot of functionality with tools provided by the platform; however, sometimes app developers want to modify the actual data in an app, and use third–party services to create more customized app behaviours. To do this, they can use the Web services API. The API provides a straightforward, powerful, and open way to programmatically access the data and capabilities of any app running on the platform. It allows programmers to access and manipulate apps from any server location, using any programming language that supports Web services, like Java, PHP, C#, or .NET. Because Web services are, not surprisingly, based on Web standards, they're well suited to traverse firewalls and leverage the rest of the Internet infrastructure already in place.

d. Apex

As you might expect from the company that delivered the world's first cloud computing platform, salesforce.com also introduced the world's first cloud computing programming language, Apex. Apex is based on Java, the most popular programming language for Web apps, and runs on the Force.com platform servers. Apex is specifically designed for building business applications to manage data and processes within the larger context of the Force.com platform. The language provides a uniquely powerful and productive approach to creating functionality and logic, allowing developers to focus just on the elements specific to their application, while leaving the rest of the "plumbing" to the Force.com platform.

e. Visualforce

At the front of any great application is a great user interface that's easy to use, powerful, and suited exactly for the tasks, users, and devices the application serves. Visualforce is a complete framework for creating such user interfaces, enabling any kind of interface design and interaction to be built and delivered entirely in the cloud. The user interfaces you build with Visualforce can extend the standard Force.com platform look and feel, or replace it with a completely unique style and set of sophisticated interactions. Because Visualforce markup is ultimately rendered into HTML, designers can use Visualforce tags alongside standard HTML, JavaScript, Flash, or any other code that can execute within an HTML page on the platform. You can also use Visualforce pages to combine data from multiple Force.com platform objects, or blend data from Web services into your applications.

f. Sites

The apps you build on the Force.com platform might contain data and functionality that you want to make accessible to people who are not Salesforce.com users.While it is

possible to use the Web services API to integrate an external Web page or application with Salesforce.com, the Force.com platform provides an easier, more efficient way of sharing data and functionality with people outside of your organization: Sites. Sites enables you to create public websites and applications that are directly integrated with your Salesforce.com organization—without requiring users to log in with a username and password. You can publicly expose any information stored in your organization through pages that match the look and feel of your company's brand. Because these sites are built and hosted on the Force.com platform servers, there are no data integration issues. And because sites are built with Visualforce pages on the platform, data validation on collected information is performed automatically.

g. The AppExchange Dictionary

The AppExchange is a Web directory where apps built on the Force.com platform are available to salesforce.com customers to browse, demo, review, and install. Developers can submit their apps for listing on the AppExchange directory if they want to share them with the community. There you'll see the hundreds of innovative and exciting apps that exist today, including everything from payroll management to telephony integration, service and support surveys, adoption dashboards, and beyond. Some of these apps have been created in-house at salesforce.com, but most are built by partners and individual developers who have chosen to take advantage of the Force.com platform.

2.6 Resource Management

Resource management is the process of using an organization's resources in the most efficient way possible. These resources include goods, equipment, financial resources and labour resources such as employees (Wikipedia “Resource Management, 2011). Resource management can include ideas such as making sure that an organization has enough physical

resources for its operation, but not in an overabundance, and making sure that people are assigned to tasks that will keep them busy and not have too much downtime.

Within the tertiary institutions, the key resources include but not limited to the following: human resources (staff and student, contractor, consultants etc), literatures (books, journals, publications, magazines, pamphlets etc), equipment (computers and its accessories etc), vehicles, fuel/gas and lots more. How these resources are managed will determine the efficient running of the institution. This paper seeks to develop a platform that will manage these resources and provide for maximization of the available resources using cloud computing technology.

2.6.1 Human Resource Management (HRM)

Human resource management is crucial to the success of any organization. Among all the resources available to an organization, the most sensitive and challenging to manage is the human resources. The human resources within a tertiary institution include the staff and student. The staff include the academic, non academic, and contract staffs. Over the years most of the key human resource management efforts has been between in companies and banks leaving out higher institutions. In recent times managing the human resources of tertiary institution has become very important as attention towards education has increased drastically especially in third world countries like Nigeria.

Since the concept of human resource management became popular in the 1980s there have been numerous studies which has tried to measure developments towards its prescriptions. A particular interesting case for a study of HRM in institutional constraints is Germany. Arguably, over the last decade German companies have been among the most successful in the world. If, as the HRM literature claims, the management of human resource is crucial for the business success, then it should at least partly explains the success of the German economy (Wikipedia “Human Resource Management, 2011). Guest suggested that considerable attention

must be given to recruitment and induction in order to ensure a high quality of staff. If selection policies aim to select highly committed and flexible people, they can help to achieve the HR outcomes of commitment and flexibility (Wikipedia “The role of Resource Management, 2011).

2.7 Database Systems and Data Models

2.7.1 What is a Database?

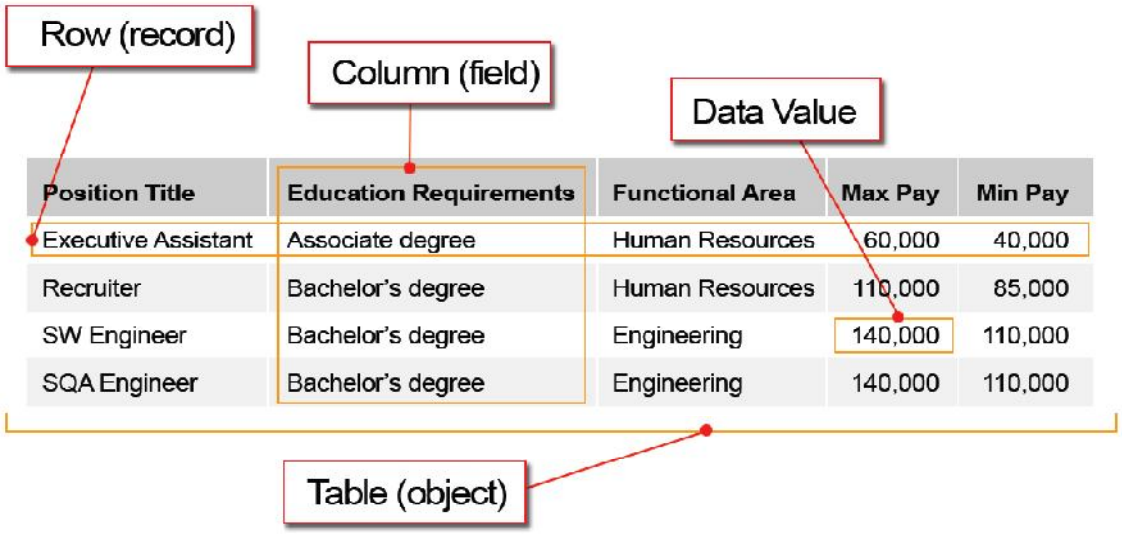
At its simplest, a database is a collection of data that is usually related in some fashion (Robert & Geoff, 2005). For instance, a database that a bookstore uses might contain information about authors, book titles, and publishers. Yet a database is more than simply a collection of related data. The data must be organized and classified in a structured format that is described by metadata, which is data that describes the data being stored. In other words, the metadata defines how the data is stored within the database. Together, the data and the metadata provide an environment that logically organizes the data in a way that can be efficiently maintained and accessed.

Over the years, as the demands for information have grown, so too have the database systems that have attempted to meet these demands. However, along with this evolution, we have seen an increase in the costs associated with storing data as well as an increase in the demand for products that can run on multiple platforms and can be optimized based on the needs of specific types of organizations.

Typically, you use a database to collect information about people, things, or concepts that are important to you and whatever project you're working on. In standard database language, the category of person, thing, or concept you want to store information about is referred to as an entity, although in standard Force.com platform terminology, we refer to this as an object. In a database, each entity is represented by a table. A database table is simply a list of information, presented with rows and columns, about the category of person, thing, or

concept you want to track. So in a phone book, you might have a table to store information about residences and another table to store information about businesses; or in a library catalog, you might have one table to store information about books and another to store information about authors.

Each row in the table represents the information about a specific instance of the object as illustrated in fig 2-6, for example, the Recruiter position or the SW Engineer position. In standard Force.com platform terminology, we call this a record. For every object you want to track in your app, you'll have multiple records to represent each individual item about which you're storing information (Phil & Chris, 2010).



Position Title	Education Requirements	Functional Area	Max Pay	Min Pay
Executive Assistant	Associate degree	Human Resources	60,000	40,000
Recruiter	Bachelor's degree	Human Resources	110,000	85,000
SW Engineer	Bachelor's degree	Engineering	140,000	110,000
SQA Engineer	Bachelor's degree	Engineering	140,000	110,000

Figure 2.8: Illustration of a Database Table

In very simplistic terms, a Force.com platform object is similar to a database table in that you'll have a separate object for each person, thing, or concept about which you want to collect information. In reality, a Force.com platform object is much more than this because the full functionality of the platform is behind each object. Each object automatically has built-in features like a user interface, a security and sharing model, workflow processes, and much more that you'll learn about in the rest of this book (Phil & Chris, 2010).

2.7.2 What is a Relational Database?

A relational database management system (RDBMS) is a database management system in which data is stored in tables and the relationships among the data are also stored in tables. The data can be accessed or reassembled in many different ways without having to change the table forms (Wikipedia “RDBMS, 2011). An RDBMS is a program that lets you create, update, and administer a relational database. Most commercial RDBMS's use the Structured Query Language (SQL) to access the database, although SQL was invented after the development of the relational model and is not necessary for its use. The leading RDBMS products are Oracle, IBM's DB2 and Microsoft's SQL Server. Despite repeated challenges by competing technologies, as well as the claim by some experts that no current RDBMS has fully implemented relational principles, the majority of new corporate databases are still being created and managed with an RDBMS (RDBMS, 2011).

2.7.3 Data Models

A data model is a conceptual representation of the data structures that are required by a database. The data structures include the data objects, the associations between data objects, and the rules which govern operations on the objects. As the name implies, the data model focuses on what data is required and how it should be organized rather than what operations will be performed on the data. To use a common analogy, the data model is equivalent to an architect's building plans.

Traditionally, database designers relied on good judgement to help them develop a good design. Unfortunately, good judgement is often in the eye of the beholder, and it is often developed after much trail and error (Peter & Carlos, 2007). A models main function is to help you understand the complexities of the real-world environment. Within the database

environment, a data model represents data structures and their characteristics, relations, constraints, and transformations (Kroenke, 2009).

2.7.4 The Evolution of Data Models

The quest for better management has lead to several different models that attempt to resolve the file system's critical shortcomings. Table 2-2 gives an overview of the major data models in roughly chronological order.

Table 2.2: Evolution of database models (Peter & Carlos, 2007)

Generation	Time	Mode	Examples	Comments
First	1960s-1970s	File system	VMS/VSAM	Used mainly on IBM mainframe e systems Managed records, not relationships
Second	1970s	Hierarchical and network	IMS ADABAS IDS-II	Early database systems Navigational access
Third	Mid-1970s to present	Relational	DB2 Oracle MS SQL-Server	Conceptual simplicity Entity relationship (ER) modelling support for relational data modelling
Forth	Mid-1980s to present	Object-oriented Extended Relational	Versant VFS/FastObjects Objectivity/DB DB/2 UDB Oracle 10g	Support complex data Extended relational products support object and data warehousing Web databases become common
Next generation	Present to future	XML	dbXML Tamino DB2 UDB Oracle 10g MS SQL Server	Organization and management of unstructured data Relational and object models add support to XML documents

The Development of Data Models can be Illustrated in figure 2.7 below.

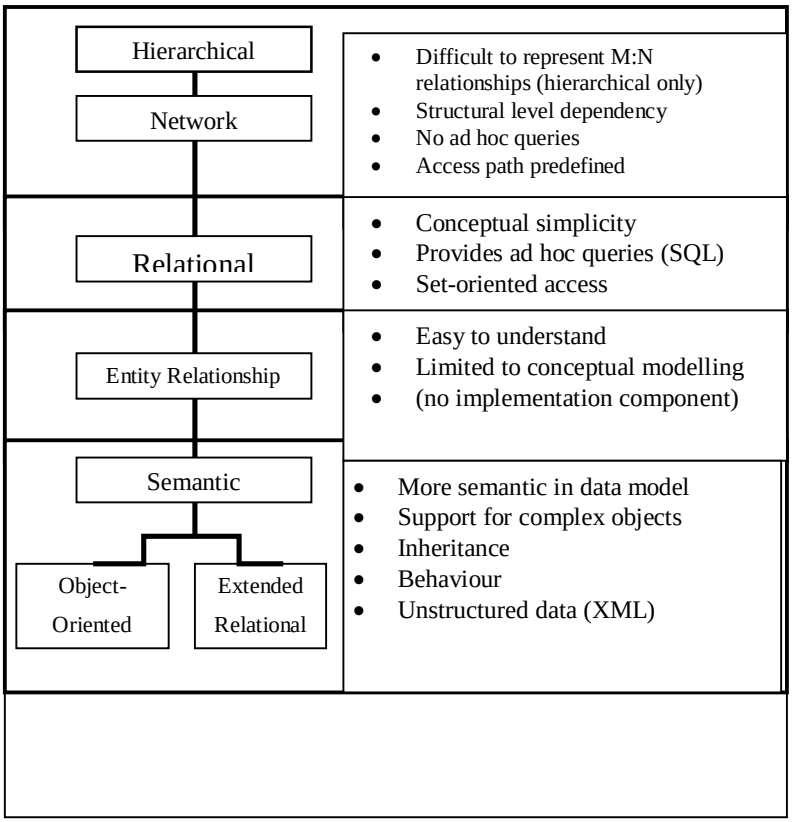


Figure 2.9: The Development of Data Models.

2.7.5 The Entity Relationship Model and Relational Database Design

The conceptual simplicity of relational database technology triggered the demand for RDBMSs In turn, the rapidly increasing transaction and information requirements created the need for more complex database implementation structure, thus creating the need for more effective database design tools (Peter & Carlos, 2007).

Data modelling as it relates to RDBMS is the process of creating a logical representation of the structure of a database (Kroenke, 2009). It is the most important task performed by the developer – the entire database and all applications that access this database are based on ti, and if the data is modelled incorrectly, poor and difficult to user applications will result. A data model is a model of the user's model of their business or organization. It

must identity the things to be store in the database and define their structure and the relationships among them (Kroenke, 2009). Eventually the data model leads to the development and implementation of a database. Whether or not this database will meet, and possibly exceed, the user's requirements depends on how closely the data model resembles the user's model of their business.

Data models fall into tow main categories; conceptual models and implementation models Conceptual models are concerned with the logical nature of the data and what is being represented, whereas implementation models are concerned with the physical nature of the database and with how the data will be represented. Two very important models used in database design are the Entity-Relationship (E-R) model and the Relational model. In the E-R model, data is represented using entities, and relationships are defined between these entities. However, with the relational model, the entities and their relationships follow strict guidelines. Usually, an E-R model is first developed, and then it is transformed into a relational model (Deriving ERD, 2011).

In software engineering, an entity-relationship model (ERM) is an abstract and conceptual representation of data (Wikipedia "Entity relationship mode", 2011)Entity-relationship modelling is a database modelling method, used to produce a type of conceptual schema or semantic data model of a system, often a relational database, and its requirements in a top-down fashion. Diagrams created by this process are called entity-relationship diagrams (ER diagrams, or ERDs).

Peter Chen first introduced the ER data model in 1976; it was the graphical representation of entities and their relationships in a database structure that quickly became popular because it completed the relational data model concepts. The ER model is based on entities, attributes and relationships (Peter & Carlos, 2007)

2.8 Data Communications

When we communicate, we are sharing information (Bohrouz, 2007). This sharing can be local or remote. Between individuals, local communication usually occurs face to face, while remote communication takes place over distance. The term telecommunication, which includes telephony, telegraphy, and television, means communication at a distance (tele is Greek for "far") (Bohrouz, 2007).

The word data refers to information presented in whatever form is agreed upon by the parties creating and using the data. Data communications are the exchange of data between two devices via some form of transmission medium such as a wire cable. For data communications to occur, the communicating devices must be part of a communication system made up of a combination of hardware (physical equipment) and software (programs) (Bohrouz, 2007).

The effectiveness of a data communications system depends on four fundamental characteristics: delivery, accuracy, timeliness, and jitter.

a. Delivery. The system must deliver data to the correct destination. Data must be received by the intended device or user and only by that device or user.
b. Accuracy. The system must deliver the data accurately. Data that have been altered in transmission and left uncorrected are unusable.
c. Timeliness. The system must deliver data in a timely manner. Data delivered late are useless. In the case of video and audio, timely delivery means delivering data as they are produced, in the same order that they are produced, and without significant delay. This kind of delivery is called real-time transmission.
d. Jitter. Jitter refers to the variation in the packet arrival time. It is the uneven delay in the delivery of audio or video packets. For example, let us assume that video packets are sent every 3D ms. If some of the packets arrive with 3D-ms delay and others with 4D-ms delay, an uneven quality in the video is the result.

2.8.1 Components of Data Communication

The five major components of a data communication system are a Message, a Sender (transmitter), a Receiver, a Transmission Medium and Protocol (Components of Data Communication, 2011). The message is the information to be communicated. This can be anything from text and pictures to audio and video. A Sender is a device which sends the data messages. This can be anything from a computer or laptop, to a workstation or a mobile phone. A Receiver is pretty much the same, except it is receiving the information. Generally receivers can also send, but sometimes they clash in terms of compatibility due to differences in technology.

A Transmission Medium is the physical path by which a message travels from sender to receiver. This can be by wire, or wireless. It can be by coaxial cable, fibre optics, or by radio waves. The Protocol is the final piece. These are the rules that govern the data communications and transfers. It represents an agreement between the communicating devices, i.e. the agreement to allow the sender access to deliver to the receiver unchallenged. Text messages transfer unchallenged, but a Bluetooth connection requires confirmation. Devices can be connected but no communication if a protocol is not confirmed. An example of this is connecting a USB device. Most devices now are Plug and Play, meaning the computer confirms protocol and opens a gateway to the device in order to allow file transfer. However, in devices that aren't Plug and Play, these need software installed in order to install protocol and validate it upon connection. Simply put, the general order of the relationship is that the Sender (or Transmitter) sends the Message via the Transmission Medium to the Receiver; all the while adhering to Protocol confirmed between the Sender and Receiver prior to the communication (Components of Data Communication, 2011). Figure 2.10 below shows the five components of a data communication system.

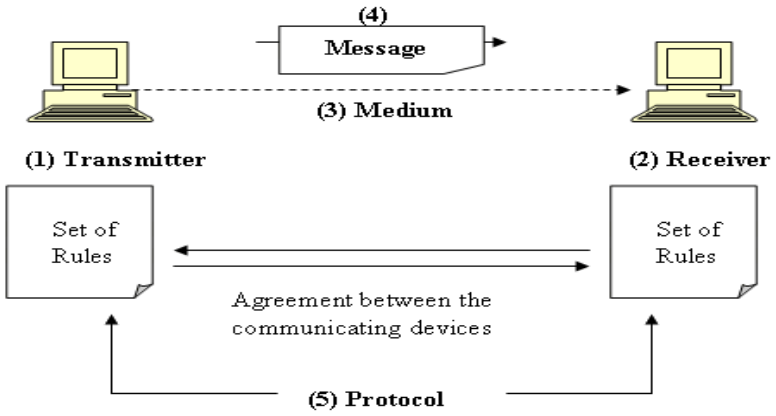


Figure 2.10: The five components of a data communication system (The five Components of Data Communication, 2011).

2.8.2 Data Flow

Communication between two devices can be simplex, half-duplex, or full-duplex as shown in Figure 2-9.

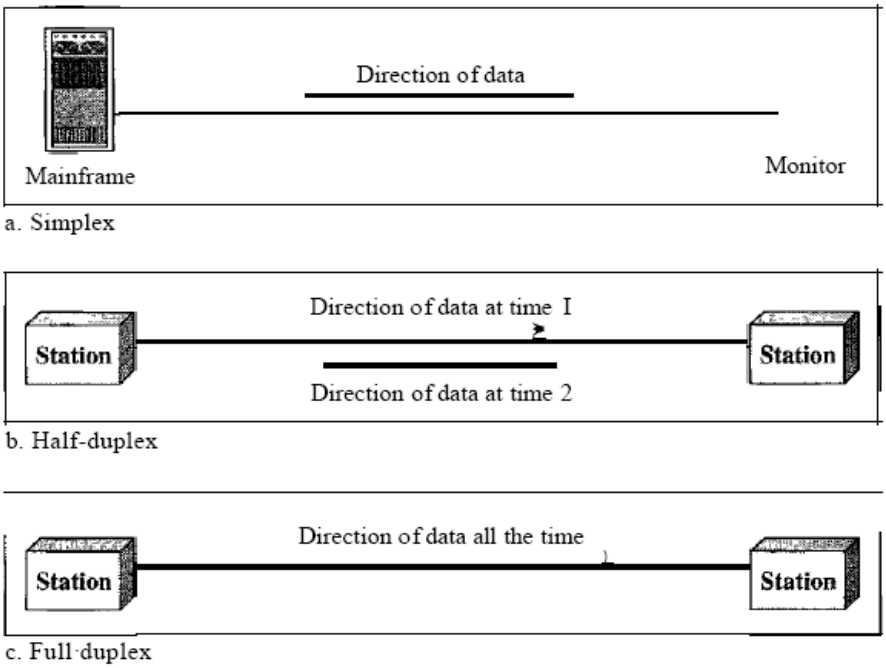


Fig 2-11: Data flow (simplex, half-duplex, and full-duplex) (Bohrouz, 2007)

a. Simplex

In simplex mode, the communication is unidirectional, as on a one-way street. Only one of the two devices on a link can transmit; the other can only receive (see Figure 2-9a). Keyboards and traditional monitors are examples of simplex devices. The keyboard can only introduce input; the monitor can only accept output. The simplex mode can use the entire capacity of the channel to send data in one direction.

b. Half-Duplex

In half-duplex mode, each station can both transmit and receive, but not at the same time. When one device is sending, the other can only receive, and vice versa (see Figure 2-9b). The half-duplex mode is like a one-lane road with traffic allowed in both directions. When cars are travelling in one direction, cars going the other way must wait. In a half-duplex transmission, the entire capacity of a channel is taken over by whichever of the two devices is transmitting at the time. Walkie-talkies and CB (citizens band) radios are both half-duplex systems. The half-duplex mode is used in cases where there is no need for communication in both directions at the same time; the entire capacity of the channel can be utilized for each direction.

c. Full-Duplex

In full-duplex, both stations can transmit and receive simultaneously (see Figure 2-9c). The full-duplex mode is like a two-way street with traffic flowing in both directions at the same time. In full-duplex mode, signals going in one direction share the capacity of the link: with signals going in the other direction. This sharing can occur in two ways: Either the link must contain two physically separate transmission paths, one for sending and the other for receiving; or the capacity of the channel is divided between signals travelling in both directions. One common example of full-duplex communication is the telephone network. When two people are communicating by a telephone line, both can

talk and listen at the same time. The full-duplex mode is used when communication in both directions is required all the time. The capacity of the channel, however, must be divided between the two directions.

2.9 Networks and Internetworking

A network is a set of devices (often referred to as nodes) connected by communication links (Behrouz, 2007). A node can be a computer, printer, or any other device capable of sending and/or receiving data generated by other nodes on the network.

Internetworking on the other hand, is the practice of connecting a computer network with other networks through the use of gateways that provide a common method of routing information packets between the networks. The resulting system of interconnected networks is called an internetwork, or simply an internet (Wikipedia "Internetworking", 2011). The most notable example of internetworking is the Internet, a network of networks based on many underlying hardware technologies, but unified by an internetworking protocol standard, the Internet Protocol Suite, often also referred to as TCP/IP.

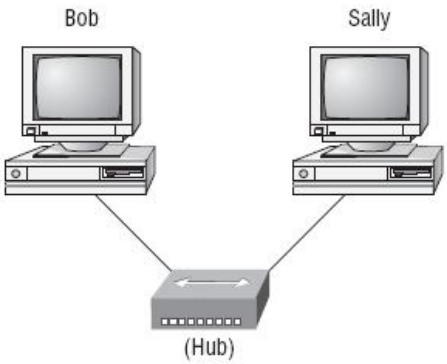


Figure 2.12: Diagram of a Basic Network

2.9.1 The Internet

A network is a group of connected communicating devices such as computers and printers. An internet (note the lowercase letter i) is two or more networks that can communicate

with each other. The most notable internet is called the Internet (uppercase letter I), a collaboration of more than hundreds of thousands of interconnected networks (Behrouz, 2007). Private individuals as well as various organizations such as government agencies, schools, research facilities, corporations, and libraries in more than 100 countries use the Internet. Millions of people are users. Yet this extraordinary communication system only came into being in 1969 (Behrouz, 2007).

In the mid-1960s, mainframe computers in research organizations were standalone devices. Computers from different manufacturers were unable to communicate with one another. The Advanced Research Projects Agency (ARPA) in the Department of Defense (DoD) was interested in finding a way to connect computers so that the researchers they funded could share their findings, thereby reducing costs and eliminating duplication of effort (Behrouz, 2007).

In 1967, at an Association for Computing Machinery (ACM) meeting, ARPA presented its ideas for ARPANET, a small network of connected computers. The idea was that each host computer (not necessarily from the same manufacturer) would be attached to a specialized computer, called an interface message processor (IMP). The IMPs, in turn, would be connected to one another. Each IMP had to be able to communicate with other IMPs as well as with its own attached host (Behrouz, 2007).

By 1969, ARPANET was a reality. Four nodes, at the University of California at Los Angeles (UCLA), the University of California at Santa Barbara (UCSB), Stanford Research Institute (SRI), and the University of Utah, were connected via the IMPs to form a network. Software called the Network Control Protocol (NCP) provided communication between the hosts (Behrouz, 2007).

In 1972, Vint Cerf and Bob Kahn, both of whom were part of the core ARPANET group, collaborated on what they called the Internetting Projec1. Cerf and Kahn's landmark

1973 paper outlined the protocols to achieve end-to-end delivery of packets. This paper on Transmission Control Protocol (TCP) included concepts such as encapsulation, the datagram, and the functions of a gateway. Shortly thereafter, authorities made a decision to split TCP into two protocols: Transmission Control Protocol (TCP) and Internetworking Protocol (lP). IP would handle datagram routing while TCP would be responsible for higher-level functions such as segmentation, reassembly, and error detection. The internetworking protocol became known as TCPIIP.

The Internet has come a long way since the 1960s. The Internet today is not a simple hierarchical structure. It is made up of many wide- and local-area networks joined by connecting devices and switching stations. It is difficult to give an accurate representation of the Internet because it is continually changing-new networks are being added, existing networks are adding addresses, and networks of defunct companies are being removed. Today most end users who want Internet connection use the services of Internet service providers (lSPs). There are international service providers, national service providers, regional service providers, and local service providers. The Internet today is run by private companies, not the government. Figure 2.13 shows a conceptual (not geographic) view of the Internet.

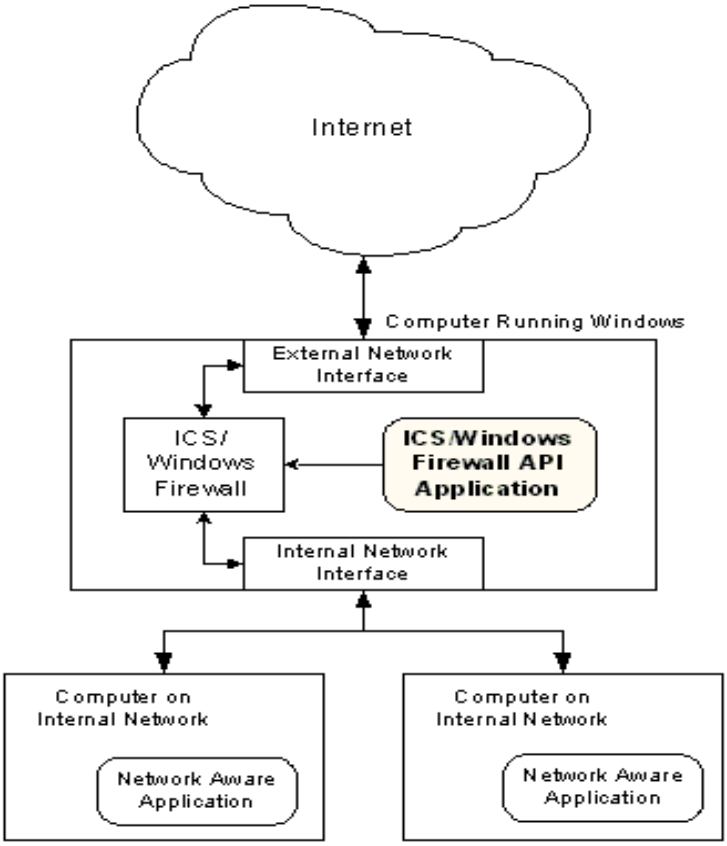


Figure 2.13: A Typical Diagram of the Internet

2.9.2 Network Criteria

A network must be able to meet a certain number of criteria. The most important of these are performance, reliability, and security (Behrouz, 2007).

a. Performance

Performance can be measured in many ways, including transit time and response time. Transit time is the amount of time required for a message to travel from one device to another. Response time is the elapsed time between an inquiry and a response. The performance of a network depends on a number of factors, including the number of users, the type of transmission medium, the capabilities of the connected hardware, and the efficiency of the software. Performance is often evaluated by two networking metrics: throughput and delay. We often need more throughputs and less delay.

However, these two criteria are often contradictory. If we try to send more data to the network, we may increase throughput but we increase the delay because of traffic congestion in the network.

b. Reliability

In addition to accuracy of delivery, network reliability is measured by the frequency of failure, the time it takes a link to recover from a failure, and the network's robustness in a catastrophe.

c. Security

Network security issues include protecting data from unauthorized access, protecting data from damage and development, and implementing policies and procedures for recovery from breaches and data losses

2.9.3 Network Reference Models

Computer networks are created by different entities. Standards are needed so that these heterogeneous networks can communicate with one another. The two best-known standards are the OSI model and the Internet model [64]. The OSI (Open Systems Interconnection) model defines a seven-layer network; the Internet model defines a five-layer network (Behrouz, 2007).

A reference model is a conceptual blueprint of how communications should take place. It addresses all the processes required for effective communication and divides these processes into logical groupings called layers. When a communication system is designed in this manner, it’s known as layered architecture (Todd, 2007).

A. The Open System Interconnection (OSI) Model

Established in 1947, the International Standards Organization (ISO) is a multinational body dedicated to worldwide agreement on international standards (Behrouz, 2007). An ISO standard that covers all aspects of network communications is the Open Systems

Interconnection model. It was first introduced in the late 1970s (Behrouz, 2007). An open system is a set of protocols that allows any two different systems to communicate regardless of their underlying architecture. The purpose of the OSI model is to show how to facilitate communication between different systems without requiring changes to the logic of the underlying hardware and software. The OSI model is not a protocol; it is a model for understanding and designing a network architecture that is flexible, robust, and interoperable (Behrouz, 2007). Table 2-3 show the OSI seven layers and their functions

Table 2.3: The OSI Seven Layers (InfoCellar, 2011)

LAYER	FUNCTION
Application (Layer 7)	This layer supports application and end-user processes. Communication partners are identified, quality of service is identified, user authentication and privacy are considered, and any constraints on data syntax are identified. Everything at this layer is application-specific. This layer provides application services for file transfers, e-mail, and other network software services. Telnet and FTP are applications that exist entirely in the application level. Tiered application architectures are part of this layer.
Presentation (Layer 6)	This layer provides independence from differences in data representation (e.g., encryption) by translating from application to network format, and vice versa. The presentation layer works to transform data into the form that the application layer can accept. This layer formats and encrypts data to be sent across a network, providing freedom from compatibility problems. It is sometimes called the syntax layer.
Session (Layer 5)	This layer establishes, manages and terminates connections between applications. The session layer sets up, coordinates, and terminates conversations, exchanges, and dialogues between the applications at each end. It deals with session and connection coordination.
Transport (Layer 4)	Usually TCP (the top half of TCP/IP). This layer provides transparent transfer of data between end systems, or hosts, and is responsible for end-to-end error recovery and flow control. It ensures complete data transfer.
Network (Layer 3)	Typically IP (the bottom half of TCP/IP). This layer provides switching and routing technologies, creating logical paths, known as virtual circuits, for transmitting data from node to node. Routing and forwarding are functions of this layer, as well as addressing, internetworking, error handling, congestion control and packet sequencing.
Data Link	Ethernet, ATM, Frame Relay, etc. At this layer, data packets are

(Layer 2)	encoded and decoded into bits. It furnishes transmission protocol knowledge and management and handles errors in the physical layer, flow control and frame synchronization. The data link layer is divided into two sublayers: The Media Access Control (MAC) layer and the Logical Link Control (LLC) layer. The MAC sublayer controls how a computer on the network gains access to the data and permission to transmit it. The LLC layer controls frame synchronization, flow control and error checking.
Physical (Layer 1)	This layer conveys the bit stream - electrical impulse, light or radio signal -- through the network at the electrical and mechanical level. It provides the hardware means of sending and receiving data on a carrier, including defining cables, cards and physical aspects. Fast Ethernet, RS232, and ATM are protocols with physical layer components.

Let's consider how these corresponding layers communicate using protocols (see figure 2-12). First, recall that every layer in the model, except the bottom (physical) layer, is really a program or algorithm running on a computer. There is no way for, say, a Web browser and a Web server to actually connect together directly. Instead, the software running at various layers communicates logically. That is to say, through the use of software and procedures, a process running at layer 5 on one machine can accomplish logical communication with a similar process running at layer 5 on another machine. Since machines are only physically connected at layer 1, this means that in order for a protocol at layer 5 to function, the data on the sending machine must “pass down” the data through the layers between layer 5 and layer 1. The data is then transmitted over the physical connection to layer 1 of the other machine, and “passed up” the protocol stack of the receiving machine to layer 5. This is how the two machines are logically linked at layer 5, even though they have no physical connection at that layer.

Thus, with the exception of the actual physical connection at layer 1, all horizontal communication also requires vertical communication—down the stack on one machine, and then back up the stack on the other. The communication doesn’t always

go all the way back up the stack for each connection, however, as in the case of routing.

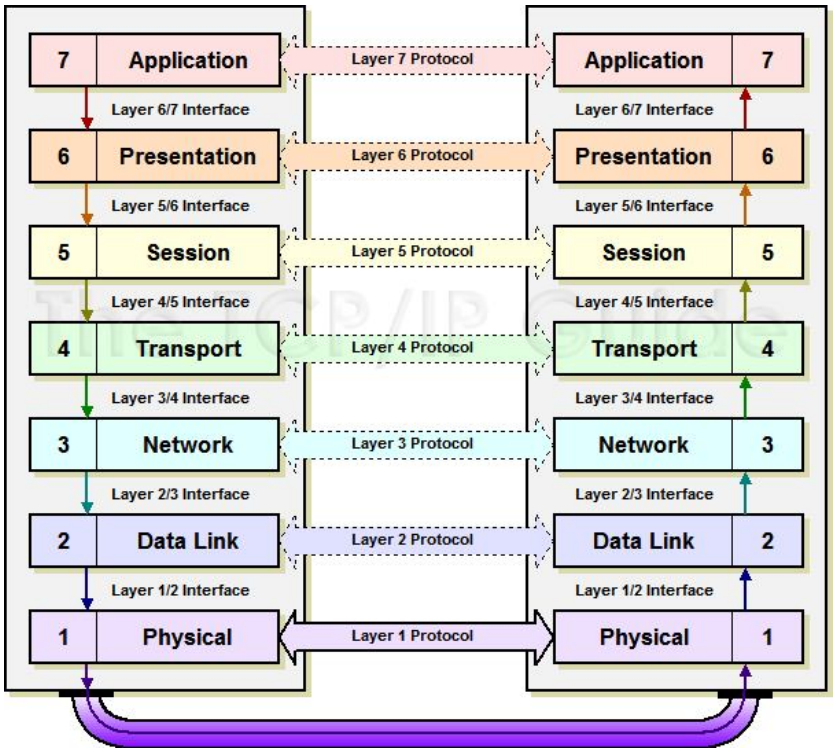


Figure 2.14: OSI Model Communication Between two Stations

B. The (TCP/IP) Protocol Suite

The transmission control protocol/internet protocol (TCP/IP) suite was developed prior to the OSI model. Therefore, the layers in the TCP/IP protocol suite do not exactly match those in the OSI model. The original TCP/IP protocol suite was defined as having four layers: host-to-network, internet, transport, and application (Behrouz, 2007). However, when TCP/IP is compared to OSI, we can say that the host-to-network layer is equivalent to the combination of the physical and data link layers. The internet layer is equivalent to the network layer, and the application layer is roughly doing the job of the session, presentation, and application layers with the transport layer in TCP/IP taking care of part of the duties of the session layer (Behrouz, 2007). (See

figure 2-14 and table 2.4). Commonly, the top three layers of the OSI model (Application, Presentation and Session) are considered as a single Application layer in the TCP/IP suite and the bottom two layers as well considered as a single Network Access layer. Because the TCP/IP suite has no unified session layer on which higher layers are built, these functions are typically carried out (or ignored) by individual applications. The most notable difference between TCP/IP and OSI models is the Application layer, as TCP/IP integrates a few steps of the OSI model into its Application layer (Tenouk, 20011). A simplified TCP/IP interpretation of the stack is shown in figure 2.15 below:

Table 2.4: A Brief Illustration of the TCP/IP Stack Functions (Behrouz, 2007).

Layer	**Description**	**Protocols**
Application	Defines TCP/IP application protocols and how host programs interface with transport layer services to use the network.	HTTP, Telnet, FTP, TFTP, SNMP, DNS, SMTP, X Windows, other application protocols
Transport	Provides communication session management between host computers. Defines the level of service and status of the connection used when transporting data.	TCP, UDP, RTP
Internet	Packages data into IP datagrams, which contain source and destination address information that is used to forward the datagrams between hosts and across networks. Performs routing of IP datagrams.	IP, ICMP, ARP, RARP
Network interface	Specifies details of how data is physically sent through the network, including how bits are electrically signaled by hardware devices that interface directly with a network medium, such as coaxial cable, optical fiber, or twisted-pair copper wire.	Ethernet, Token Ring, FDDI, X.25, Frame Relay, RS-232, v.

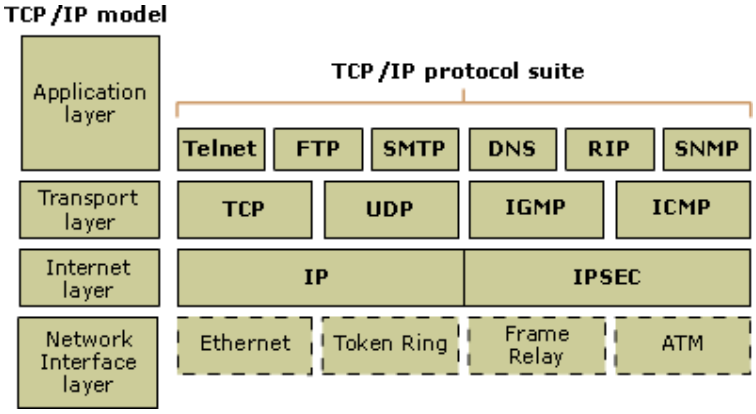


Figure 2.15: The TCP/IP Model

Each layer has its own independent data structures. Conceptually a layer is unaware of the data structure used by the layers above and below it. In reality, the data structures of a layer are designed to be compatible with the structures used by the surrounding layers for the sake of more efficient data transmission. Still, each layer has its own data structure and its own terminology to describe that structure. Figure 2-16 shows the terms used by different layers of TCP/IP to refer to the data being transmitted. As a general term, most networks refer to a transmitted data as packets of frames.

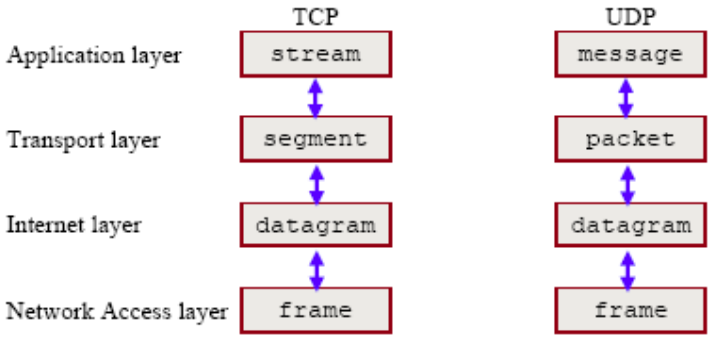


Figure 2.16: Different Terms for Packets at Different TCP/IP Layers

2.9.4 The Three Layer Hierarchical Model

Network design experts have developed the hierarchical network design model to help you develop a topology in discrete layers. Each layer can be focused on specific functions, allowing you to choose the right systems and features for the layer (Edrawsoft, 2011). For example, high-speed WAN routers can carry traffic across the enterprise WAN backbone, medium-speed routers can connect buildings at each campus, and switches can connect user devices and servers within buildings.

Networks that grow unheeded without any plan in place tend to develop in an unstructured format. Dr. Peter Welcher, the author of network design and technology articles for Cisco World and other publications, refers to unplanned networks as fur-ball network (Edrawsoft, 2011). A typical hierarchical topology is made of three layers (Edrawsoft, 2011):

- A core layer of high-end routers and switches that are optimized for availability and performance. This layer is considered the backbone of the network and includes the high-end switches and high-speed cables such as fibre cables. This layer of the network does not route traffic at the LAN. In addition, no packet manipulation is done by devices in this layer. Rather, this layer is concerned with speed and ensures reliable delivery of packets.
- A distribution layer of routers and switches that implement policies. This layer includes LAN-based routers and layer 3 switches. This layer ensures that packets are properly routed between subnets and and virtual local area networks (VLANs) in your enterprise. This layer is also called the Workgroup layer.
- An access layer that connects users via lower-end switches and wireless access points. This layer includes hubs and switches. This layer is also called the desktop layer because it focuses on connecting client nodes, such as workstations to the network. This

layer ensures that packets are delivered to end user computers.

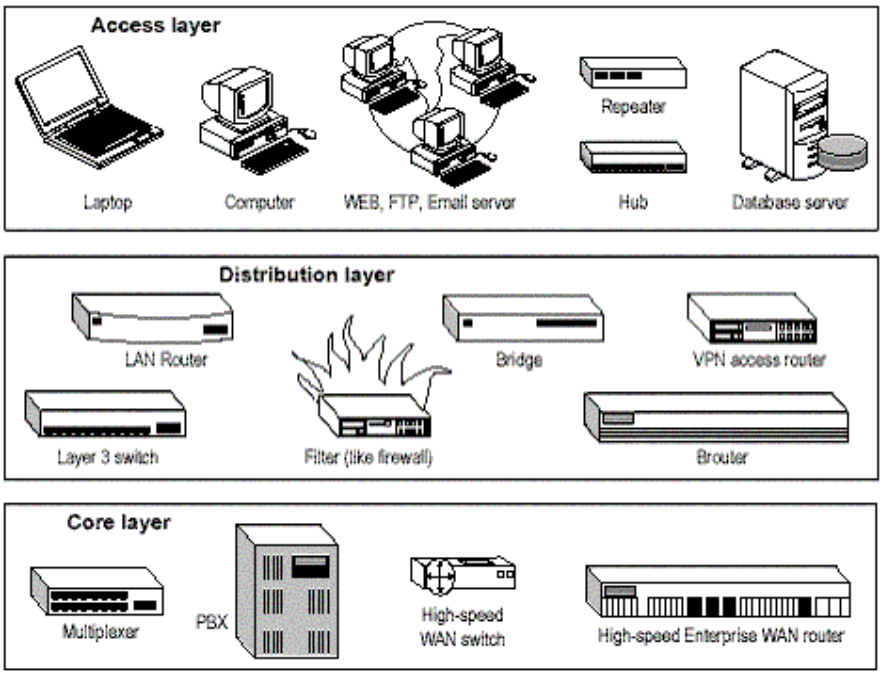


Figure 2.17: the three layer hierarchical model (Mcmcse, 2011)

2.9.4 Network Types

There are about eight types of network which are used world wide these days, both in houses and commercially. These networks are used on the bases of their scale and scope, historical reasons, preferences for networking industries, and their design and implementation issues (Wifinotes, 2011). LAN and WAN are mostly known and used widely. LAN, local area network was first invented for communication between two computers. LAN operates through cables and network cards. Later WLAN, Wireless local area network was formed through LAN concept, there are no wires involved in communication between computers, and Wireless LAN cards are required to connect to wireless network. LAN is the original network out of which other networks are formed according to requirements. They are as follow (Wifinotes, 2011).

a. LAN - Local Area Network

A local area network (LAN) is usually privately owned and links the devices in a single office, building, or campus (see Figure 1.10). Depending on the needs of an organization and the type of technology used, a LAN can be as simple as two PCs and a printer in someone's home office; or it can extend throughout a company and include audio and video peripherals. Currently, LAN size is limited to a few kilometres.

b. WAN – Wide Area Network

A wide area network (WAN) provides long-distance transmission of data, image, audio, and video information over large geographic areas that may comprise a country, a continent, or even the whole world. The Internet it self is the biggest example of Wide area network, WAN, which is covering the entire earth. WAN is distributed collection of geographically LANs. WAN used network protocols like ATM, X.25, and Frame Relay for long distance connectivity.

c. WLAN – Wireless Local Area Network

A WLAN is a LAN based on wireless network technology mostly referred as Wi-Fi. Unlike LAN, in WLAN no wires are used, but radio signals are the medium for communication. Wireless network cards are required to be installed in the systems for accessing any wireless network around. Mostly wireless cards connect to wireless routers for communication among computers or accessing WAN, internet.

d. MAN - Wide Area Network

A metropolitan area network (MAN) is a network with a size between a LAN and a WAN. It normally covers the area inside a town or a city. It is designed for customers who need a high-speed connectivity, normally to the Internet, and have endpoints spread over a city or part of city. A good example of a MAN is the part of

the telephone company network that can provide a high-speed DSL line to the customer. Another example is the cable TV network that originally was designed for cable TV, but today can also be used for high-speed data connection to the Internet.

e. SAN - Storage Area Network

SAM technology is used for data storage and it has no use for most of the organization but data oriented organizations. Storage area network connects servers to data storage devices by using Fiber channel technology.

f. **CAN - Campus Area Network**

Networking spanning with multiple LANs but smaller than a Metropolitan area network, MAN. This kind of network mostly used in relatively large universities or local business offices and buildings.

g. SAN –Storage Area Network

SAM technology is used for data storage and it has no use for most of the organization but data oriented organizations. Storage area network connects using fibre channel technology.

h. SAN – System Area Network

SAN, system area networks are also known as cluster area network and it connects high performance computers with high speed connections in cluster configuration.

2.9.6 Network Topology (Physical and Logical Topology)

Network topology is the layout pattern of interconnections of the various elements (links, nodes, etc.) of a computer or biological network (Groth & Toby, 2005) (ATIS, 2007) (Proulx et al., 2005). Network topologies may be physical or logical. Physical topology refers to the physical design of a network including the devices, location and cable installation. Logical topology refers to how data is actually transferred in a network as opposed

to its physical design. In general physical topology relates to a core network whereas logical topology relates to basic network (Wikipedia “Network Topology”, 2011).

Topology can be understood as the shape or structure of a network. This shape does not necessarily correspond to the actual physical design of the devices on the computer network. The computers on a home network can be arranged in a circle but it does not necessarily mean that it represents a ring topology.

The shape of the cabling layout used to link devices is called the physical topology of the network. This refers to the layout of cabling, the locations of nodes, and the interconnections between the nodes and the cabling (Groth & Toby, 2007). The physical topology of a network is determined by the capabilities of the network access devices and media, the level of control or fault tolerance desired, and the cost associated with cabling or telecommunications circuits. The logical topology, in contrast, is the way that the signals act on the network media, or the way that the data passes through the network from one device to the next without regard to the physical interconnection of the devices. A network's logical topology is not necessarily the same as its physical topology. For example, the original twisted pair Ethernet using repeater hubs was a logical bus topology with a physical star topology layout. Token Ring is a logical ring topology, but is wired a physical star from the Media Access Unit (Wikipedia “Network Topology”, 2011).

The logical classification of network topologies generally follows the same classifications as those in the physical classifications of network topologies but describes the path that the data takes between nodes being used as opposed to the actual physical connections between nodes. The logical topologies are generally determined by network protocols as opposed to being determined by the physical layout of cables, wires, and network devices or by the flow of the electrical signals, although in many cases the paths that the electrical signals take between nodes may closely match the logical flow of data, hence the convention of using

the terms logical topology and signal topology interchangeably (Wikipedia "Network Topology", 2011).

Logical topologies are often closely associated with Media Access Control methods and protocols. Logical topologies are able to be dynamically reconfigured by special types of equipment such as routers and switches.

The study of network topology recognizes seven basic topologies (Bicsi, 2002)

a. Bus

In local area networks where bus topology is used, each node is connected to a single cable. Each computer or server is connected to the single bus cable. A signal from the source travels in both directions to all machines connected on the bus cable until it finds the intended recipient. If the machine address does not match the intended address for the data, the machine ignores the data. Alternatively, if the data matches the machine address, the data is accepted. Since the bus topology consists of only one wire, it is rather inexpensive to implement when compared to other topologies. However, the low cost of implementing the technology is offset by the high cost of managing the network. Additionally, since only one cable is utilized, it can be the single point of failure. If the network cable is terminated on both ends and when without termination data transfer stop and when cable breaks, the entire network will be down.

Linear bus

The type of network topology in which all of the nodes of the network are connected to a common transmission medium which has exactly two endpoints (this is the 'bus', which is also commonly referred to as the backbone, or trunk) – all data that is transmitted between nodes in the network is transmitted over this common transmission medium and is able to be received by all nodes in the network simultaneously (Groth & Toby, 2005).

Distributed Bus

The type of network topology in which all of the nodes of the network are connected to a common transmission medium which has more than two endpoints that are created by adding branches to the main section of the transmission medium – the physical distributed bus topology functions in exactly the same fashion as the physical linear bus topology (i.e., all nodes share a common transmission medium).

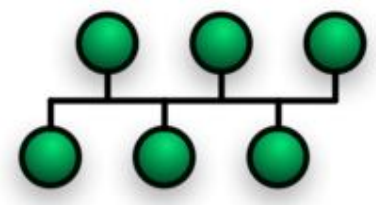


Figure 2.18: Bus Network Topology

b. Star

In local area networks with a star topology, each network host is connected to a central hub with a point-to-point connection. The network does not necessarily have to resemble a star to be classified as a star network, but all of the nodes on the network must be connected to one central device. All traffic that traverses the network passes through the central hub. The hub acts as a signal repeater. The star topology is considered the easiest topology to design and implement. An advantage of the star topology is the simplicity of adding additional nodes. The primary disadvantage of the star topology is that the hub represents a single point of failure.

Extended star

A type of network topology in which a network that is based upon the physical star topology has one or more repeaters between the central node (the 'hub' of the star) and

the peripheral or 'spoke' nodes, the repeaters being used to extend the maximum transmission distance of the point-to-point links between the central node and the peripheral nodes beyond that which is supported by the transmitter power of the central node or beyond that which is supported by the standard upon which the physical layer of the physical star network is based.

If the repeaters in a network that is based upon the physical extended star topology are replaced with hubs or switches, then a hybrid network topology is created that is referred to as a physical hierarchical star topology, although some texts make no distinction between the two topologies.

Distributed Star

A type of network topology that is composed of individual networks that are based upon the physical star topology connected together in a linear fashion – i.e., 'daisy-chained' – with no central or top level connection point (e.g., two or more 'stacked' hubs, along with their associated star connected nodes or 'spokes').

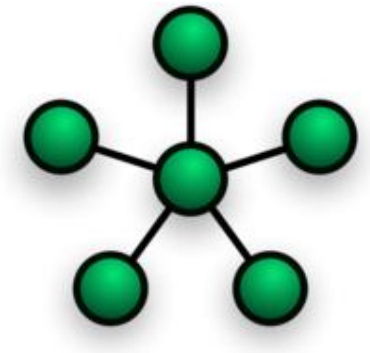


Figure 2.19: Star Network Topology

c. Ring

A network topology that is set up in a circular fashion in which data travels around the ring in one direction and each device on the right acts as a repeater to keep the signal strong as it travels. Each device incorporates a receiver for the incoming signal and a

transmitter to send the data on to the next device in the ring. The network is dependent on the ability of the signal to travel around the ring (Dave et al., 2002).

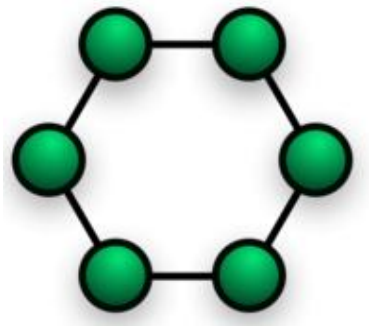


Figure 2.20: Ring Network Topology

d. Mesh

The value of fully meshed networks is proportional to the exponent of the number of subscribers, assuming that communicating groups of any two endpoints, up to and including all the endpoints, is approximated by Reed's Law.

The number of connections in a full mesh = n(n - 1) / 2.

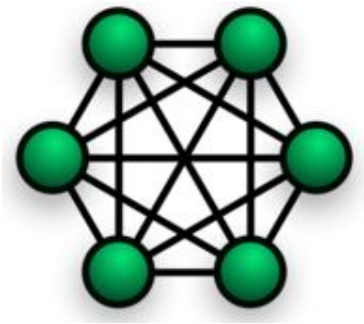


Fig 2.21: Mesh Network Topology

e. Tree

The type of network topology in which a central 'root' node (the top level of the hierarchy) is connected to one or more other nodes that are one level lower in the

hierarchy (i.e., the second level) with a point-to-point link between each of the second level nodes and the top level central 'root' node, while each of the second level nodes that are connected to the top level central 'root' node will also have one or more other nodes that are one level lower in the hierarchy (i.e., the third level) connected to it, also with a point-to-point link, the top level central 'root' node being the only node that has no other node above it in the hierarchy (The hierarchy of the tree is symmetrical.) Each node in the network having a specific fixed number, of nodes connected to it at the next lower level in the hierarchy, the number, being referred to as the 'branching factor' of the hierarchical tree. This tree has individual peripheral nodes.

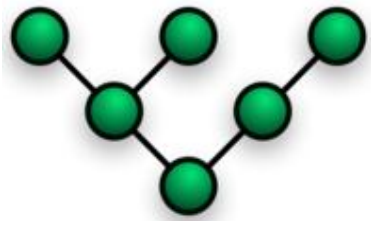


Fig 2.22: Tree Network Topology

f. Hybrid

Hybrid networks use a combination of any two or more topologies in such a way that the resulting network does not exhibit one of the standard topologies (e.g., bus, star, ring, etc.). For example, a tree network connected to a tree network is still a tree network topology. A hybrid topology is always produced when two different basic network topologies are connected. Two common examples for Hybrid network are: star ring network and star bus network.

- A Star ring network consists of two or more star topologies connected using a multistation access unit (MAU) as a centralized hub.

- A Star Bus network consists of two or more star topologies connected using a bus trunk (the bus trunk serves as the network's backbone)

While grid and torus networks have found popularity in high-performance computing applications, some systems have used genetic algorithms to design custom networks that have the fewest possible hops in between different nodes. Some of the resulting layouts are nearly incomprehensible, although they function quite well. [77]

A Snowflake topology is really a "Star of Stars" network, so it exhibits characteristics of a hybrid network topology but is not composed of two different basic network topologies being connected together.

g. Daisy Chain

Except for star-based networks, the easiest way to add more computers into a network is by daisy-chaining, or connecting each computer in series to the next. If a message is intended for a computer partway down the line, each system bounces it along in sequence until it reaches the destination. A daisy-chained network can take two basic forms: linear and ring.

- A linear topology puts a two-way link between one computer and the next. However, this was expensive in the early days of computing, since each computer (except for the ones at each end) required two receivers and two transmitters.
- By connecting the computers at each end, a ring topology can be formed. An advantage of the ring is that the number of transmitters and receivers can be cut in half, since a message will eventually loop all of the way around. When a node sends a message, the message is processed by each computer in the ring. If a computer is not the destination node, it will pass the message to the next node, until the message arrives at its destination. If the message is not accepted by any node on the network, it will travel

around the entire ring and return to the sender. This potentially results in a doubling of travel time for data.

2.9.7 Networking Media

A transmission medium can be broadly defined as anything that can carry information from a source to a destination. For example, the transmission medium for two people having a dinner conversation is the air. The air can also be used to convey the message in a smoke signal or semaphore. For a written message, the transmission medium might be a mail carrier, a truck, or an airplane. In data communications the definition of the information and the transmission medium is more specific. The transmission medium is usually free space, metallic cable, or fibre-optic cable. (Behrouz, 2007).

Until 1985, there was a lack of standards for building telecommunications cabling systems. At that time, the Computer Communications Industry Association (CCIA) requested the Electronic Industries Association (EIA) to develop a standard for building telecommunications cabling systems (Wikipedia “Telecommunication Networks”, 2011).

EIA accepted the request and assigned the project to Engineering Committee TR-41. The TR-41 Committee established the following working groups to develop building telecommunications cabling system standards (Wikipedia “Telecommunication Networks”, 2011).

- TR-41.7.2 Working Group for Grounding and Bonding.
- TR-41.7.3 Working Group on Electromagnetic Compatibility Considerations.
- TR-41.8.1 Working Group on Commercial and Industrial Building Wiring Standard.
- TR-41.8.2 Working Group on Residential and Light Commercial Building Wiring Standard.
- TR-41.8.3 Working Group on Telecommunications Pathways and Spaces Administration.

- TR-41.8.4 Working Group on Backbone Cabling Systems for Residential and Light Commercial Buildings.
- TR-41.8.5 Working Group for Definitions.

The above-mentioned Working Groups, under the jurisdiction of the TR-41.8 Subcommittee and approved by the Technical Committee TR-41, have developed a series of technical standards on building cabling for telecommunications products and services. These documents are meant to fill a recognized need in the telecommunications industry brought about by the evolving industry structure. Currently available telecommunications wiring standards are:

- ANSI/TIA/EIA-568-A, Commercial Building Telecommunications Cabling Standard.
- ANSI/EIA/TIA-569, Commercial Building Standard for Telecommunications Pathways and Spaces.
- ANSI/EIA/TIA-570, Residential and Light Commercial Telecommunications Wiring Standard.
- ANSI/TIA/EIA-606, Administration Standard for the Telecommunications Infrastructure of Commercial Buildings.
- ANSI/TIA/EIA-607, Grounding and Bonding Requirements for Telecommunications in Commercial Buildings (Bicsi, 2011).

In telecommunications, transmission media can be divided into two broad categories: guided and unguided. Guided media include twisted-pair cable, coaxial cable, and fibre-optic cable. Unguided medium is free space. Guided media are those that provide a conduit from one device to another (Behrouz, 2007). A signal travelling along any of these media is directed and contained by the physical limits of the medium. Twisted-pair and coaxial cable use metallic (copper) conductors that accept and transport signals in the form of electric current. Optical fibre is a cable that accepts and transports signals in the form of light.

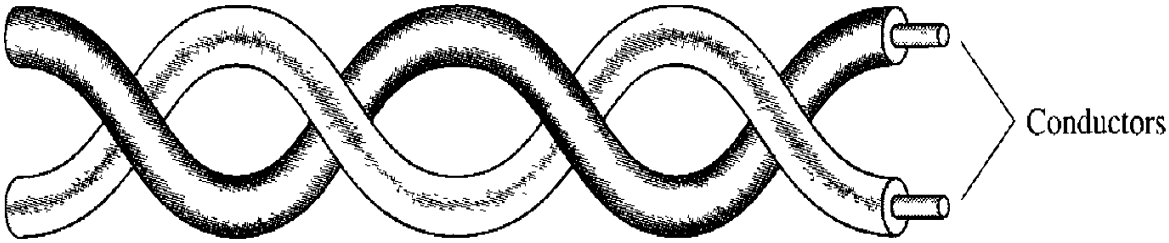


Fig 2.23a: A twisted pair cable (Behrouz, 2007).

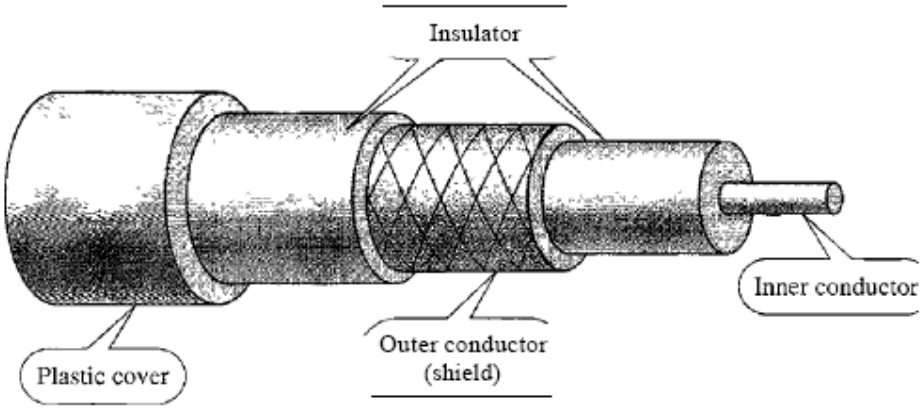


Figure 2.23b: A cross section of a coaxial cable (Behrouz, 2007).

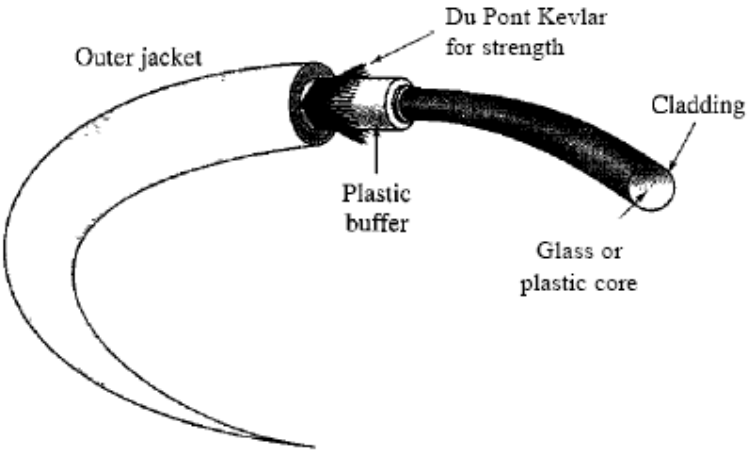


Figure 2.23c: Cross section of a fibre optics cable (Behrouz, 2007).

Unguided media on the other hand transport electromagnetic waves without using a physical conductor. This type of communication is often referred to as wireless communication. Signals are normally broadcast through free space and thus are available to

anyone who has a device capable of receiving them. Figure 2.24 shows the part of the electromagnetic spectrum, ranging from 3 kHz to 900 THz, used for wireless communication.

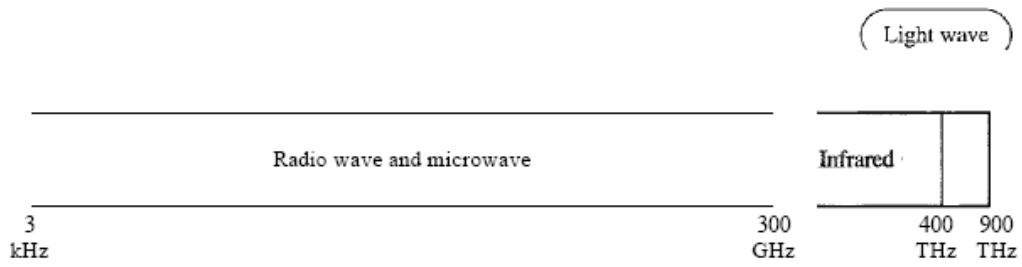


Fig 2-24: Part of the electromagnetic spectrum (Behrouz, 2007).

We can divide wireless transmission into three broad groups: radio waves, microwaves, and infrared waves.

Although there is no clear-cut demarcation between radio waves and microwaves, electromagnetic waves ranging in frequencies between 3 kHz and 1 GHz are normally called radio waves; waves ranging in frequencies between 1 and 300 GHz are called microwaves. However, the behaviour of the waves, rather than the frequencies, is a better criterion for classification. Radio waves, particularly those waves that propagate in the sky mode, can travel long distances. This makes radio waves a good candidate for long-distance broadcasting such as AM radio.

Electromagnetic waves having frequencies between I and 300 GHz are called microwaves. Microwaves are unidirectional. When an antenna transmits microwave waves, they can be narrowly focused. This means that the sending and receiving antennas need to be aligned. The unidirectional property has an obvious advantage. A pair of antennas can be aligned without interfering with another pair of aligned antennas.

Infrared waves, with frequencies from 300 GHz to 400 THz (wavelengths from 1 mm to 770 nm), can be used for short-range communication. Infrared waves, having high frequencies, cannot penetrate walls. This advantageous characteristic prevents interference

between one system and another; a short-range communication system in one room cannot be affected by another system in the next room.

2.9.8 Networking Components

All telecommunication networks are made up of five basic components that are present in each network environment regardless of type or use. These basic components include terminals, telecommunications processors, telecommunications channels, computers, and telecommunications control software (Wikipedia "Telecommunication Networks", 2011).

- Terminals are the starting and stopping points in any telecommunication network environment. Any input or output device that is used to transmit or receive data can be classified as a terminal component.

- Telecommunications processors support data transmission and reception between terminals and computers by providing a variety of control and support functions. (i.e. convert data from digital to analog and back)

- Telecommunications channels are the way by which data is transmitted and received. Telecommunication channels are created through a variety of media of which the most popular include copper wires and coaxial cables (structured cabling). Fiber-optic cables are increasingly used to bring faster and more robust connections to businesses and homes.

- In a telecommunication environment computers are connected through media to perform their communication assignments.
- Telecommunications control software is present on all networked computers and is responsible for controlling network activities and functionality (O'Brien & Marakas, 2008).

Early networks were built without computers, but late in the 20th century their switching centres were computerized or the networks replaced with computer networks.

Some of the basic networking components used in a typical networked environment include the following: Hubs, Switches, Bridges, Routers, Gateways, CSU/DSU, Wireless access points (WAPs), Modems, Network interface cards (NICs), ISDN adapters, Transceivers, Firewalls (Personshighered, 2011)

a. Hubs:

Hubs are simple network devices, and their simplicity is reflected in their low cost. Small hubs with four of five ports are available for small networks and are quite cheap. Hubs with more ports are available for networks that require greater capacity. Figure 2-25 shows an example of a workgroup hub. Computers connect to a hub via a length of twisted-pair cabling. In addition to ports for connecting computers, even an inexpensive hub generally has a port designated as an uplink port that enables the hub to be connected to another hub to create larger networks.

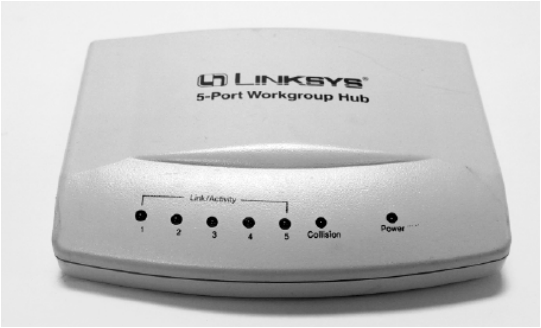


Figure 2.25: A Workgroup Hub

b. Switches:

On the surface, a switch looks much like a hub. Despite their similar appearance, switches arefar more efficient than hubs and are far more desirable for today's network environments. Figure 2.26 shows an example of a 32-port Ethernet

switch. As with a hub, computers connect to a switch via a length of twisted-pair cable. Multiple switches are often interconnected to create larger networks. Despite their similarity in appearance and their identical physical connections to computers, switches offer significant operational advantages over hubs. A hub forwards data to all ports, regardless of whether the data is intended for the system connected to the port. This arrangement is inefficient; however, it requires little intelligence on the part of the hub, which is why hubs are inexpensive. Rather than forwarding data to all the connected ports, a switch forwards data only to the port on which the destination system is connected. It looks at the Media Access Control (MAC) addresses of the devices connected to it to determine the correct port. A MAC address is a unique number that is stamped into every NIC. By forwarding data only to the system to which the data is addressed, the switch decreases the amount of traffic on each network link dramatically. In effect, the switch literally channels (or switches, if you prefer) data between the ports.

Fig 2.26: A 32 Port Ethernet Switch

c. Bridges:

Bridges are networking devices that connect networks. Sometimes it is necessary to divide networks into subnets to reduce the amount of traffic on each larger subnet or for security reasons. Once divided, the bridge connects the two subnets and manages the traffic flow between them. Today, network switches have largely replaced bridges.

A bridge functions by blocking or forwarding data, based on the destination MAC address written into each frame of data. If the bridge believes the destination address is on a network other than that from which the data was received, it can forward the data to the other networks to which it is connected. If the address is not on the other side of the bridge, the data is blocked from passing. Bridges “learn” the MAC addresses of devices on connected networks by “listening” to network traffic and recording the network from which the traffic originates.

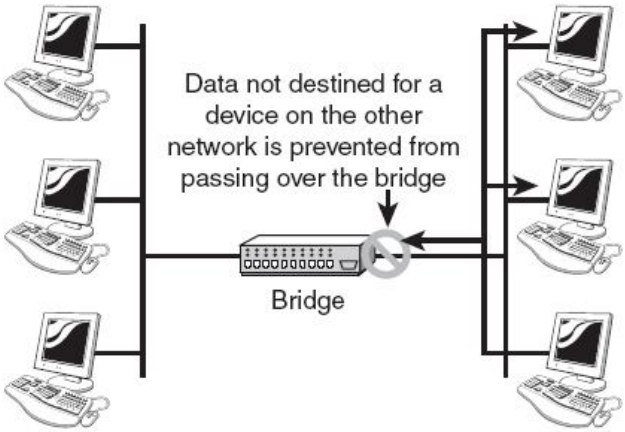


Fig 2-27: Diagram Showing how a Bridge Works

d. Routers:

Routers are an increasingly common sight in any network environment, from a small home office that uses one to connect to an Internet service provider (ISP) to a corporate IT environment where racks of routers manage data communication with disparate remote sites. Routers make internetworking possible, and in view of this, they warrant detailed attention. Routers are network devices that literally route data around the network. By examining data as it arrives, the router can determine the destination address for the data; then, by using tables of defined routes, the router determines the best way for the data to continue its journey. Unlike bridges and switches, which use the hardware-configured MAC address to determine the

destination of the data, routers use the software-configured network address to make decisions. This approach makes routers more functional than bridges or switches, and it also makes them more complex because they have to work harder to determine the information. Figure 2.25 shows basically how a router functions. The basic requirement for a router is that it must have at least two network interfaces. If they are LAN interfaces, the router can manage and route the information between two LAN segments. More commonly, a router is used to provide connectivity across wide area network (WAN) links.

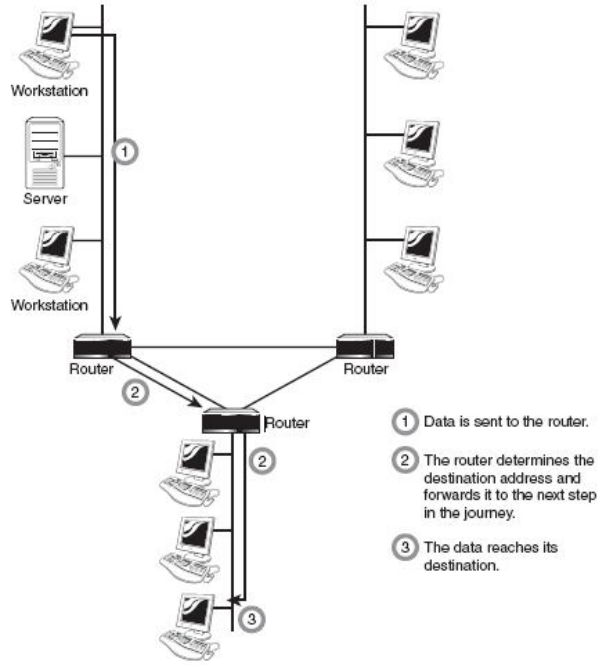


Figure 2.28: Diagram Showing the Basic Functions of a Router

e. Gateways:

The term gateway is applied to any device, system, or software application that can perform the function of translating data from one format to another. The key feature of a gateway is that it converts the format of the data, not the data itself. You can use gateway functionality in many ways. For example, a router that can route data from an

IPX network to an IP network is, technically, a gateway. The same can be said of a translational bridge that, converts from an Ethernet network to a Token Ring network and back again. Software gateways can be found everywhere. Many companies use an email system such as Microsoft Exchange or Novell GroupWise. These systems transmit mail internally in a certain format. When email needs to be sent across the Internet to users using a different email system, the email must be converted to another format, usually to Simple Mail Transfer Protocol (SMTP). This conversion process is performed by a software gateway. Another good (and often used) example of a gateway involves the Systems Network Architecture (SNA) gateway, which converts the data format used on a PC to that used on an IBM mainframe or minicomputer. A system that acts as an SNA gateway sits between the client PC and the mainframe and translates requests and replies from both directions. Figure 2.26 shows how this would work in a practical implementation.

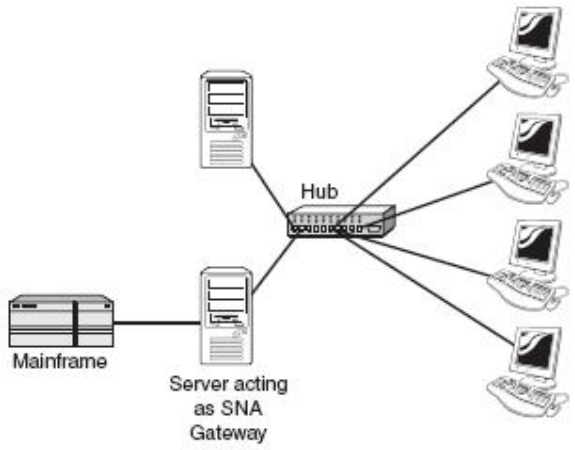


Figure 2.29: An SNA gateway

f. CSUs/DSUs:

A Channel Service Unit/Data Service Unit (CSU/DSU) acts as a translator between the LAN data format and the WAN data format. Such a conversion is necessary because the technologies used on WAN links are different from those used on LANs.

Some consider a CSU/DSU as a type of digital modem; but unlike a normal modem, which changes the signal from digital to analog, a CSU/DSU changes the signal from one digital format to another. Figure 2.27 shows how a CSU/DSU might fit into a network. A CSU/DSU has physical connections for the LAN equipment, normally via a serial interface, and another connection for a WAN. Traditionally, the CSU/DSU has been in a separate box from other networking equipment; however, the increasing use of WAN links means that some router manufacturers are now including the CSU/DSU functionality in routers or are providing the expansion capability to do so.

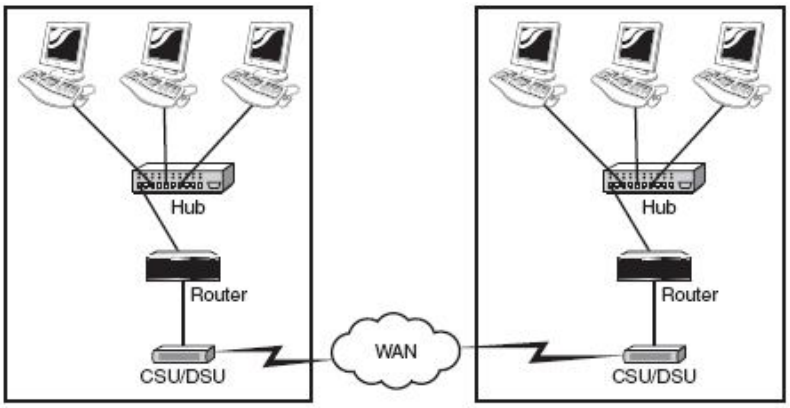


Figure 2.30: Diagram Showing how a CSU/DSU is used in a Network

g. Modems

Modem is a contraction of the terms modulator and demodulator. Modems perform a simple function: They translate digital signals from a computer into analog signals that can travel across conventional phone lines. The modem modulates the signal at the sending end and demodulates at the receiving end. Modems provide a relatively slow method of communication. Modems are available as internal devices that plug into expansion slots in a system; external devices that plug into serial or USB ports; PCMCIA cards designed for use in laptops; and specialized devices designed for use in systems such as handheld computers. In addition, many laptops now come with

integrated modems. For large-scale modem implementations, such as at an ISP, rack-mounted modems are also available. Figure 2.28 shows an internal modem and a PCMCIA modem.

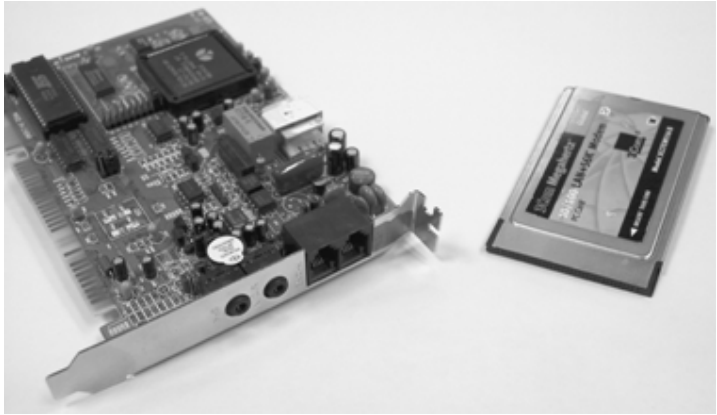


Figure 2.31: An Internet Modem (left) and a PCMCIA Modem (right).

h. Network Interface Cards (NICs):

NICs—sometimes called network cards—are the mechanisms by which computers connect to a network. NICs come in all shapes and sizes, and they come in prices to suit all budgets. Figure 2-29 show a typical structure of an expansion NIC

Figure 2.32: An Expansion NIC

i. ISDN Terminal Adapters:

When the speed provided by a modem just isn't enough, you must seek alternatives. One of the speedier options available is an ISDN link. ISDN is a digital communication method that can be used over a conventional phone line, although certain criteria must be met for an ISDN line to be available (such as the availability of the service and the proximity of your location to the telco's site). To use ISDN, you need a device called an ISDN terminal adapter. ISDN terminal adapters are available as add-in expansion cards installed into computers, external devices that connect to the serial interfaces of PC systems, or modules in a router. You can think of an ISDN terminal adapter as a kind of digital modem. (Remember that a modem converts a signal from digital to analog and vice versa. An ISDN terminal adapter translates the signal between two digital formats.) Figure 3.24 shows an external ISDN terminal adapter, and Figure 2.30 shows an example of an internal ISDN adapter. Notice that an ISDN terminal adapter is similar in appearance to a standard NIC.

Figure 2.33 An Internal ISDN Adaptor

j. Wireless Access Point

Wireless access points, referred to as wireless APs, are a transmitter and receiver (transceiver) device used for wireless LAN (WLAN) radio signals. A wireless access point is typically a separate network device with a built-in antenna, transmitter, and adapter. Wireless access points use the wireless infrastructure network mode to provide a connection point between WLANs and a wired Ethernet LAN. wireless access points also typically have several ports allowing a way to expand the network to support additional clients. Depending on the size of the network, one or more wireless access points may be required. Additional WAPs are used to allow access to more wireless clients and to expand the range of the wireless network. Each wireless access point is limited by a transmissions range, the distance a client can be from a wireless access point and still get a useable signal. The actual distance depends on the wireless standard being used and the obstructions and environmental conditions between the client and the wireless access point. As mentioned, a wireless access point is used in an infrastructure wireless network design. Used in the infrastructure mode, the WAP receives transmissions from wireless devices within a specific range and transmits those signals to the network beyond. This network may be a private Ethernet network or the Internet. The transmission range a wireless access point can support and number of wireless devices that can connect to it depends on the wireless standard being used and the signal interference between the two devices. In infrastructure wireless networking, there may be multiple access points to cover a large area or only a single access point for a small area such as a single home or small building. Figure 2.31 shows an example of an infrastructure wireless network using a wireless access point.

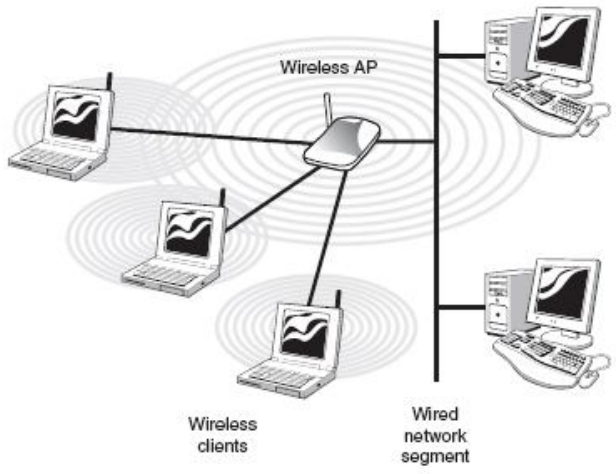


Figure 2.34: Diagram of an Infrastructure Wireless Network using a Wireless Access Point

k. Transceivers

The term transceiver does not necessarily describe a separate network device but rather an integrated technology embedded in devices such as network cards. In a network environment, a transceiver gets its name from being both a transmitter and a receiver of signals, such as analog or digital. Technically, on a LAN the transceiver is responsible to place signals onto the network media and also detecting incoming signals travelling through the same cable. Given the description of the function of a transceiver, it makes sense that this technology would be found with network cards. Although transceivers are found in network cards, they can be external devices as well. As far as networking is concerned, transceivers can ship as a module or chip type. Chip transceivers are small and are inserted into a system board or wired directly on a circuit board. Module transceivers are external to the network and are installed and function similarly to other computer peripherals, or they may function as standalone devices. There are many types of transceivers: RF transceivers, fibre-optic transceivers, Ethernet transceivers, wireless (WAP) transceivers, and more. Though each of these media types is different, the function of the transceiver remains the

same. Each type of the transceiver used has different characteristics such as the number of ports available to connect to the network and whether full-duplex communication is supported. Listed with transceivers in the CompTIA objectives are media converters. Media converters are a technology that allows administrators to interconnect different media types—for example, twisted pair, fibre, and thin or thick coax—within an existing network. Using a media converter, it is possible to connect newer 100Mbps, Gigabit Ethernet, or ATM equipment to existing networks such as 10Base-T or 100Base-T. They can also be used in pairs to insert a fibre segment into copper networks to increase cabling distances and enhance immunity to electromagnetic interference (EMI).

l. Firewalls

Today, firewalls are an essential part of a network's design. A firewall is a networking device, either hardware or software based, that controls access to your organization's network. This controlled access is designed to protect data and resources from outside threat. To do this, firewalls are typically placed at entry/exit points of a network. For example, a firewall might be placed between an internal network and the Internet. After the firewall is in place, it can control access in and out of that point. Although firewalls typically protect internal networks from public networks, they are also used to control access between specific network segments within a network. For example, you might place a firewall between the Accounts Department and the Sales Department. As mentioned, firewalls can be implemented through software or through a dedicated hardware device. Organizations implement software firewalls through network operating systems (NOS) such as Linux/Unix, Windows servers, and Mac OS servers. The firewall is configured on the server to allow or permit certain types of network traffic. In small offices and for regular home use, a firewall is commonly

installed on the local system and configured to control traffic. Many third-party firewalls are available. Hardware firewalls are used in networks of all sizes today. Hardware firewalls are often dedicated network devices and can be implemented with very little configuration and protect all system behind it from outside sources. Hardware firewalls are readily available and often combined with other devices today. For example, many broadband routers and wireless access points have firewall functionality built in.

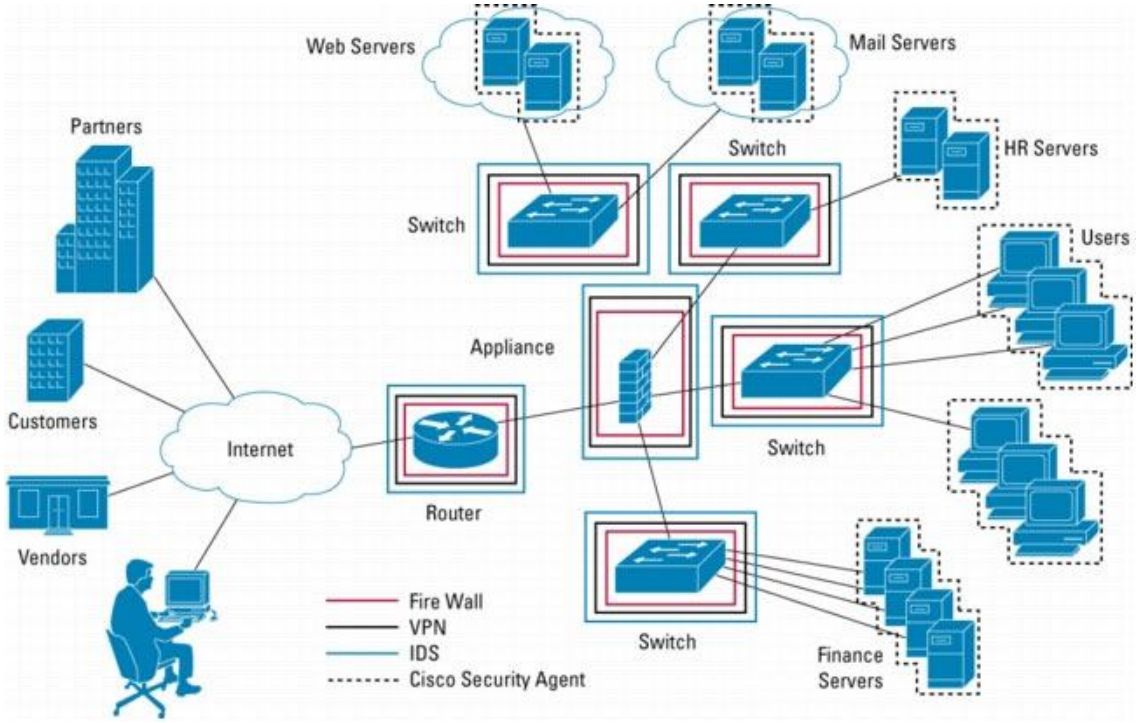


Figure 2.35: Diagram Showing Various Networking Components (Cisco Self-Defending Network, 2011).

2.10 System Design

Systems design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. There are various system design methodologies available for use and the choice of a methodology depends on the kind of

system to be developed. In the field of computer application and information technology, the Object-oriented analysis and design methods are becoming the most widely used methods for design (Wikipedia “System Design”, 2011).

2.10.1 Overview of System Development Methodology

A system development methodology can be simply defined as a set of procedure that one follows from the beginning to the completion of the system development process (Wikipedia “SDLC”, 2011). The nature of the methodology is dependent on a number of factors, including the software development environment, the organization's practices, the nature or type of the software being developed, the requirements of the users, the qualification and training of the software development team, the available hardware and software resources, the availability of existing design modules, and even the budget and the time schedule. Since the 1970s, there has been a proliferation of software design methodologies. Different methodologies have been developed to resolve different types of problems. In describing these problems, it is often possible or necessary to group problems with similar characteristics together. This is called the problem domain. Many of these methodologies have evolved from the specific milieu where the software was developed. By this it is meant that specific methodologies are often developed (to be applied) to resolve certain "classes" of problems, also called domain of application, for which it is well-suited.

According to Russell,

Even though the design mechanics are different in each methodology each of these methods have a number of common characteristics: (1) a mechanism for the translation of information domain representation into design representation, (2) a representing functional components and their interfaces, (3) heuristics for refinement and partitioning, and (4) guidelines for quality assessment (Russell, 2002).

There are two broad categories of design methodologies: the systematic and the formal types. As the name imply, the formal type makes extensive use of mathematical notations for the object transformations and for checking consistencies. The systematic type is less mathematical and it consists of the procedural component, which prescribes what action or task to perform and the representation component, which prescribes how the software structure should be represented. Generally, techniques from the systematic design methodologies can be integrated and can utilize representation schemes from other techniques when and as appropriate. Due to the fact that methodologies have been developed from different milieu specifically to address certain problems or groups of problems, there is no common baseline on which to evaluate or compare the methodologies against each other. However, the underlying principles of the methodologies can be analyzed and examined for a better understanding of the basis for each methodology. With a better understanding of the methodology, its domain of application can be more effectively applied or more accurately defined. Generally, alternative design allows for important trade-off analysis before coding the software. Thus, familiarity with several methodologies makes creating competitive designs more logical and systematic with less reliance on inspiration (Wikipedia "SDLC", 2011).

2.10.2 System Development Life Cycle

The systems development life cycle (SDLC), or software development life cycle in systems engineering, information systems and software engineering, is a process of creating or altering information systems, and the models and methodologies that people use to develop these systems.

In software engineering the SDLC concept underpins many kinds of software development methodologies. These methodologies form the framework for planning and controlling the creation of an information system: the software development process.

The System Development Life Cycle framework provides a sequence of activities for system designers and developers to follow. It consists of a set of steps or phases in which each phase of the SDLC uses the results of the previous one. A Systems Development Life Cycle (SDLC) adheres to important phases that are essential for developers, such as planning, analysis, design, and implementation, and are explained in the section below. A number of system development life cycle (SDLC) models have been created: waterfall, fountain, spiral, build and fix, rapid prototyping, incremental, and synchronize and stabilize. The oldest of these, and the best known, is the waterfall model: a sequence of stages in which the output of each stage becomes the input for the next. These stages can be characterized and divided up in different ways, including the following:

- Project planning, feasibility study: Establishes a high-level view of the intended project and determines its goals.
- Systems analysis, requirements definition: Defines project goals into defined functions and operation of the intended application. Analyzes end-user information needs.
- Systems design: Describes desired features and operations in detail, including screen layouts, business rules, process diagrams, pseudocode and other documentation.
- Implementation: The real code is written here.
- Integration and testing: Brings all the pieces together into a special testing environment, then checks for errors, bugs and interoperability.
- Acceptance, installation, deployment: The final stage of initial development, where the software is put into production and runs actual business.
- Maintenance: What happens during the rest of the software's life: changes, correction, additions, moves to a different computing platform and more. This, the

least glamorous and perhaps most important step of all, goes on seemingly forever (Russell, 2002).

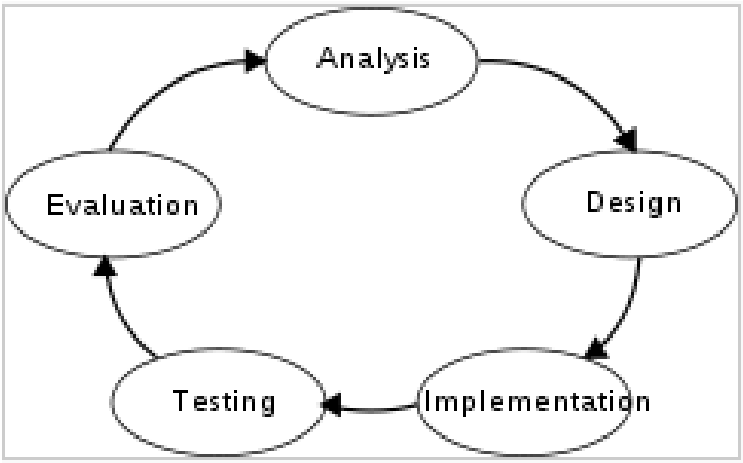


Figure 2.36: Model of a System Development Life Cycle (Russell, 2002).

CHAPTER THREE

3.0 DESIGN METHODOLOGY AND ANALYSIS FOR THE TIRMP

Resource management has become a key issue in every institution especially as expansion is inevitable. Thus the need for a better and more accessible platform for resource management and evaluation has become a great puzzle every institution needs to crack. The problems currently being experienced in resource management within higher institutions especially in developing economies can be summed up as follow:

- The problem of loss of data stored in file cabinets or on single computers/servers, arising due to fire outbreak, mismanagement or equipment failure.
- High cost of replacing obsolete equipment due to constant change in technology and increased traffic on the existing system
- High cost of maintaining local servers due to inadequate power supply
- Increased awareness on the availability of improved computing platforms giving rise to a constant increase in the demand for convenience in resource management.
- The gap created between staffs as a result of poor information flow across the institutional setting.
- The need for a more cost efficient platform for information resource management.
- The flexibility of the system in terms of usability, timeliness and availability.

In solving the above problem, several interactions will be made with high level officers within the higher institution, and other staff members especially the registrar, information departments, and the heads of departments and deans of studies among others. A thorough study of the existing paper system for appraisal of staff and collection of staff records will also be made.

3.1 Software Design Methodologies

A system development methodology refers to the framework that is used to structure, plan, and control the process of developing an information system (CMS, 2012). A wide variety of such frameworks have evolved over the years, each with its own recognized strengths and weaknesses. One system development methodology is not necessarily suitable for use by all projects. Each of the available methodologies is best suited to specific kinds of projects, based on various technical, organizational, project and team considerations.

This section describes various design methodologies used in system development illustrating their basic principle of usage and their merits/demerits. It went further to provide reasons while the prototyping technique is best suitable for the development of web based application and particularly the TIRMP application. Adopted by the researcher is the idea proposed by David Gustafson (David, 2002). In his book 'software engineering', he presented four common software life cycle models. The following sub section describes these models.

3.1.1 The Linear Sequential Model

This model, shown in Figure 3.1, is also called the waterfall model, since the typical diagram looks like a series of cascades. First described by Royce in 1970, it was the first realization of a standard sequence of tasks (David, 2002). There are many versions of the waterfall model. Although the specific development tasks will occur in almost every development, there are many ways to divide them into phases. Note that in this version of the waterfall, the project planning activities are included in the requirements phase. Similarly, the delivery and maintenance phases have been left off.

This model though basic has lots of limitation in software design. Its rigid nature has makes iteration and interaction with the users of the software limited.

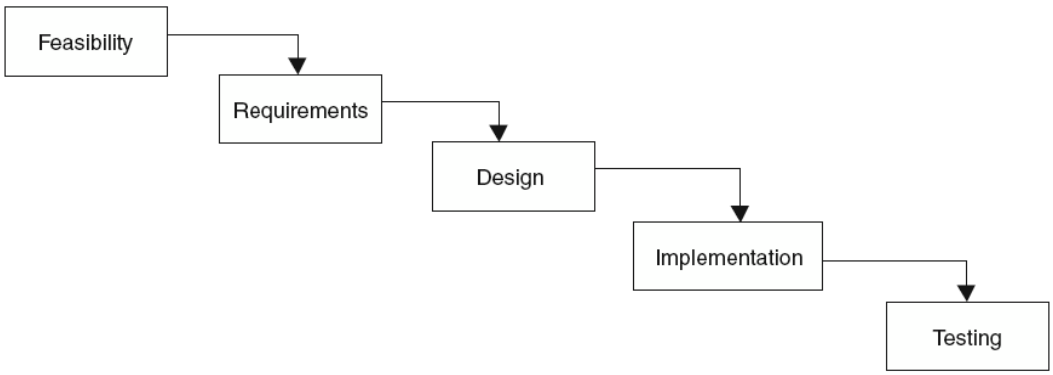


Figure 3.1: Waterfall Model (David, 2002)

3.1.2 Boehm's Spiral Model

B. Boehm introduced the spiral model (Boehm, 1988). The image of the model is a spiral that starts in the middle and continually revisits the basic tasks of customer communication, planning, risk analysis, engineering, construction and release, and customer evaluation.

3.1.3 Incremental Model

D. L. Parnas proposed the incremental model (Parnas, 1979). The goal was to design and deliver to the customer a minimal subset of the whole system that was still a useful system. The process will continue to iterate through the whole life cycle with additional minimal increments. The advantages include giving the customer a working system early and working increments.

3.1.4 Prototyping Model

This software life cycle model builds a throwaway version (or prototype). This prototype is intended to test concepts and the requirements. The prototype will be used to demonstrate the proposed behaviour to the customers. After agreement from the customer, then the software

development usually follows the same phases as the linear sequential model. The effort spent on the prototype usually pays for itself by not developing unnecessary features.

3.2 Selecting a Design Methodology for the TIRMP App.

The researcher came up with the real time evolutionary prototyping model as the best tool for the design of the TIRMP app that runs on the cloud. After a thorough review of existing design models, the evolutionary prototyping seemed most appropriate for the application design but with a few adjustments to its implementation this gave rise to the real time evolutionary prototyping model. Several cloud computing apps like facebook and twitter keep evolving with new features as days go by. This adjustment is what the researcher leveraged on to come up with a slight modification of the prototyping model. The following sub sections describe prototyping, evolutionary prototyping, throw away prototyping and the researches real time evolutionary prototyping adopted for the TIRMP app.

3.2.1 The Prototyping Process

According to Ian Som meville, Prototyping is the rapid development of a system. In the past, the developed system was normally thought of as inferior in some way to the required system so further development was required. Now, the boundary between prototyping and normal system development is blurred and many systems are developed using an evolutionary approach (Ian, 2000).

The principal use is to help customers and developers understand the requirements for the system i.e:

- Requirements elicitation. Users can experiment with a prototype to see how the system supports their work
- Requirements validation. The prototype can reveal errors and omissions in the requirements

Prototyping can be considered as a risk reduction activity which reduces requirements risks. Some of the benefits derived from the use of the prototyping model include the following:

- Misunderstandings between software users and developers are exposed
- Missing services may be detected and confusing services may be identified
- A working system is available early in the process
- The prototype may serve as a basis for deriving a system specification
- The system can support user training and system testing
- Improved system usability
- Closer match to the system needed
- Improved design quality
- Improved maintainability
- Reduced overall development effort

The prototyping process is illustrated in figure 3.2

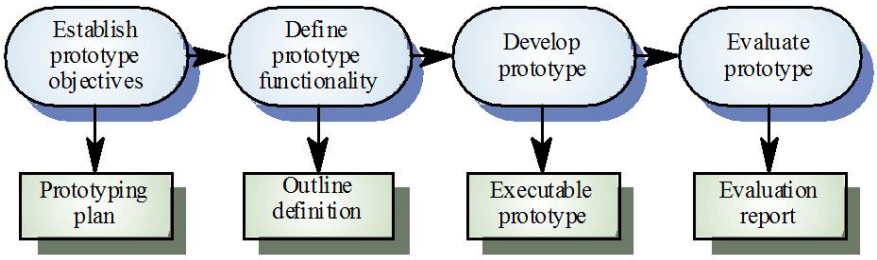


Figure 3.2: The prototyping process

3.2.2 Evolutionary Prototyping

Evolutionary prototyping model is used for systems where the specification cannot be developed in advance e.g. AI systems and user interface systems. It is based on techniques which allow rapid system iterations and verification is impossible as there is no specification. Figure 3.3 shows the evolutionary process with a feedback at the use prototype stage. Ones the system is developed and tested, it is either further worked upon based on testing or delivered if

considered adequate. This process is used mainly in software design project especially user friendly ones.

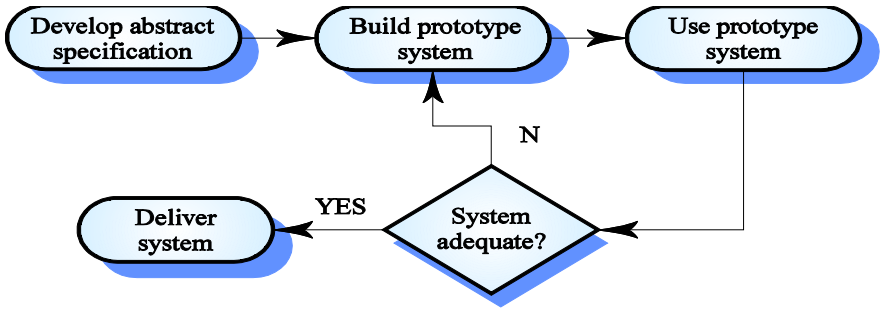


Figure 3.3: The evolutionary prototyping process

The evolutionary prototyping technique, specification, design and implementation are inter-twined. The system is developed as a series of increments that are delivered to the customer and makes use of such rapid system development tools such as CASE tools and 4GLs. Here the user interfaces are usually developed using a GUI development toolkit available to the designer. Some of the advantages this technique provides includes the following:

- Accelerated delivery of the system: Rapid delivery and deployment are sometimes more important than functionality or long-term software maintainability
- User engagement with the system: Not only is the system more likely to meet user requirements, they are more likely to commit to the use of the system

The evolutionary technique has some problems associated with it. These challenges include the following:

- Management problems: Existing management processes assume a waterfall model of development and specialist skills are required which may not be available in all development teams
- Maintenance problems: Continual change tends to corrupt system structure so long-term maintenance is expensive

- Contractual problems

3.2.3 Throw Away Prototyping

The throw away prototyping model is used to reduce requirements risk in system development process. The prototype is developed from an initial specification, delivered for experiment then discarded. The throw-away prototype is not considered as a final system. Figure 3.4 shows the throw away prototyping process.

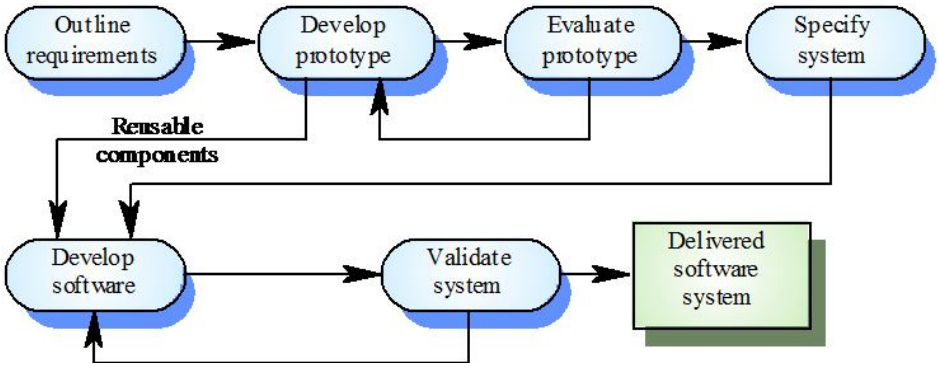


Figure 3.4: The throw away prototyping process

3.2.4 The Real Time Evolutionary Prototyping Model for the TIRMP App Design

The real time evolutionary prototyping model is a system development model proposed by the researcher for the development of cloud based applications. In this model the system is tested on the fly as it is being designed. The system design process continues even after deployment or better still the design process is done on a deployed mode. Figure 3.5 shows the real time evolutionary prototyping process.

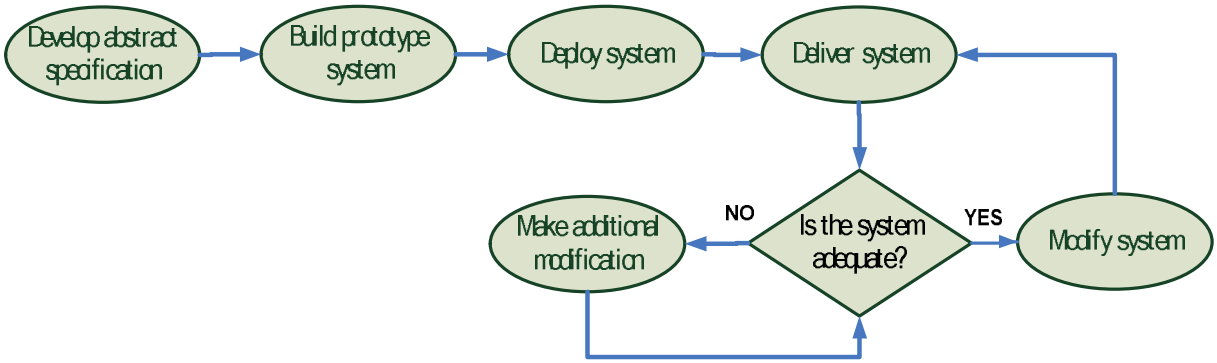


Figure 3.5: Real time Evolutionary Prototyping Process

The real time evolutionary prototyping is basically used for systems that are continuously modified even when delivered or deployed. Continuous modification is a process that never stops even when the system is delivered. The process takes into consideration the fact that technology keeps changing with time, and there is a need to keep up with this changed by adopting a flexible system that can be modified continuously.

The real time evolutionary prototyping model possesses the advantages of the prototyping model with the additional advantage of continuous modification after deployment.

a. **Developing the Abstract Specification:**

   In developing the abstract specification, the existing system is studied and a structure is drawn for the new system. This process also involves interaction with persons the system is intended for to ensure that salient fields are captured. The entity relationship diagram (ERD) is build for the system after determining the tables and fields for the application.

b. **Build the Prototype System**

   Here the system is build on the required platform. The ERD diagram is used to build the various application interfaces and interlinked to achieve set requirement. Security features are also configured at this stage.

c. **Deploy System**

   The system is deployed at this stage. For systems that are deployed while in development process, this stage is superimposed on the previous stage ie the 'build the prototype system' stage. Ones the system is deployed, usage begins.

d. **Deliver System**

   The system is delivered to the user and comments and inputs taken into consideration for modification.

e. Make Additional Modification

As inputs are made additional modification takes place on the system and this modification is seen by the user.

f. Modify System

If the system is adequate, the developer still has the system at his or her disposal for further modification in the future, as the need arises. This modification, range from the overall look of the system to system capacity and addition of modules not previously taken into consideration.

3.3 TIRMP App Design Tools (Applications and Devices)

The TIRMP app though similar to a conventional software or portal, it has it unique features as it runs on the cloud and the design technique is little different. Some of the design tools used to develop the TIRMP app includes: force.com cloud computing platform, Microsoft visio, internet modem, compute system.

a. **Force.com Cloud Computing Platform:** is a platform build basically for application developers on the cloud.

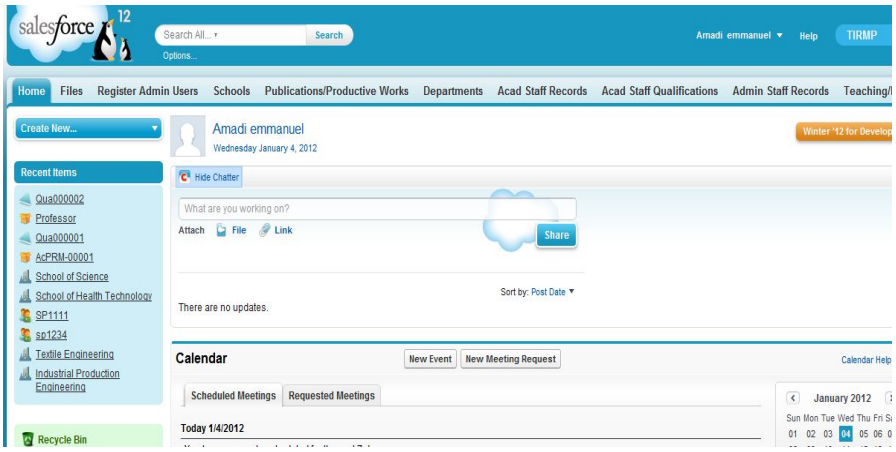


Figure 3.6: The force.com development platform

b. **Microsoft Visio:** is software that comes with the Microsoft office package. It is used basically for drawings such as network models and ERD diagrams.

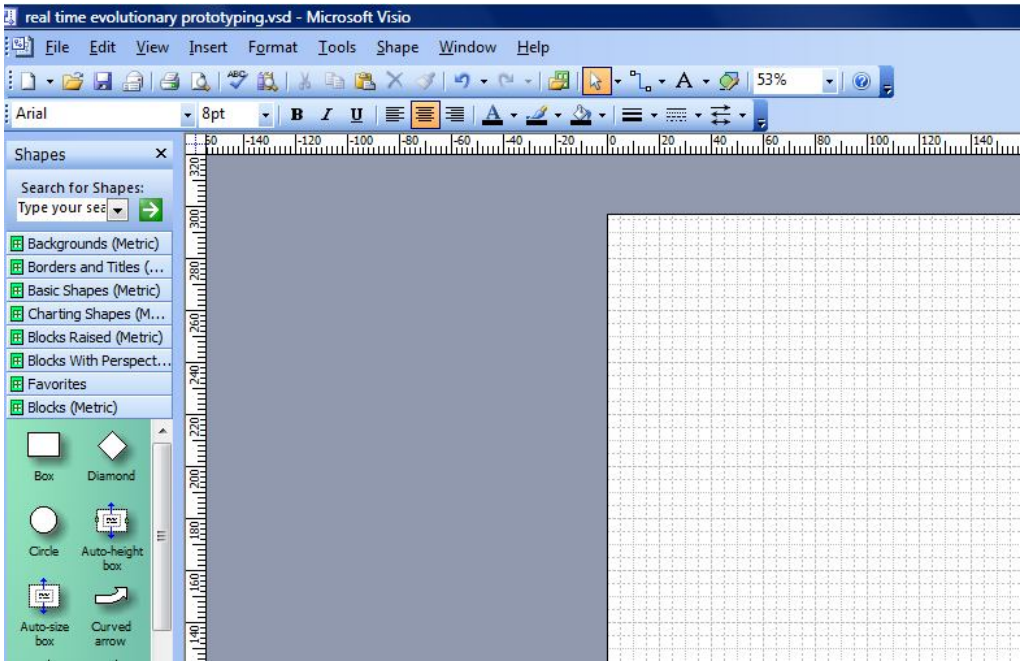


Figure 3.7: The Microsoft vision Development Environment

c. **Internet Modem:** It is used to connect the computer to the internet for development on the cloud platform. All designs are done on the cloud platform deployed thus the need for a steady and efficient internet connection.

Figure 3.8: An Internet Modem

d. **The Computer System:** the computer is key in the design of the TIRM app. without the computer system nothing can be done at all. The computer is required to be able to connect to the internet. The storage capacity is of little important as all design does not reside on the physical computer system but is hosted directly on the cloud distributed server system.

3.4 TIRMP Campus Area Network Design Tool (CAD)

One key recommendation for the smooth running of the TIRMP app is the availability of a reliable and relatively fast internet network for the organizations using the app. This design is done using Computer aided design (CAD) software called packet tracer. This software was designed by CISCO and is suitable for network design and simulation/testing.

CHAPTER FOUR

4.0 TIRMP SYSTEM DESIGN AND ANALYSIS

The goal of this chapter is to show how to build a complete TIRMP system for any institution in Nigeria. It is made up of two design areas highlighted below.

Firstly it involves creating a robust cost effective and fast network infrastructure for internet access called the campus area network (CAN). This aspect of the design is done on a CAD package called packet tracer.

Secondly, it deals with the basic steps in using the salesforce cloud computing platform to develop the user friendly platform for resource management. This last process involves creating tables, fields from the entity relationship diagram (ERD) diagram and interlinking these tables and fields based on information flow. It also involves implementing roles, privileges; roll up summaries and security on the platform.

Like many sectors of the economy that have grown rapidly, the educational sector is currently experiencing a few growth pains, especially in it resource management (human and material). This thesis seeks to provide for institutions an efficient platform for resource management allowing it to move away from the traditional Microsoft word document and Microsoft excel spreadsheets that it has traditionally used to an application that is available on demand.

4.1 Considerations for the TIRMP -TIPMP App

This platform is called the “Tertiary Institution Resource Management Platform” (TIRMP). This design focuses on the Tertiary Institutions Personnel Management Platform (TIPMP). The major objectives of this work are as follows:

1. Development of a database structure for the four key resources within the academic institutions in Nigeria namely: Staff, Student, Books/publication, and equipment. This structure can adopt to any platform for development
2. Virtualization of the management of both human and material resources within tertiary institutions using cloud computing. This will provide for easy access, retrieval and manipulation. The force.com cloud platform will be used in this work.
3. Implement security roles, privileges and authentication within the cloud platform to ensure data security and integrity.
4. This work also provides a proposal for developing a robust, cost effective and efficient internet infrastructure design with a scalable bandwidth adjustment table. This design can be adopted by any institution in Nigeria.

The following are the design considerations put forward for the TIRMP. The system should be able to perform the following functions:

- Track all the personnel records in the institution both academic and non academic
- Track staff publications, books, research works at various levels.
- Track teaching and professional experience of staffs and their growth in the institution for the purpose of appraisal for promotion.
- Allow for the appraisal of staff members at various levels
- Track available vacancy within the institution and provide a list of possible candidates to fill those vacancies
- Keep track of employment status of staff members provide such information to the staff members in real time.
- Inform staff members of necessary steps to take as regards their growth within the institution.
- Automate the posting of vacancies to the institutions website.

4.2 TIRMP Complete System Structure (Including the CAN)

The TIRMP complete system is as shown in figure 4.1 below. The system consist of campus area network (CAN), connecting to the internet cloud (salesforce) via the Internet service provider (ISP). The CAN consist of 4 basic section as illustrated in the diagram; the internal user, the access layer, the distribution layer, the core layer and the gateway. Each of these subsystems will be discussed in more detail latter on in this chapter. Various cloud computing platforms run on the internet, the salesforce platform runs the TIRMP. Both internal and external users can access the platform with no restriction to location.

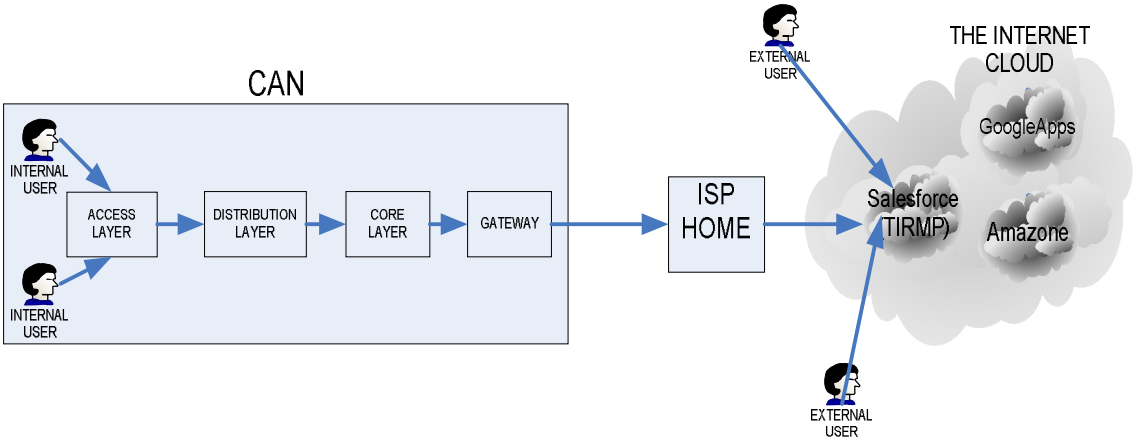


Figure 4.1: The TIRMP complete system

4.3 The Campus Area Network (CAN)

The campus area network (CAN) is made up of four different segments as illustrated in figure 4.1. The segments include: the access layer (also includes the internal users), the distribution layer, the core layer and the gateway. The CAN designed using the packet tracer 5.3 software developed by Cisco Corporation.

4.3.1 The Network Gateway

A gateway is a network point that acts as an entrance to another network. For example, the network device that control traffic between organizations networks or the device used by internet service providers (ISPs) to connect users to the internet are gateway nodes. On the

campus area network, the gateway device is a layer three switch. This switch connects to the ISP home on one end and connects to the CAN core layer on the other end. This is illustrated in figure 4.2.

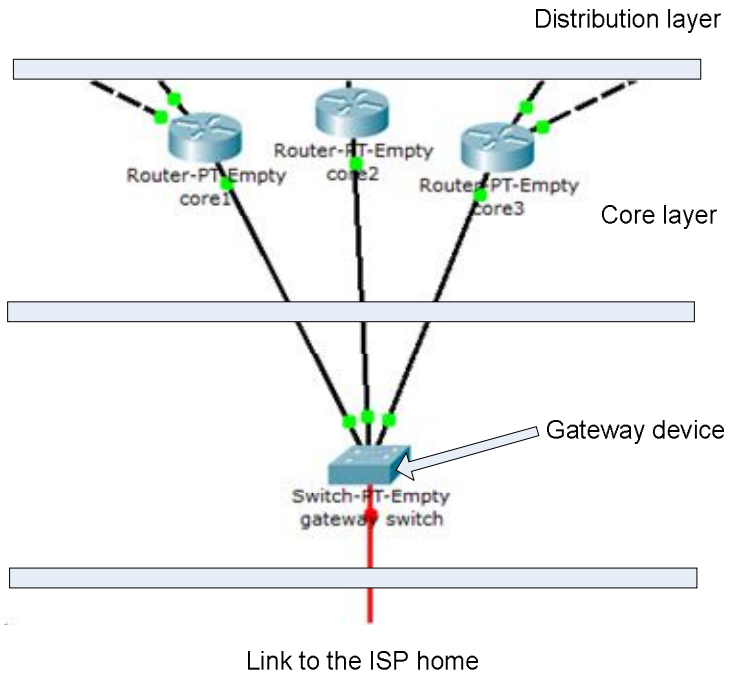


Figure 4.2: The Gateway and the Core Layer of the CAN

4.3.2 The Core Layer

The core layer is literally the internet backbone, the simplest yet most critical layer. The primary purpose of the core is to provide fault isolation and backbone connectivity; in other words, the core is highly reliable and switches traffic as fast as possible. Therefore, on one hand, the core provide the appropriate level of redundancy to allow fault tolerance in case of hardware or software failure or upgrade; and on the other hand, the high-end switches and high-speed cables are implemented to achieve High data transfer rate and Low latency period.

The core layer as depicted in figure 4.2 is made up of routers that are linked to the gateway switch on one side and the distribution layer on the other side. The core layer serves as

the main stream of the network. It connects to the distribution layer switches, the mail server and the Hypertext Transfer Protocol (HTTP) server all on the distribution layer.

4.3.3 The Distribution Layer

The distribution layer acts as an interface between the access layer and the core layer. The primary function of the distribution layer is to provide routing, filtering, and WAN access and to determine how packets can access the core, if.

While core layer and access layer are special purpose layers, the distribution layer on the other hand serves multiple purposes. It is an aggregation point for all of the access layer switches and also participates in the core routing design. This layer includes LAN-based routers and OSI layer 3 switches. It ensures that packets are properly routed between subnets and virtual Local Area Networks (VLANs).

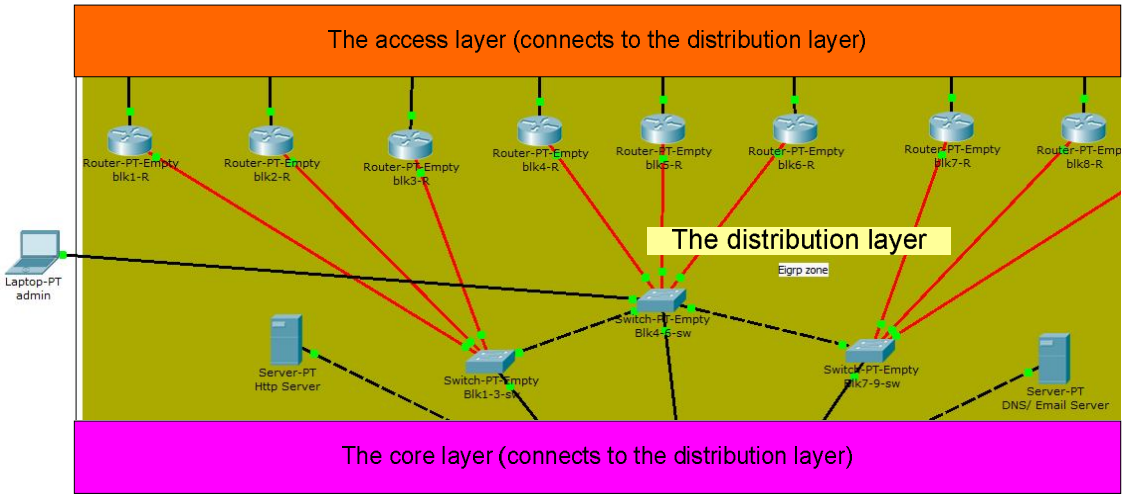


Figure 4.3: The Distribution Layer of the CAN

4.3.4 The Access Layer

The access layer is sometimes referred to as the desktop layer. The network resources the workgroup and users needed will be available locally.

The access layer is the edge of the entire network, where a wide variety of types of consumer devices such as PCs, printers, cameras attach to the wired portion of the network, various services are provided, and dynamic configuration mechanisms like the Dynamic Host Configuration Protocol (DHCP) is implemented. As a result, the access layer is most feature-rich layer of the three-layered model. The users of the network shown on figure 4.4 operate on the access layer.

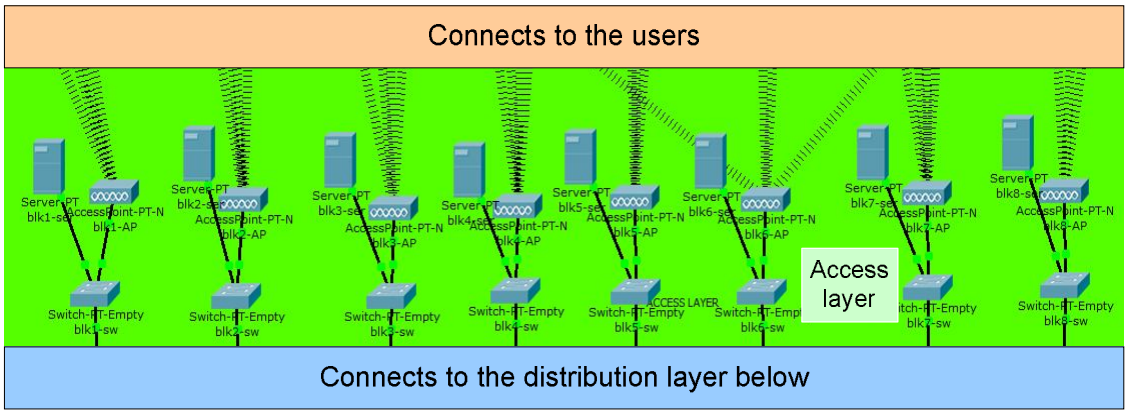


Figure 4.4: The Access Layer of the CAN

4.4 Database Implementation

The TIRMP database is made up of tables (objects). Figure 4.5 is a screen shot of the list of tables used for the TIRMP on the salesforce platform. Each table is made up of fields (attributes) that represent different parameter to be collected within the educational institution environment. These objects and fields are discussed in more detail in the next section.

New Custom Object | Schema Build

Action	Label	Master Object	Deployed
Edit | Del	Acad Staff Appraisal	Acad Staff Promotion Form	✓
Edit | Del	Acad Staff Promotion Form	Acad Staff Record	✓
Edit | Del	Acad Staff Qualification	Acad Staff Record	✓
Edit | Del	Acad Staff Record		✓
Edit | Del	Admin Staff Appraisal		✓
Edit | Del	Admin Staff Promotion Form		✓
Edit | Del	Admin Staff Record		✓
Edit | Del	Department		✓
Edit | Del	Publication/Productive Works		✓
Edit | Del	Register Admin User		✓
Edit | Del	Register Position		✓
Edit | Del	School		✓
Edit | Del	Teaching/Professional Experiences		✓
Edit | Del	Unit		✓

Figure 4.5: The objects (tables) used for the TIRMP database design

4.4.1 Entity Relationship Diagram (ERD)

Data modelling is the process of creating a logical representation of the structure of a database. It is the most important task performed by the developer. The model identifies the things to be stored in the database and define their structure and the relationships among them. The data model leads to the implementation of the database. The two very important models used in database design are the entity relationship model and the relational model. This section focuses on the entity relationship model used to develop the database structure for the TIRMP platform. The ERD is divided into 2 categories. The Academic staff ERD and the Admin staff ERD.

a. Academic Staff ERD Diagram

The academic staff category is made up of eight tables as illustrated in figure 4.6. The ERD was developed using the Microsoft Vision Platform optimised for the design of ERD diagrams. A screen shot of the Visio platform is shown in figure 4.6

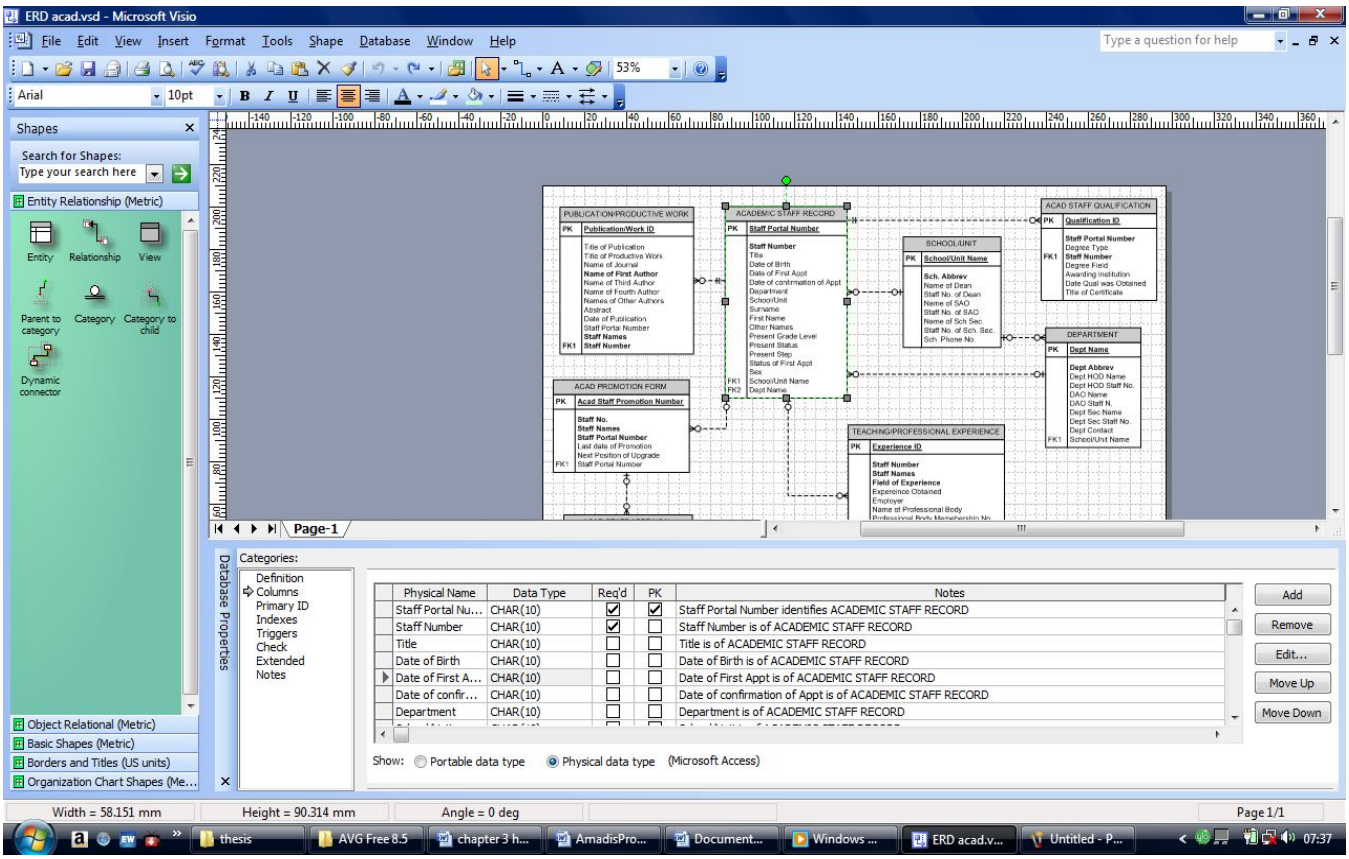


Figure 4.6: Screen shot of the Microsoft Visio Design Platform

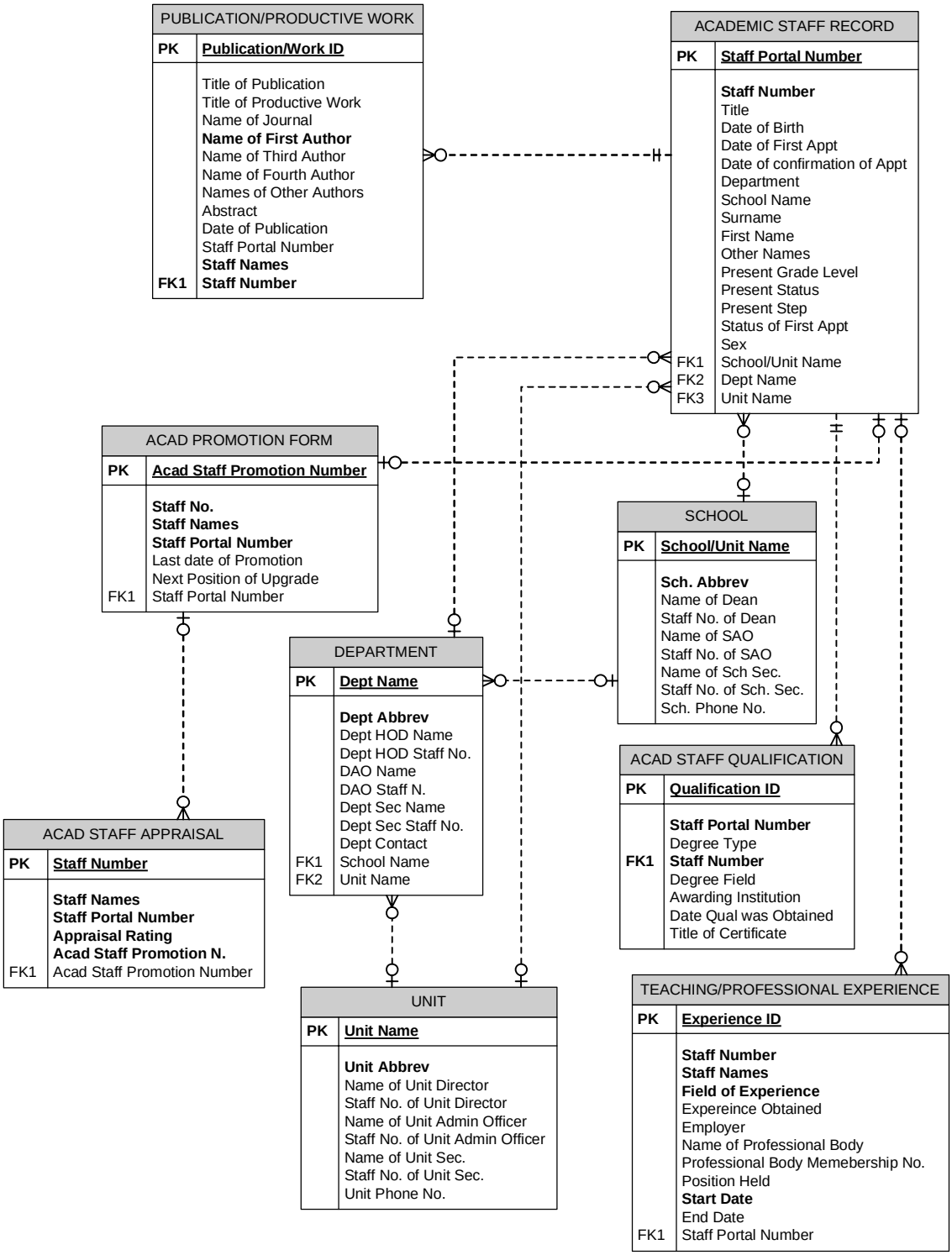


Figure 4.7: The entity relationship diagram of the academic staff category of the TIRMP (Source: Author, 2012)

b. Admin Staff ERD Diagram

The admin staff category of the TIRMP is made up of eight tables as illustrated in figure 4-8 below.

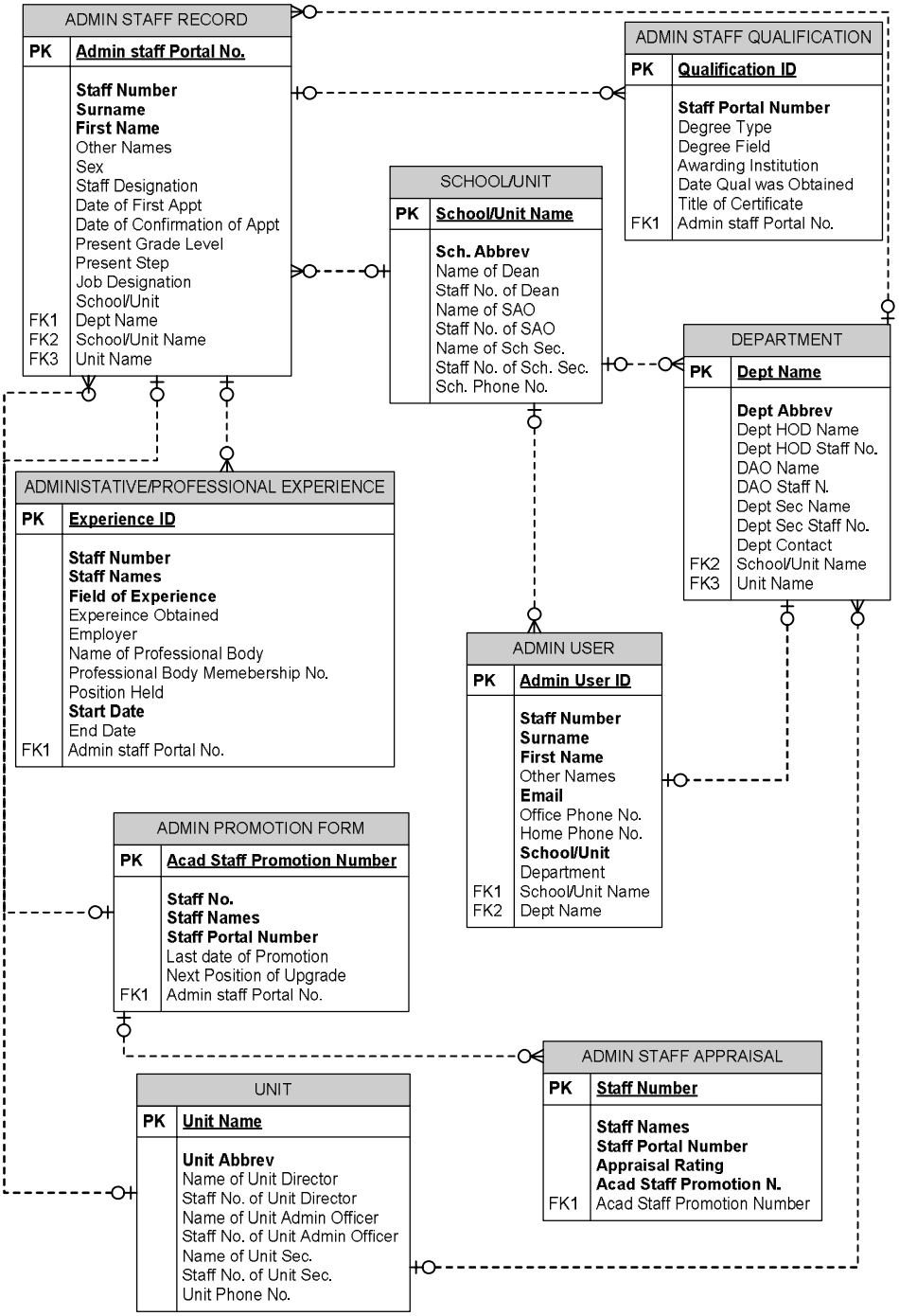


Figure 4.8: Entity Relationship Diagram of Admin Staff Category (Source: Author, 2012))

4.5 Platform Development

The platform development was done using the force.com cloud computing platform. The Platform development is strictly an online developmental process. This is one of the great advantages of using cloud computing as applications are deployed as soon as they are being developed. Thus from the start of the development process testing can begin and development can be done on the fly with no restriction to location or design gadgets like the laptops. Another great advantage provided by the platform for development it that collaboration on the design of a particular application by several users with their separate task integrated into the overall system. This development process is coordinated by the lead design administrator.

4.5.1 Flow Charts Illustrating Major Activities Performed on the TIRMP

The following sections illustrate in a flow chart the activities that can be performed on personnel section of the TIRMP platform. The activities include

- Register new user
- User login
- Appraisal/ promotion

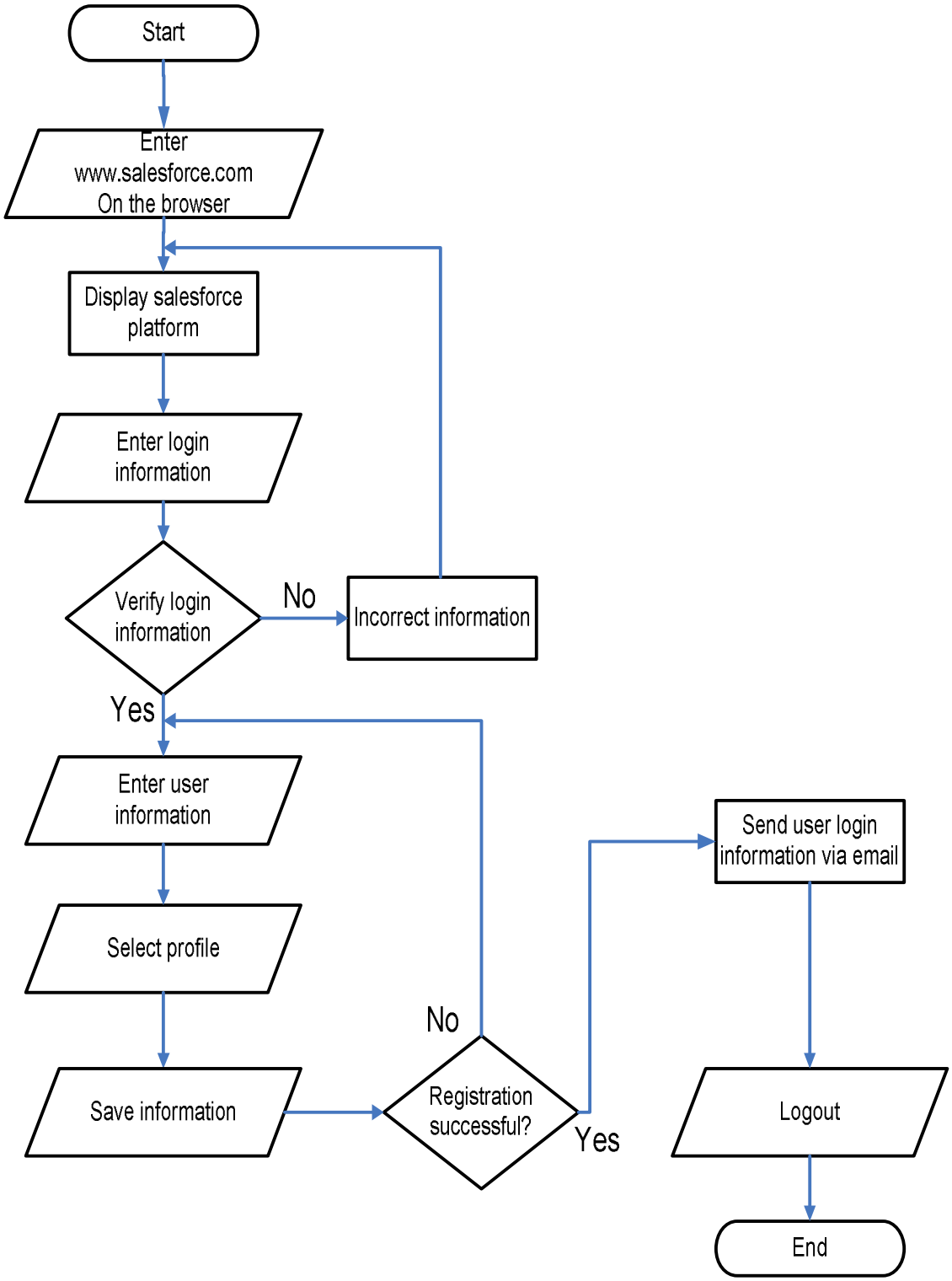


Figure 4.9: Register new user Flow Chart

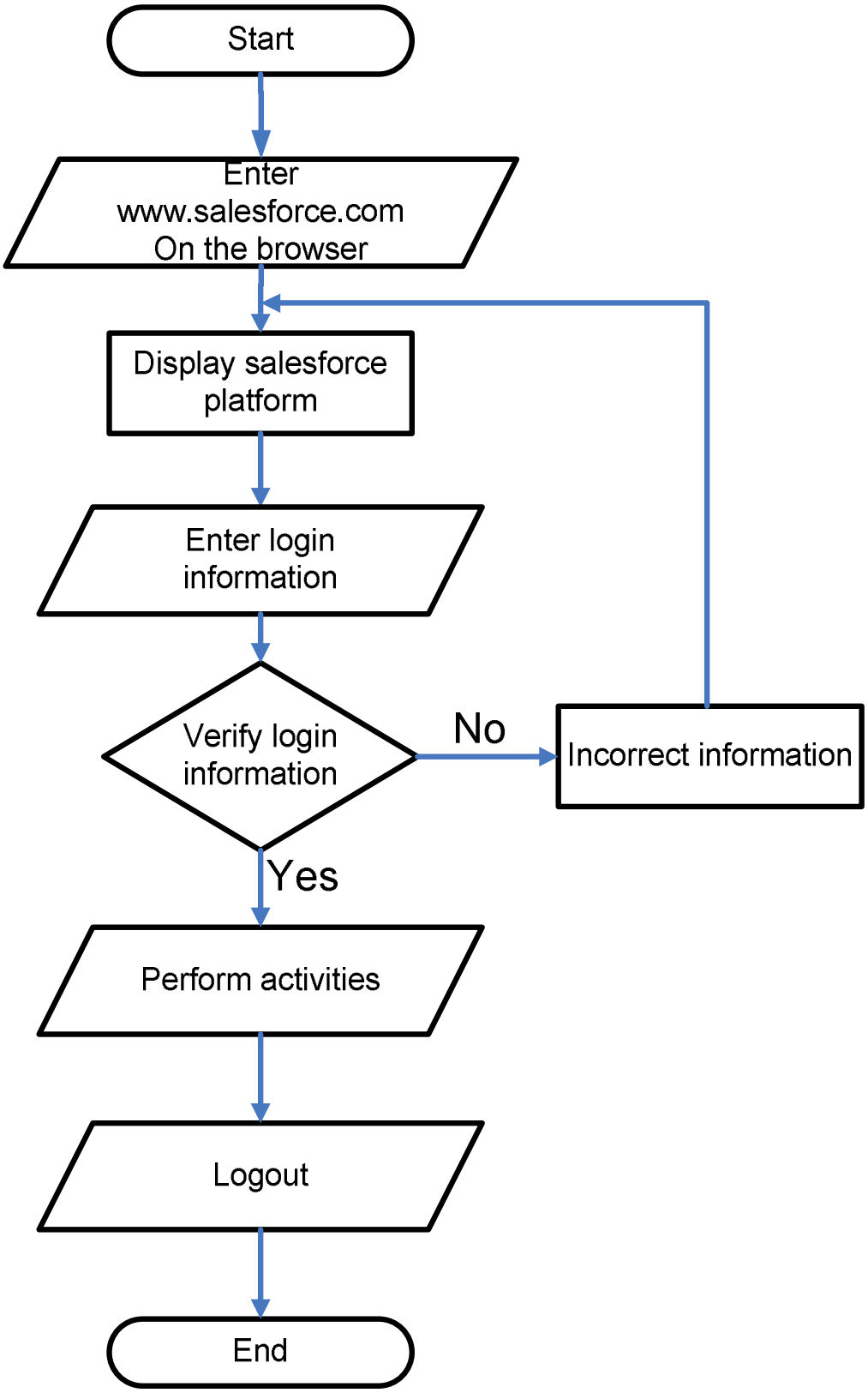


Figure 4.10: User login flow chart

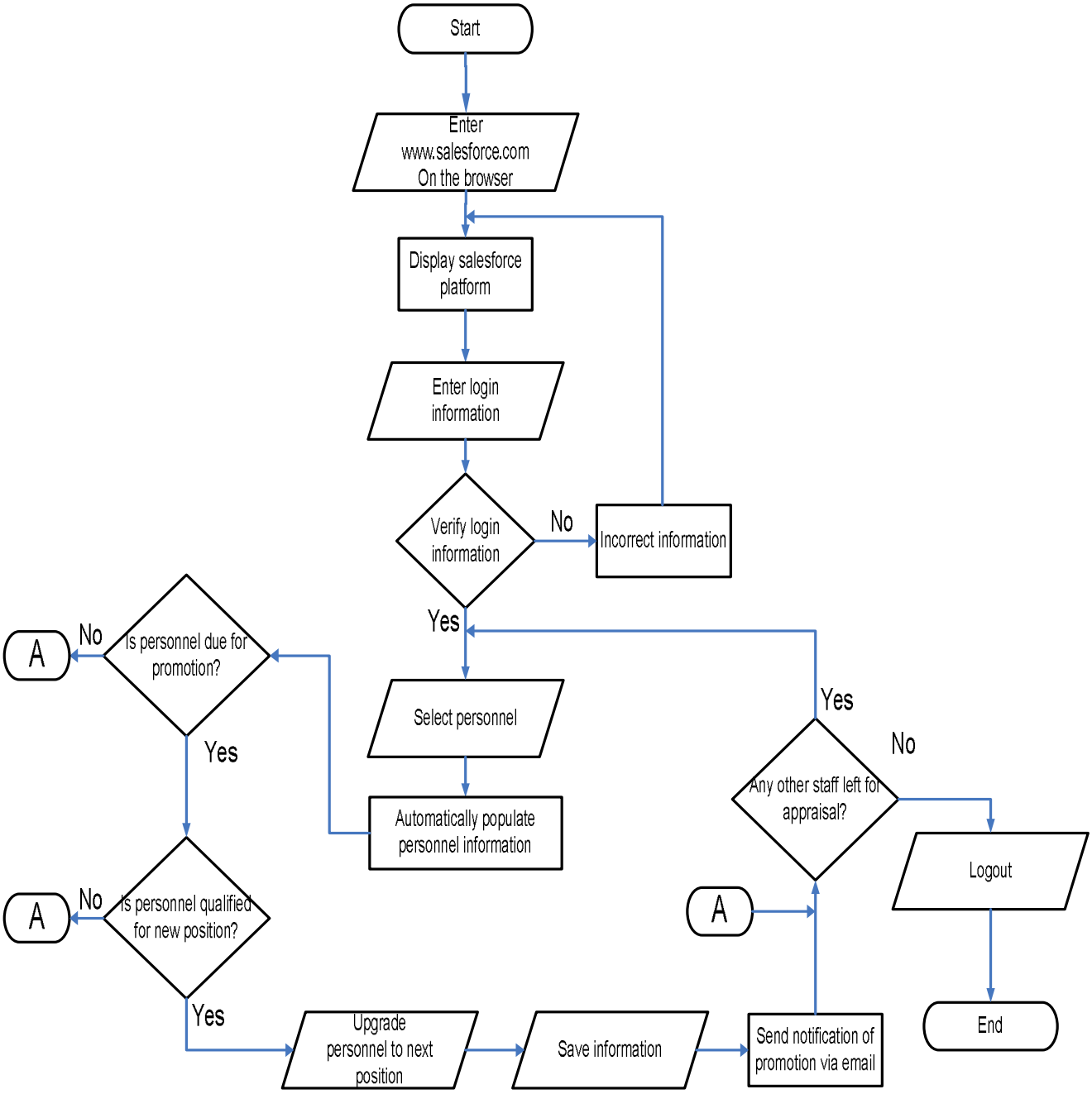


Figure 4.11: Personnel Appraisal/Promotion Flow Chart

4.5.2 Registering as a Developer on the Force.com Platform

Before one can make use of the force.com platform, registration as a user is required. To register as a developer on the platform, go to http://developer.force.com/ the following screen will be displayed for you to create a free user account on the platform.

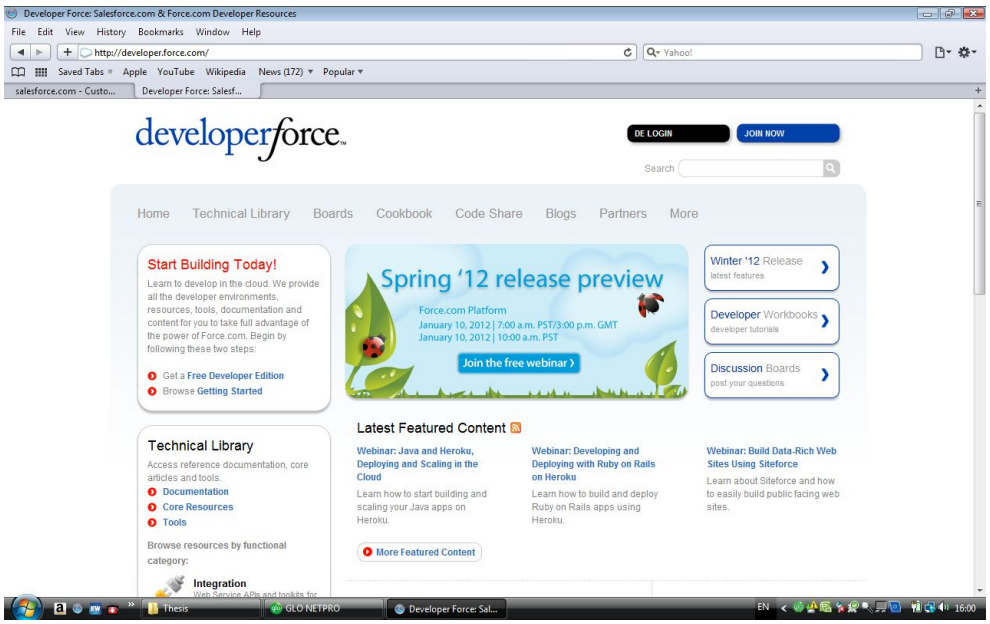


Figure 4.12: Force.com home page.

The developer force is a community of developers who customize and build applications that run in the cloud and are built with the force.com platform. Developer Force members have access to a full range of resources, including sample code, toolkits, an online developer community, and the test environments necessary for building applications.

4.5.3 The Setup Area

The setup area is the user preference area, an application building and customization environment, and an administration tool, all in one. We perform almost every task we need to create our application in the setup area.

After registering as a user, using your email address and password, you simply login to the platform using the following address https://login.salesforce.com/. Figure 3.13 show the setup area displayed after login.

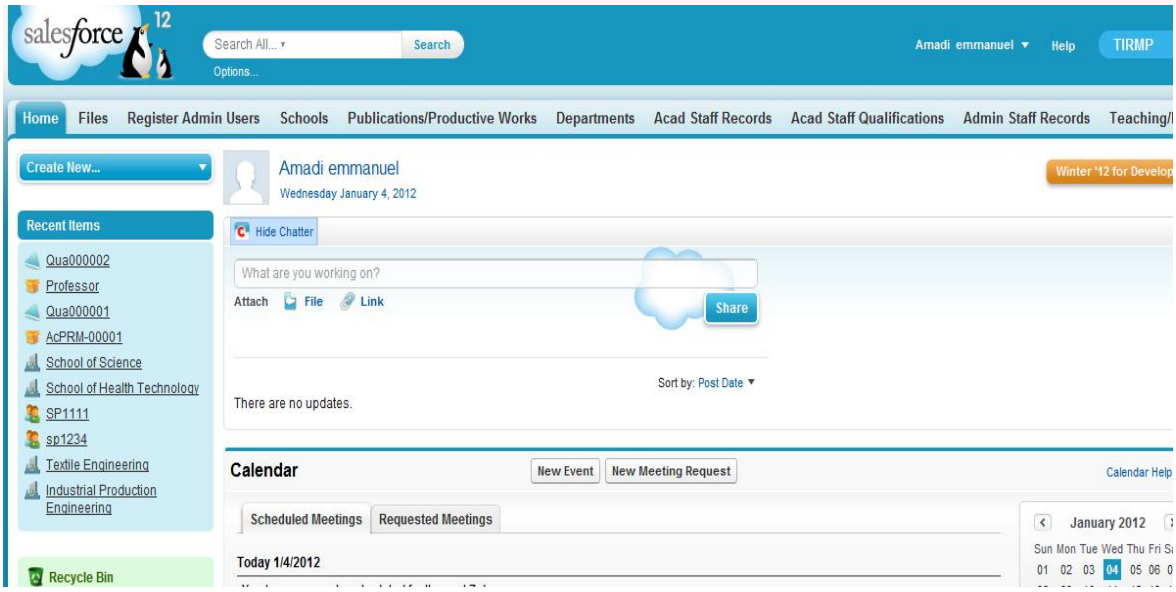


Figure 4.13: The salesforce setup area of the force.com platform (home page)

4.5.4 Configuring the TIRMP Application

The TIRMP application is created from setup using the following steps

1. login as a user using your email and password as discussed in the earlier section
2. Click ***your name – setup – create – Apps.***
3. If you see a splash page, simply click ***continue.***
4. Click ***New***. The New custom App wizard appears
5. In the *Apps Label* field, Enter **TIRMP**

The Apps label is the name that will represent the new application in the force.com app menu that appears. Below is a screen shot that appears and you simply follow the prompts

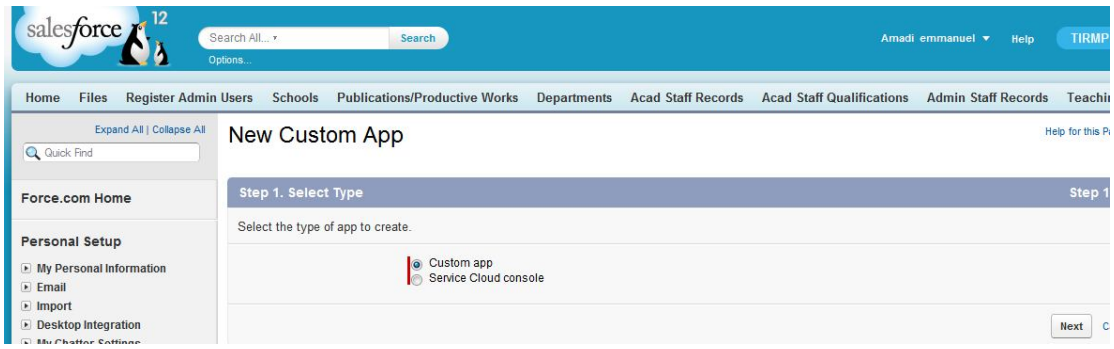


Figure 4.14: Creating new application on the salesforce environment.

4.5.5 Introducing Objects (Tables)

This section will illustrate how a table is created using the first object i.e the Acad Staff Record and its associated field. The object represents a table on the ERD diagram.

To create the follow the following steps:

1. Click your ***Name – Setup – Create – Objects***
2. On the *Custom Objects page*, click ***New Custom Object***
3. In the *Label Field*, enter *Acad Staff Record*
4. In the *Plural Label Field* enter *Acad Staff Records*
5. The *Object Name Field* is defaulted to *Acad Staff Record*.
6. In the *Description Field*, enter *This object stored information about each academic staff of the institution*.
7. For the *Context – Sensitive Help Setting*, accept the default.
8. In the *Record Name* field, enter *Staff Portal ID*
9. In the *Data Type* drop-down list, select *Text*
10. In the *Optional Feature* are, select *Allow Reports, Allow Activities, and Track Fields History* checkboxes
11. In the *Deployment Status* area, select *Deployed*
12. In the *Object Creation Option* area, select the *Add Notes and Attachments related list to default page layout* and *launch New Custom Tab Wizard after saving this custom object* checkbox.
13. Click save

The table below shows a list of tables created for the TIRMP app for both the admin and academic categories indicated on the ERD diagrams of figure 4.7 and figure 4.8.

Table 4.1: list of Objects on the TIRMP App.

1	Publication/productive work
2	Academic staff record
3	Academic promotion form
4	School
5	Department
6	Academic staff qualification
7	Academic staff appraisal
8	Unit
9	Teaching/professional experience
10	Admin staff record
11	Admin staff qualification
12	Administrative/professional qualification
13	Admin user
14	Admin staff appraisal
15	Admin promotion form

(Source, Author)

4.5.6 Introducing Tabs

After the table has been created tabs are created to enable easy navigation on the platform. Every tab serves as the starting point for viewing, editing, and entering information for a particular object. When you click at the top of the page, the corresponding home page for that object appears. To create a custom tab for our *Acad Staff Record*, we make use of the new custom object tab wizard that is launched after clicking save after defining the object above. The following steps are used to create the Acad Staff Record Tab of the TIRMP app for illustration purpose. Note that the School, Unit and Department objects are created first as they are key lookup fields in the TIRMP app.

1. Click your Name – Setup – Create – Tab
2. In the custom object tabs area, click New
3. In the object drop-down list, select Acad Staff Record

4. Click the Tab style lookup icon to launch the Tab style selectors and select a tab style
5. Following the prompts and select all defaults
6. Click save

4.5.7 Introducing Fields

The fields are like a database column. The primary characteristic of a field is its data type. Some fields hold text values while other hold currency values percentage, phone number, email addresses, or dates. Some field look like checkboxes, while still others are drop-down lists or records lookup form which a user makes a selection.

The data type of field of a field controls the way the field is ultimately displayed in the user interface and how data entered into the field is stored in the platform. To add the Staff Name fields on the Acad Staff Record object we follow the following steps:

1. Click your ***Name – Setup – Create – Object***
2. Click ***Acad Staff Record***
3. In the *Custom Field and Relationships* related list, click ***New***
4. Choose the *Text Area (Long)* data type and click ***Next***
5. In the *Field Label* field, enter *First Name*
6. In the *Length* field, enter 30
7. In the # *Visible Lines* field Enter *3*
8. In the *Description and Help Text* fields, enter *High-Level description of each staff member*
9. Click ***Next***
10. Click ***Save*** and ***New***

In creating the field, the primary keys are not included as these are already created while creating the object in the previous two sections (creating objects). These fields are

automatically included in the list of fields on the object page. Table 3-2 shows the list of fields to be created using the

Table 4.2: List of fields in the Academic Staff Record Object

Field Type	Field Label	Length	Pick list Values	Default Value
Text	Staff Number	30	-	Leave unspecified
Pick list	Title	-	Mr, Mrs, Miss, Engr, Dr, Prof	Leave unspecified
Date	Date of Birth	-	-	Leave unspecified
Date	Date of First Appointment	-	-	Leave unspecified
Date	Date of Confirmation of Appt.	-	-	Leave unspecified
Lookup	School Name	-	-	Leave unspecified
Text	Surname	30	-	Leave unspecified
Text	First Name	30	-	Leave unspecified
Text	Other Names	100	-	Leave unspecified
Pick list	Present Grade Level	-	1,2,3,4,5,6,7	None
Pick list	Present Status	-	Temporary, Full Time, Contract, Adjunct	Temporary
Pick list	Present Step	-	1,2,3,4,5,6,7	1
Pick list	Sex	-	Male, Female	Male
Lookup	Dept Name	-		Leave unspecified
Lookup	Unit Name	-		Leave unspecified

(Source: Author, 2012)

The following screen shot shows a complete page of the Academic Staff Record page developed using table 3.2 and following steps 1 to 10.

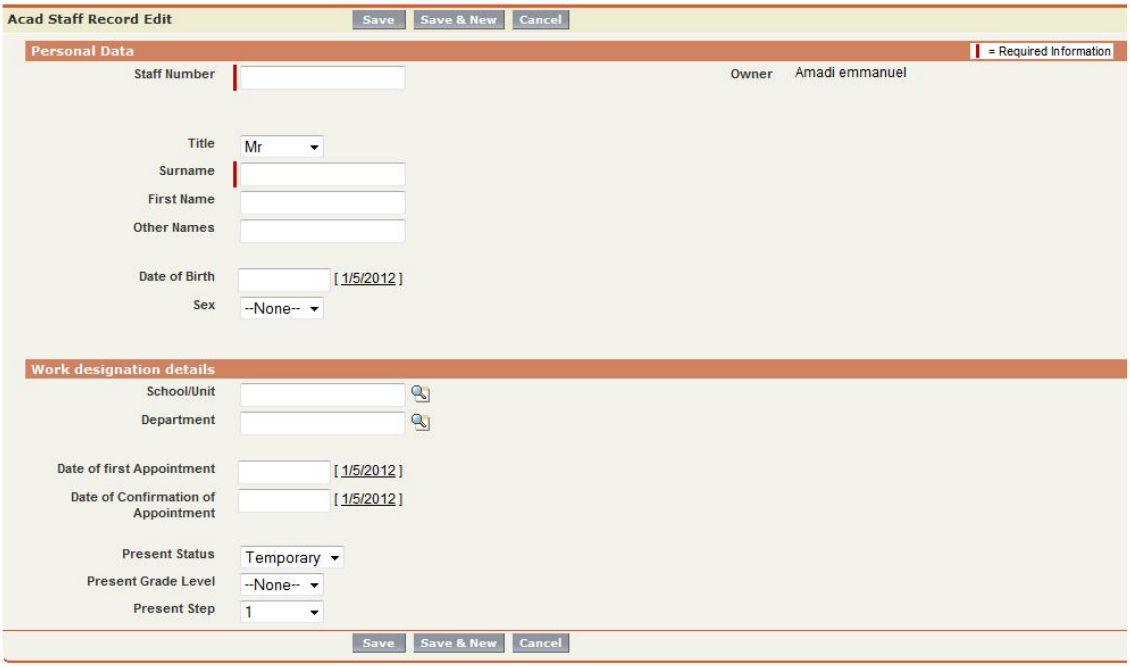


Figure 4.15: The Academic Staff Record Page

Tables 4.3 – 4.16 below show the various objects created on the TIRMP. The fields are created using steps 1 to 10 above of this section. Each table can be modified to suit a particular purpose or data type.

Table 4.3: List of fields in the Academic Publication/Productive Works object

Field Type	Field Label	Length	Pick list Values	Default Value
Text	Title of publication	50	-	Unspecified
Text	Title of productive work	50	-	Unspecified
Text	Name of journal	50	-	Unspecified
Text	Name of first author	30	-	Unspecified
Text	Name of second author	30	-	Unspecified
Text	Name of third author	30	-	Unspecified
Text	Name of fourth author	30	-	Unspecified
Text area	Names of other authors	255	-	Unspecified
Long text area	Abstract	32768	-	Unspecified
Date	Date of publication	-	-	Unspecified
Lookup	Staff portal number	-	-	-
Lookup	Staff number	-	-	-

(Source: Author, 2012)

Table 4.4: List of Fields in the Academic Promotion Form

Field Type	Field Label	Length	Pick list Values	Default Value
Lookup	Staff number	-	-	Unspecified
Auto (Lookup)	Staff names	-	-	Unspecified
Auto (Lookup)	Staff portal number	-	-	Unspecified
Date	Last date of promotion	-	-	Unspecified
Pick list	Next position of upgrade	-		-
Auto (Lookup)	Staff portal number	-	-	Unspecified

(Source: Author, 2012)

Table 4.5: List of fields in the Academic Staff Appraisal Form

Field Type	Field Label	Length	Pick list Values	Default Value
Lookup	Staff number	-	-	Unspecified
Auto (Lookup)	Staff names	-	-	Unspecified
Auto (Lookup)	Staff portal number	-	-	Unspecified
Date	Last date of promotion	-	-	Unspecified
Pick list	Appraisal rating	-	1,2,3,4,5	Unspecified
Auto (Lookup)	Acad staff promotion no.	-	-	Unspecified

(Source: Author, 2012)

Table 4.6: List of Fields in the Academic Staff Qualification Object

Field Type	Field Label	Length	Pick list Values	Default Value
Lookup	Staff number	-	-	Unspecified
Auto (Lookup)	Staff names	-	-	Unspecified
Auto (Lookup)	Staff portal number	-	-	Unspecified
Pick list	Degree type	-	B-Eng, Msc, Phd etc.	Unspecified
Pick list	Awarding institution	-	List of schools	Unspecified
Date	Date qualification was obtained	-	-	Unspecified
Text	Title of certificate	-	-	Unspecified
Pick list	Degree field	-	-	Unspecified

(Source: Author, 2012)

Table 4.7: List of Fields in the Teaching/Professional Experience Object

Field Type	Field Label	Length	Pick list Values	Default Value
Lookup	Staff number	-	-	Unspecified
Auto (Lookup)	Staff names	-	-	Unspecified
Auto (Lookup)	Staff portal number	-	-	Unspecified
Text	Field of experience	100	-	Unspecified
Long text area	Experience obtained	32,000	-	Unspecified
Text	Employer	100	-	Unspecified
Text	Name of professional body	100	-	Unspecified
Text	Professional body membership no.	100	-	Unspecified
Text area	Position held	255	-	Unspecified
Date	Start date	-	-	Unspecified
Date	End date	-	-	Unspecified

(Source: Author, 2012)

Table 4.8: List of Fields in the Register School Object

Field Type	Field Label	Length	Pick list Values	Default Value
Text	School abbreviation	3	-	Unspecified
Lookup	Name of dean	-	-	Unspecified
Auto (Lookup)	Staff no of dean	-	-	Unspecified
Lookup	Name of SAO	-	-	Unspecified
Auto (lookup)	Staff no of SAO	-	-	Unspecified
Lookup	Name of school sec.	-	-	Unspecified
Auto (lookup)	Staff no of school sec.	-	-	Unspecified
Number	School phone no.	11	-	Unspecified

(Source: Author, 2012)

Table 4.9: List of Fields in the Department Object

Field Type	Field Label	Length	Pick list Values	Default Value
Text	Dept abbreviation	3	-	Unspecified
Lookup	Name of HOD	-	-	Unspecified
Auto (Lookup)	Staff no of HOD	-	-	Unspecified
Lookup	Name of DAO	-	-	Unspecified
Auto (lookup)	Staff no of DAO	-	-	Unspecified
Lookup	Name of dept sec.	-	-	Unspecified
Auto (lookup)	Staff no of dept sec.	-	-	Unspecified
Number	Dept phone no.	11	-	Unspecified

Table 4.10: List of Fields in the Create new Unit Object

Field Type	Field Label	Length	Pick list Values	Default Value
Text	Unit abbreviation	3	-	Unspecified
Lookup	Name of unit coordinator	-	-	Unspecified
Auto (Lookup)	Staff no of unit coordinator	-	-	Unspecified
Lookup	Name of unit admin officer	-	-	Unspecified
Auto (lookup)	Staff no of unit admin officer	-	-	Unspecified
Lookup	Name of unit sec.	-	-	Unspecified
Auto (lookup)	Staff no of unit sec.	-	-	Unspecified
Number	Unit phone no.	11	-	Unspecified

(Source: Author, 2012)

Table 4.11: List of Fields in the Admin Staff Record Object

Field Type	Field Label	Length	Pick list Values	Default Value
Text	Staff Number	30	-	Leave unspecified
Pick list	Title	-	Mr, Mrs, Miss, Engr, Dr, Prof	Leave unspecified
Date	Date of Birth	-	-	Leave unspecified
Date	Date of First Appointment	-	-	Leave unspecified
Date	Date of Confirmation of Appt.	-	-	Leave unspecified
Lookup	School Name	-	-	Leave unspecified
Text	Surname	30	-	Leave unspecified
Text	First Name	30	-	Leave unspecified
Text	Other Names	100	-	Leave unspecified
Pick list	Present Grade Level	-	1,2,3,4,5,6,7	None
Pick list	Present Status	-	Temporary, Full Time, Contract, Adjunct	Temporary
Pick list	Present Step	-	1,2,3,4,5,6,7	1
Pick list	Sex	-	Male, Female	Male
Lookup	Dept Name	-		Leave unspecified
Lookup	Unit Name	-		Leave unspecified
Pick list	Job Description	-	List of job description available	

(Source: Author, 2012)

Table 4.12: List of Fields in the Admin Staff Appraisal Object

Field Type	Field Label	Length	Pick list Values	Default Value
Lookup	Staff number	-	-	Unspecified
Auto (Lookup)	Staff names	-	-	Unspecified
Auto (Lookup)	Staff portal number	-	-	Unspecified
Date	Last date of promotion	-	-	Unspecified
Pick list	Appraisal rating	-	1,2,3,4,5	Unspecified
Auto (Lookup)	Admin staff promotion no.	-	-	Unspecified

(Source: Author, 2012)

Table 4.13: List of Fields in the Admin Staff Qualification Object

Field Type	Field Label	Length	Pick list Values	Default Value
Lookup	Staff number	-	-	Unspecified
Auto (Lookup)	Staff names	-	-	Unspecified
Auto (Lookup)	Staff portal number	-	-	Unspecified
Pick list	Degree type	-	B-Eng, Msc, Phd etc.	Unspecified
Pick list	Awarding institution	-	List of schools	Unspecified
Date	Date qualification was obtained	-	-	Unspecified
Text	Title of certificate	-	-	Unspecified
Pick list	Degree field	-	-	Unspecified

Table 4.14: List of Fields in the Administrative/Professional Experience Object

Field Type	Field Label	Length	Pick list Values	Default Value
Lookup	Staff number	-	-	Unspecified
Auto (Lookup)	Staff names	-	-	Unspecified
Auto (Lookup)	Staff portal number	-	-	Unspecified
Text	Field of experience	100	-	Unspecified
Long text area	Experience obtained	32,000	-	Unspecified
Text	Employer	100	-	Unspecified
Text	Name of professional	100	-	Unspecified

	body			
Text	Professional body membership no.	100	-	Unspecified
Text area	Position held	255	-	Unspecified
Date	Start date	-	-	Unspecified
Date	End date	-	-	Unspecified

(Source: Author, 2012)

Table 4.15: List of Fields in the Admin Promotion Form

Field Type	**Field Label**	**Length**	**Pick list Values**	**Default Value**
Lookup	Staff number	-	-	Unspecified
Auto (Lookup)	Staff names	-	-	Unspecified
Auto (Lookup)	Staff portal number	-	-	Unspecified
Date	Last date of promotion	-	-	Unspecified
Pick list	Next position of upgrade	-		-
Auto (Lookup)	Staff portal number	-	-	Unspecified

Table 4.16:Llist of Fields in the Admin user Object

Field Type	**Field Label**	**Length**	**Pick list Values**	**Default Value**
Lookup	Staff number	-	-	Unspecified
Auto (Lookup)	Surname	-	-	Unspecified
Auto (Lookup)	First name	-	-	Unspecified
Auto (Lookup)	Other names	-	-	Unspecified
Email	Email	30	-	Unspecified
Number	Office phone no.	11	-	
Number	Home phone no.	11	-	Unspecified
Lookup	School name	-	-	Unspecified
Lookup	Department name	-	-	Unspecified
Lookup	Unit name	-	-	Unspecified

(Source: Author, 2012)

The following screen shots i.e figure 4.16 - 4.24 shows some of the forms created using the tables created in this section. The rows on the table represent a field on the form.

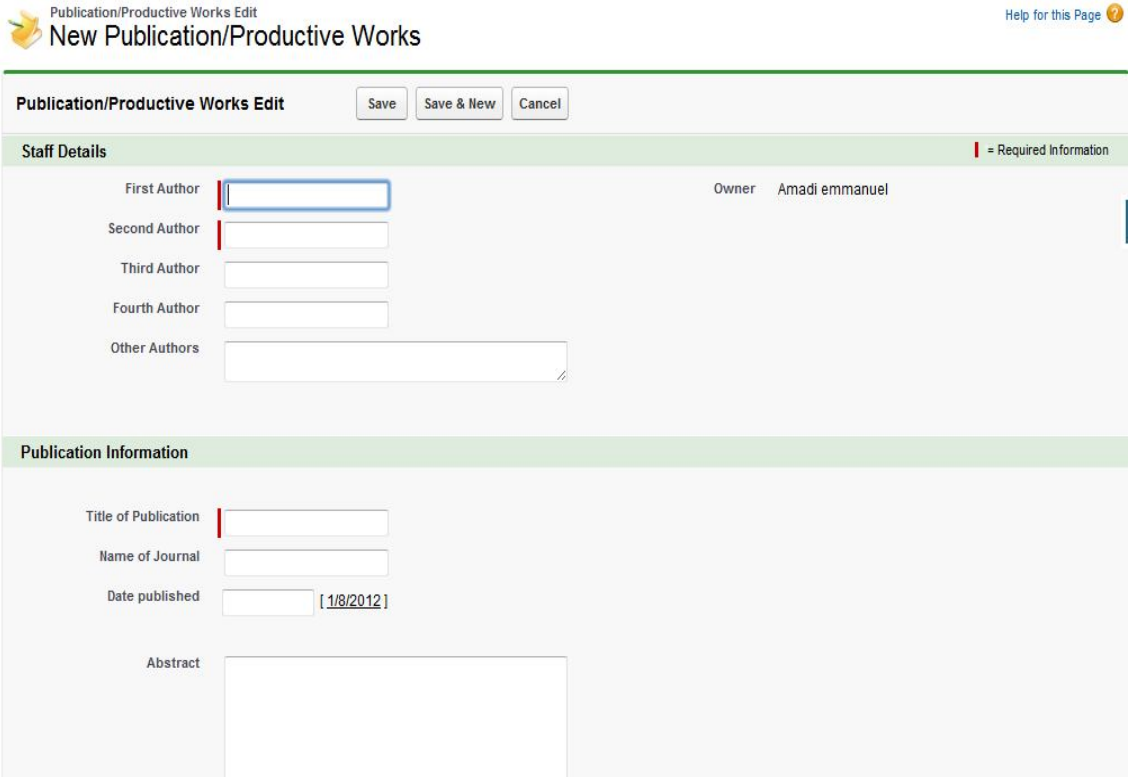


Figure 4.16: Publication/Productive work form

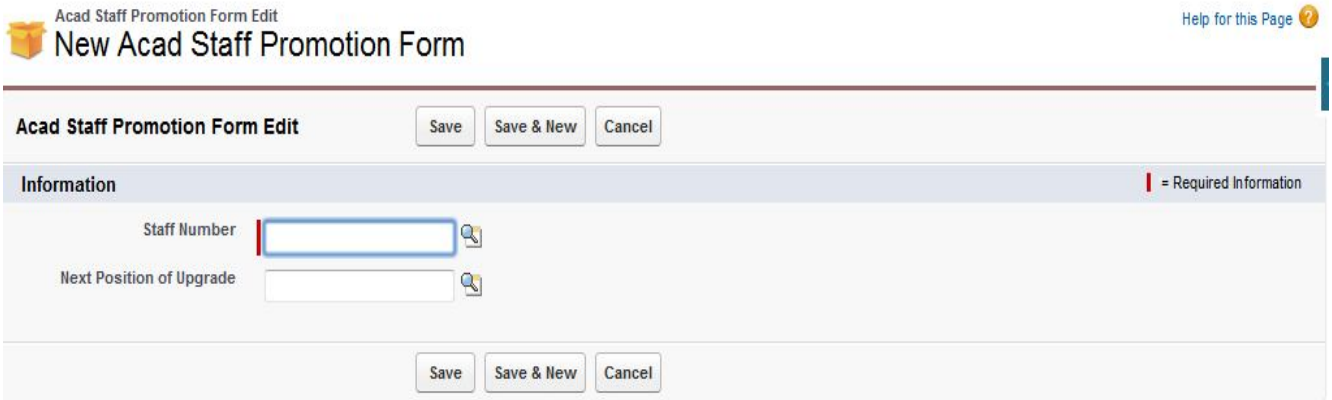


Figure 4.17: Academic Staff Promotion Form

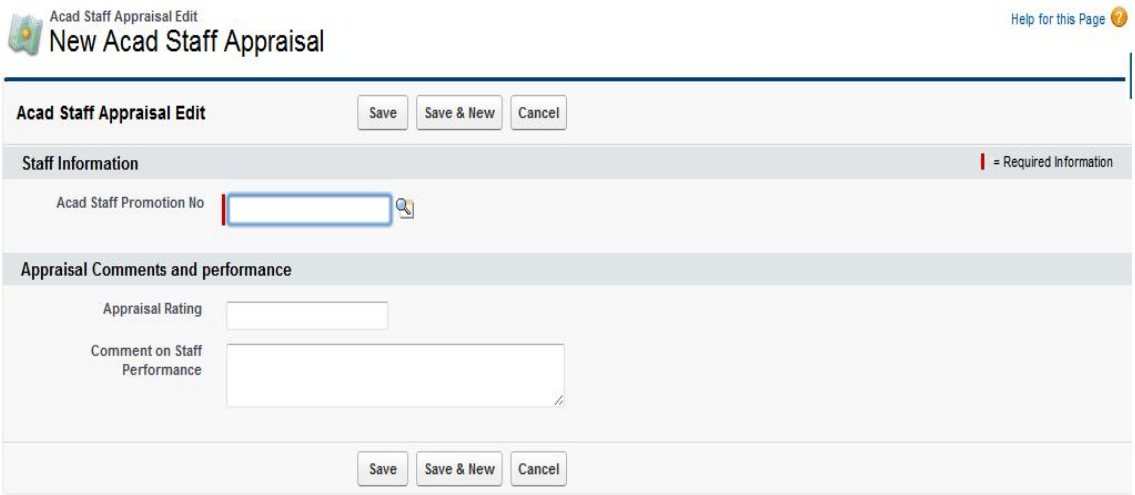


Figure 4.18: Academic Staff Appraisal form

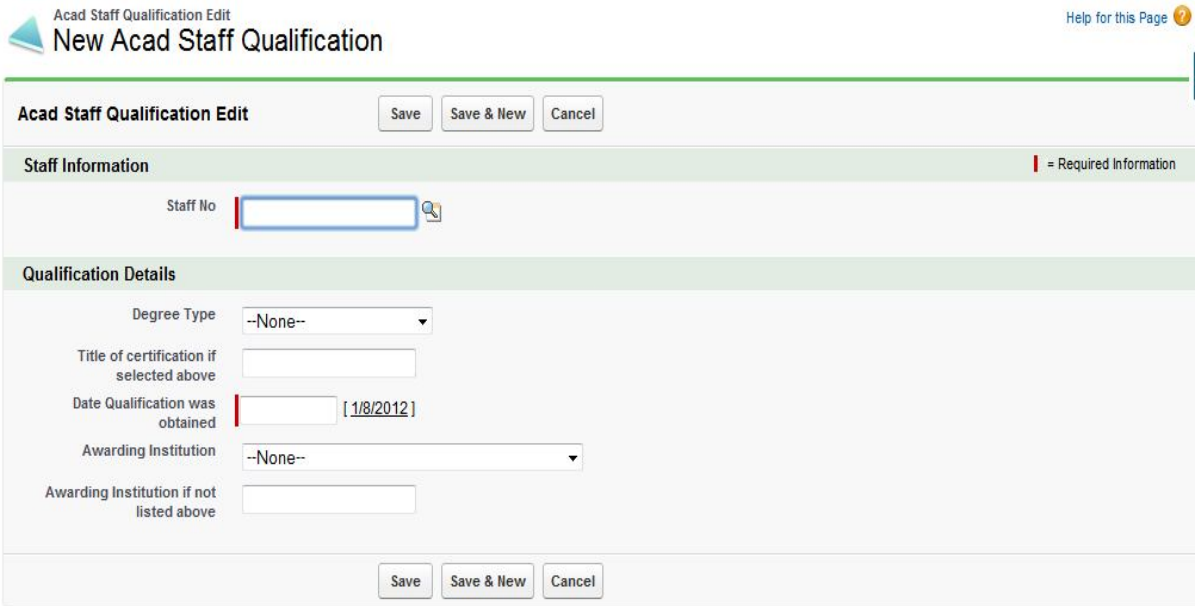


Figure 4.19: Academic Staff Qualification Form

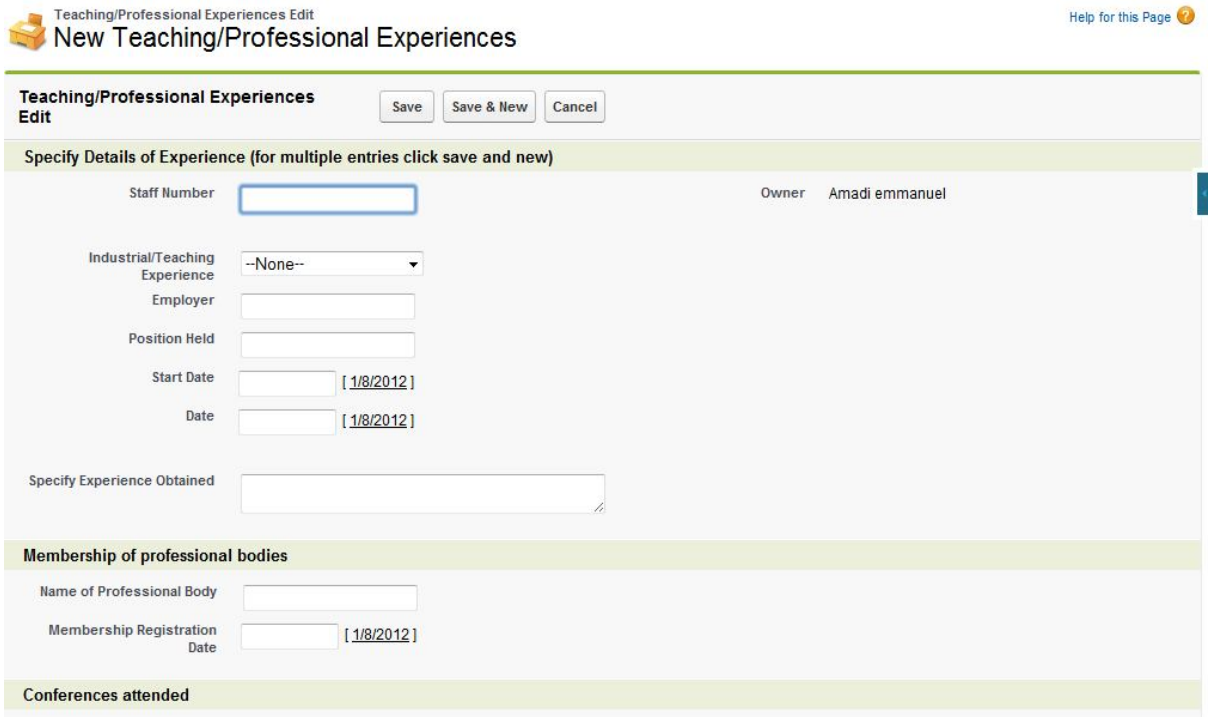


Figure 4.20: Teaching/Professional experience form

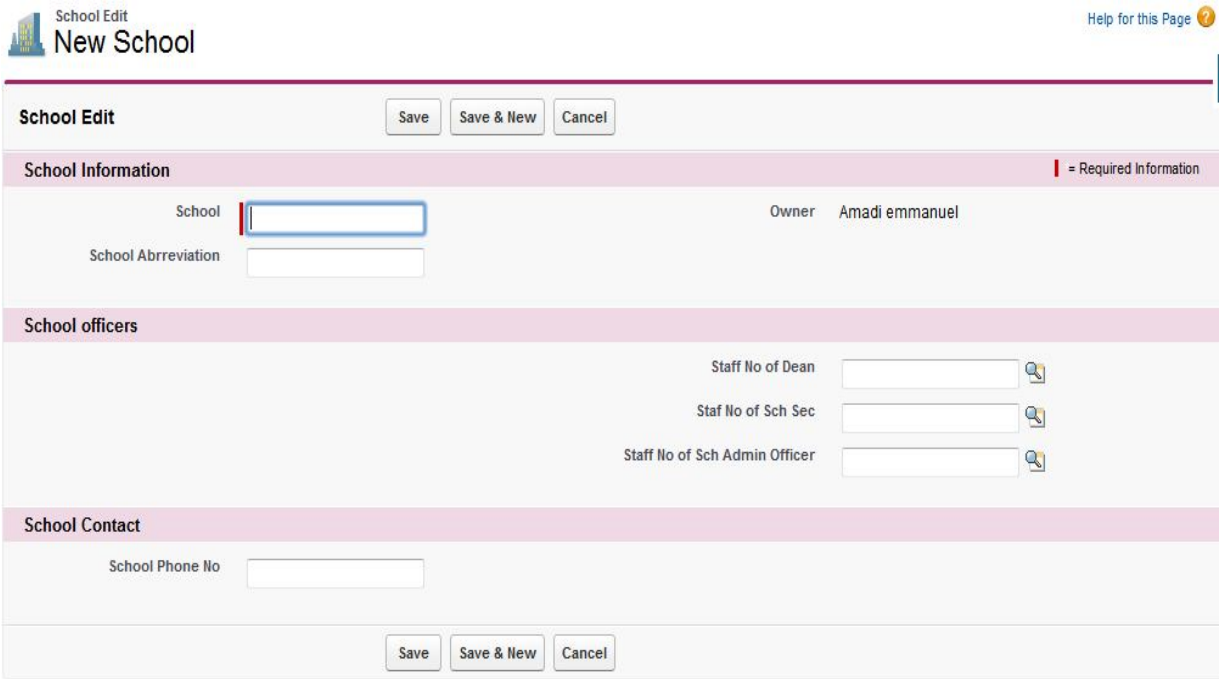


Figure 4.21: Register New School Form

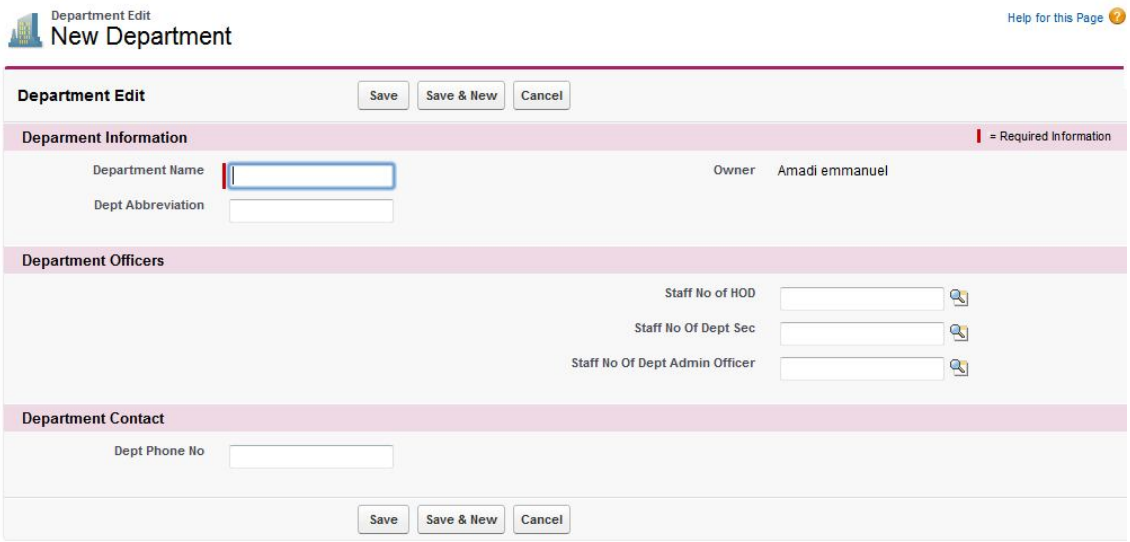


Figure 4.22: Register new department form

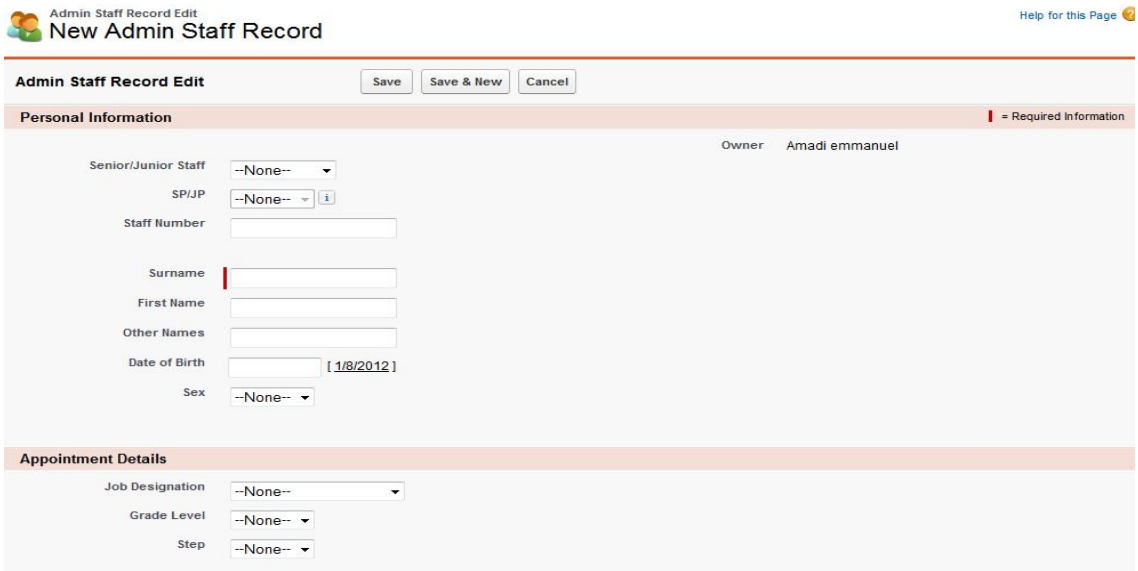


Figure 4.23: Register Admin Staff Record form

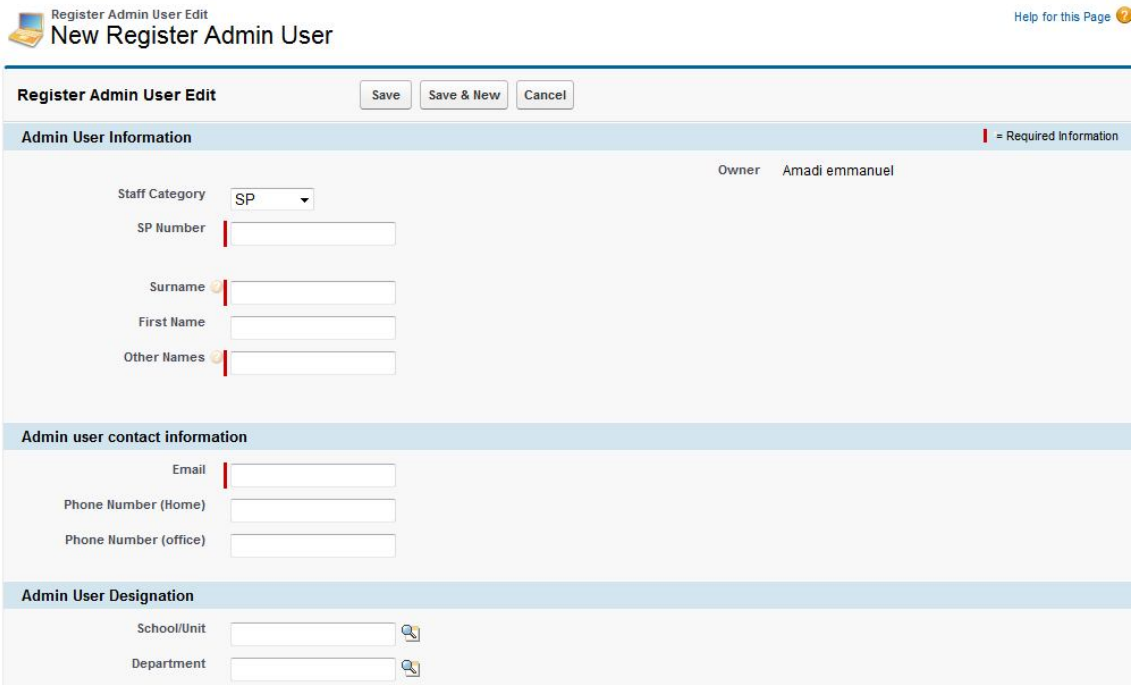


Figure 4.24: Register Admin user Form

4.6 Implementing Security and Sharing Rules

The TIRMP app is used by all personnel's of the institution thus the need for roles and privileges to be implemented. To assign roles and privileges on the platform, the organizational structure of the institution need to be taken into consideration. The institutional organizational structure shows how information flows from top to bottom. Figure 4.13 shows the organizational structure of a typical institution.

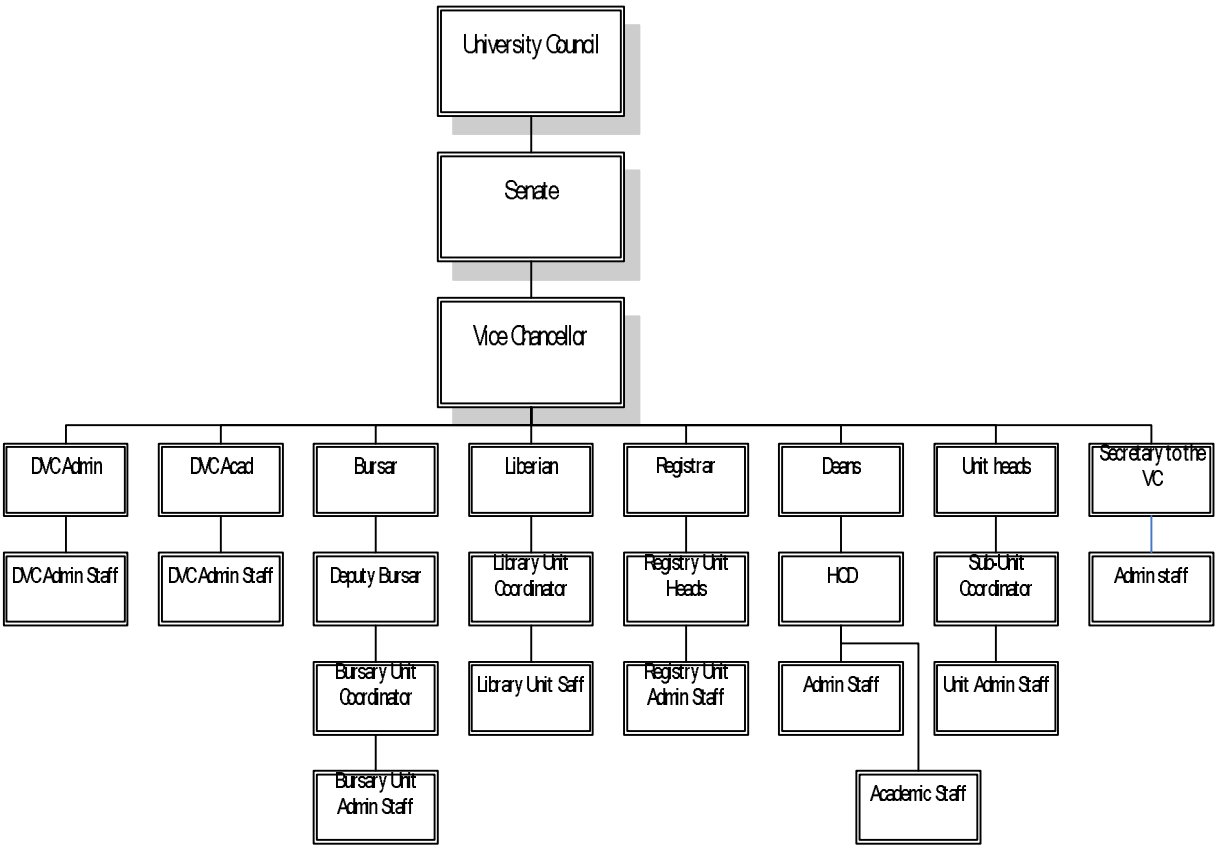


Figure 4.25: A typical institution's organizational structure

Before we get started implementing our security and sharing rules, the following are some of the ways that we can control data on the platform:

a. Object-Level Security

The bluntest way that we can control data is by preventing a user from seeing, creating, editing, or deleting any instance of a particular type of object, like a position or review. Object-level access allows us to hide whole tabs and objects from particular users, so that they don't even know that type of data exists. On the platform, we set object-level access rules with object permissions on user profiles.

b. Field-Level Security

A variation on object-level access is field-level access, in which a user can be prevented from seeing, editing, and/or deleting the value for a particular field on an object. Field-level access allows us to hide sensitive information without having to hide the whole object. On the platform, we set field-level access rules with the field-level security.

c. Record-Level Security

To control data with a little more finesse, we can allow particular users to view an object, but then restrict the individual object records that they're allowed to see. On the platform, we actually have four ways of setting record-level access rules:

- *Organization-wide defaults* allow us to specify the baseline level of access that a user has in your organization. For example, we can make it so that any user can see any record of a particular object to which their user profile gives them access, but so that they'll need extra permissions to actually edit one.
- *Role hierarchies* allow us to make sure that a manager will always have access to the same records as his or her subordinates.
- *Sharing rules* allow us to make automatic exceptions to organization-wide defaults for particular groups of users.
- *Manual sharing* allows record owners to give read and edit permissions to folks who might not have access to the record any other way.

4.6.1 Controlling Access to Objects

First we will configure access to the TIRMP app custom objects. We can control whether a user knows that a particular object exits in the app by modifying his or her profile.

A profile is a collection of settings and permissions that determine what a user can do in the platform. Profiles control:

- The objects the user can view, create, edit, and delete
- The object fields the user can view and edit
- The tabs the user can view in the app
- The standard and custom apps the user can access
- The page layouts a user sees
- The record types available to the user
- The hours during which the user can log in to the app
- The IP addresses from which the user can log in to the app

Profiles are typically defined by a user's job function. A profile can be assigned to many users, but a user can be assigned to only one profile at a time.

The platform provides the following set of standard profiles in every organization. These standard profiles can then be modified to suit any organizational structure:

- Read Only
- Standard User
- Marketing User
- Contract Manager
- Solution Manager
- System Administrator

Each of these standard profiles includes a default set of permissions for all of the standard objects available on the platform. For example, users assigned to the Standard User profile can never create, edit, or delete a campaign.

In customizing the profiles for the TIRMP app, we have the following app

Table 4.17: Table Showing the Various Profiles on the TIRMP App

S/N	Type of Users	Officers
1	Executive user	Executive council members and the Vice Chancellor
2	Principal users	DVC acad, DVC admin, Bursar, Registrar, Liberian,
3	Unit users	Unit heads
4	Sub unit users	Unit coordinator
5	School users	Dean
6	Departmental users	HOD
7	Administrative users	DAO,SAO
8	Standard users	Staff (academic and non academic staff members)

(Source: Author, 2012)

The following steps are used to create the profiles on the TIRMP app:

1. Click *Your Name – Setup – Manage Users – Profiles*
2. Click on new to create a new profile called *Executive* (the first profile in table 4.17), based on the standard user profile.
3. In the *New Executive* profile's detail page, click Edit
4. In the Custom App area, make the TIRMP app visible to users assigned to the Executive profile, as shown in figure 4.26.

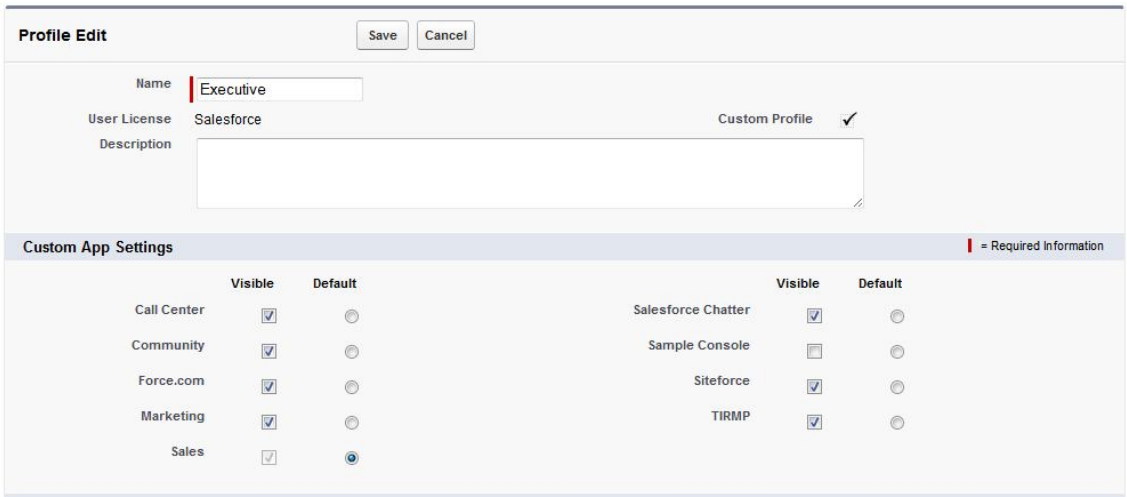


Figure 4.26: Profile Edit page

5. Next is the tab setting for the Executive profile users as shown in figure 4.27. For the Executive user, all tabs are default on.

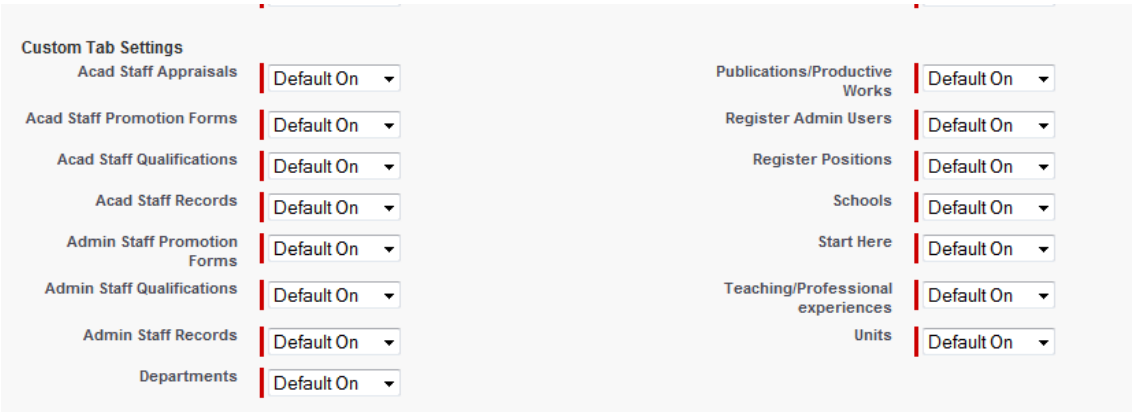


Figure 4.27: Custom tab settings page

6. Next is the custom object permission setting for the Executive users. What the Executive user can do and view on the platform against each object are shown in figure 4.258.

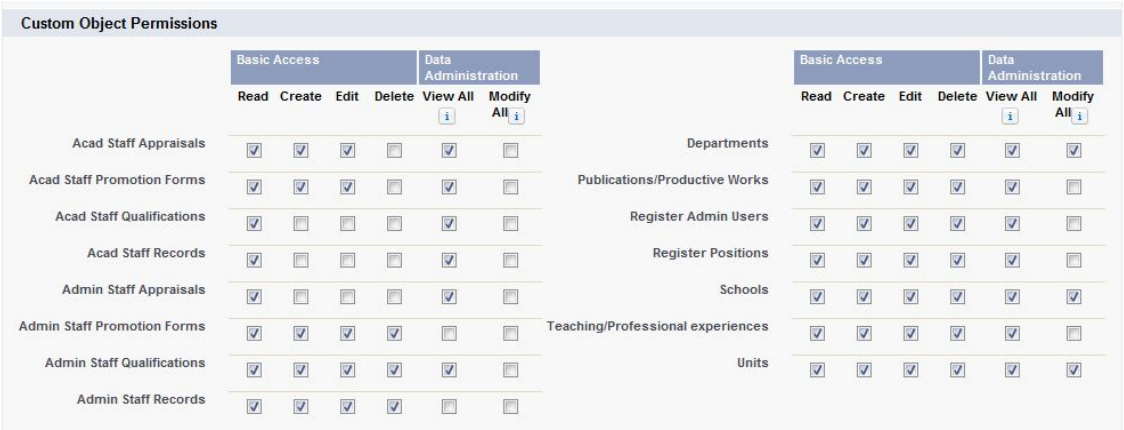


Figure 4.28: Custom object permission

7. Click save to create the profile and return to the profile detail page

4.7. Domain Naming and Site Creation on the Force.com Platform

The login address of the force.com platform could be mapped to a domain name registered for the institution using the app. Sites are hosted on the Force.com domain. This saves you the hassle of dealing with a third-party Web hosting service, and allows for seamless integration with the Force.com platform. Each Salesforce.com account that uses sites has its own reserved section on the Force.com domain servers.

Before you begin creating sites, you must register a unique Force.com domain name that will identify your account's space on the Force.com domain. Your Force.com domain name is a URL that consists of a prefix followed by a string of characters that Salesforce.com provides.

Prior to creating your first site, you must register your Force.com domain name. You only have to do this one time, though. Once your Force.com domain name is registered, you never have to repeat this part of the process again.

To register your Force.com domain name:

1. Click ***Your Name*** ➢ **Setup** ➢ **Develop** ➢ **Sites**.

   The sites registration page appears. Near the bottom of the page is a text box in a URL. This is where you will enter the prefix for your Force.com domain name. Before you do, though, consider the following:

   - The Force.com domain name can contain only underscores and alphanumeric characters (no spaces or punctuation), and must begin with a letter. Also, it can't end with an underscore or contain two consecutive underscores.
   - Your Force.com domain name must be unique throughout the entire Force.com domain. To avoid any duplicate entries, consider using your institution's name or a variation of your institution's name, such as FUTOng

- Salesforce.com combines the prefix you enter with the rest of the URL on the page to create your Force.com domain name. For example, if you enter FUTOng as your prefix and the rest of the URL is -developer-edition.na1.force.com, your full Force.com domain name will be www.FUTOng-developer-edition.na1.force.com.

2. In the text box, enter a prefix for your Force.com domain name.
3. Click **Check Availability**. Salesforce.com quickly scans all of the other Force.com domain names to make sure the one you entered is unique. If your Force.com domain name is unique, you'll see a message indicating it's available. If it's not unique, you'll have to try another name.
4. Click the link to review the Sites Terms of Use. Acknowledge your acceptance by selecting the checkbox.
5. Click **Register My Force.com Domain**. A prompt appears warning that you can't change your Force.com domain name after you register it.
6. Click **OK**.

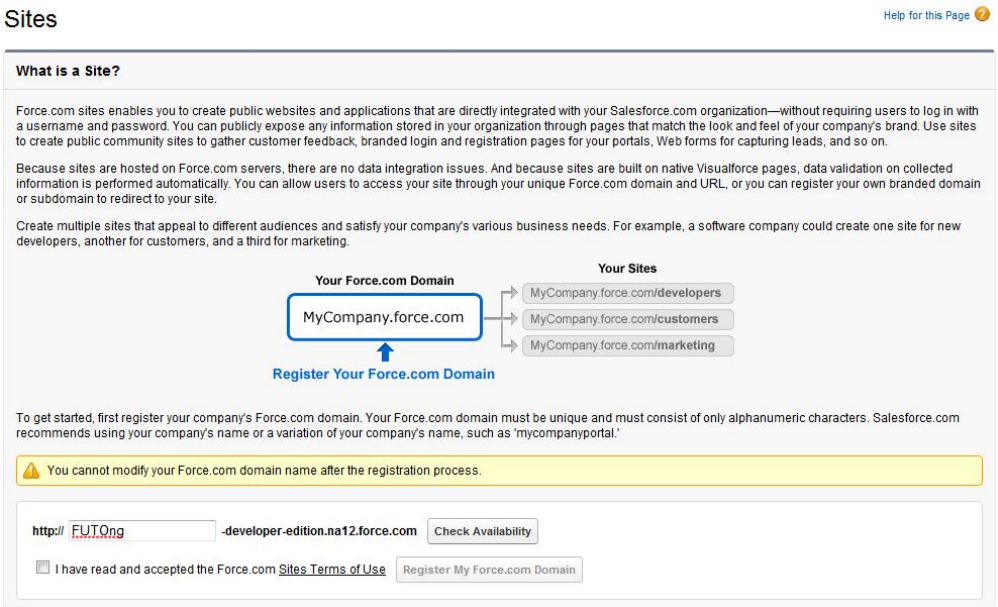


Figure 4.29: The Site Domain Registration Page of the Force.com Platform

4.8 TIRMP Platform Overview and Login

The TIRMP platform has seven basic users as already presented in chapter three of this work. The executive user creates the principal users and the principal users particularly the registrar creates the Admin users. The admin users are responsible for creating staff users at their various levels. They are restricted to their schools, units or departments.

The platform is designed such that a staff member i.e. the standard user at the lowest level of the hierarchy can access their records on the platform but are not permitted to edit all of them at will. Some areas of the platform are however available for the staff personnel to edit like the publication and productive work session. Their submissions however on this note will have to be verified based on hard copy document submitted to their appraisal heads before these entries are authenticated.

Once a platform user is registered, his login email and password is immediately emailed to him. The user simply logs into to the platform using his email and password via the platform address. For the test data, the platform address used is www.FUTOng-developer-edition.na12.force.com.

The TIRMP app runs on the cloud thus the problem of packaging and unpacking does not come into play. What a user simply does is to login form where ever and he or she is onto the platform. The platform is restricted to only registered users and each user has limits to what they can do on the platform. The platform is scalable i.e. it can be expanded with increase in users.

4.9 Navigating through the TIRMP app

The TIRMP app is made up of taps and fields. Tabs are used to navigate to forms that contain fields for data entry. Fields are entered and are saved on a distributed database network managed by the cloud provider in this case Force.com.

4.10 System Specification

The minimum system requirement for accessing the cloud platform is the ability of the system to connect to the internet via a gateway. The system should have sufficient capacity to stream the app. The system should have a web browser running on it such as safari, firefox or internet explorer. The memory and processor speed of the computer used to access the platform will determine the overall stream speed of the platform.

4.11 TIRMP Testing and Result

TIRMP testing shows that when ones a user is registered on the platform, the user is send his login parameter which are his email address and password. These parameters are used to log onto the platform. Each user belongs to a profile and has levels of privileges. After login, the home page of the platform is displayed. Tabs that are viewable by the user are displayed and can be navigated through. The use fills out his records on the forms and in each case the save button is clicked. When a record is saved after all required fields have been filled, it can be viewed, edited or deleted by the user based on his level of privilege.

Roll up summaries of records are easily generated on the platform for evaluation and verification purposes.

Parts of the app are also displayed as web pages to be viewed by the general public thus save the institution the extra cost of hosting a website.

4.12 TIRMP Report

This session displays some reports that are generated on the TIRMP platform using test data. On the TIRPM platform various reports of records can be generated using filters, field specification. This reports can thus be used to take decisions and also record institutional performance based on staff content. Table 4.18 to 4.22 shows certain drill down summaries generated from the TIRMP platform.

Table 4.18: Drill Down Summary of Academic Staff Showing-Title, Staff Number, Name and School/Unit

Title	Staff Number	Surname	First Name	School/Unit
Mr	SP2224	Ajoku	Peter	SCHOOL OF ENGINEERING AND ENGINEERING TECHNOLOGY
Dr	SP2223	Mary	Ekenta	SCHOOL OF ENGINEERING AND ENGINEERING TECHNOLOGY
Dr	SP1134	Andrew	Benjamin	SCHOOL OF ENGINEERING AND ENGINEERING TECHNOLOGY
Dr	SP2225	Magnus	Akpan	SCHOOL OF ENGINEERING AND ENGINEERING TECHNOLOGY
Prof	SP7771	Apkan	Eronini	SCHOOL OF ENGINEERING AND ENGINEERING TECHNOLOGY
Dr	SP1298	Jackson	Glaty	SCHOOL OF ENGINEERING AND ENGINEERING TECHNOLOGY
Engr	SP1114	Chukwuma	Admas	SCHOOL OF ENGINEERING AND ENGINEERING TECHNOLOGY
Mr	SP1214	Amadi	Okafor	SCHOOL OF ENGINEERING AND ENGINEERING TECHNOLOGY
Mrs	SP1136	Angela	Otowo	SCHOOL OF HEALTH TECHNOLOGY
Prof	SP5551	Jonathan	Okorobuchi	SCHOOL OF HEALTH TECHNOLOGY
Mr	SP1111	James	John	SCHOOL OF HEALTH TECHNOLOGY
Mr	SP6661	Ikenna	Gladys	SCHOOL OF SCIENCE
Dr	SP2221	Jude	Ekeocha	SCHOOL OF SCIENCE
Dr	SP5551	Amara	Ebube	SCHOOL OF SCIENCE
Prof	SP4441	Ikechukwu	Okocha	SCHOOL OF MANAGEMENT TECHNOLOGY
Dr	SP4451	Jonas	Ibru	SCHOOL OF MANAGEMENT TECHNOLOGY
Dr	SP2222	Maduka	Ugbo	SCHOOL OF MANAGEMENT TECHNOLOGY
Dr	SP3345	Nkechi	Okafor	SCHOOL OF MANAGEMENT TECHNOLOGY
Dr	SP1113	Amaobi	Uwaleke	SCHOOL OF MANAGEMENT TECHNOLOGY
Mr	SP1112	okafor	Chukwu	SCHOOL OF MANAGEMENT TECHNOLOGY
Engr	SP1454	Adaku	Okechukwu	SCHOOL OF AGRICULTURE
Prof	SP8881	James	Ukaga	SCHOOL OF AGRICULTURE
Mr	SP7723	Amadi	Uchechukwu	SCHOOL OF AGRICULTURE

Staff
report1

Generated By: Amadi Emmanuel 4/10/2012 5:00 AM

(Source: TIRMP App, 2012)

Table 4.19: Drill Down Summary of Schools, Deans and School Phone Numbers

School: School	School Abbreviation	Name of Dean	School Phone No
SCHOOL OF HEALTH TECHNOLOGY	SOHT	Jonathan,Okorobuchi	(080) 645-4876
SCHOOL OF MANAGEMENT TECHNOLOGY	SMAT	Nguzo,Chuka	(080) 621-4239
SCHOOL OF ENGINEERING AND ENGINEERING TECHNOLOGY	SEET	Apkan,Eronini	(080) 976-4321
SCHOOL OF SCIENCE	SOSC	Ikenna,Gladys	(080) 245-9872
SCHOOL OF AGRICULTURE	SAAT	James,Ukaga	(080) 654-2132

school report1

Generated By: Amadi Emmanuel 4/10/2012 5:17 AM

(Source: TIRMP App, 2012)

Table 4.20: Drill Down Summary of Departments, HOD, and Department Phone Numbers

Department: Department Name	Dept Abbreviation	Name of HOD	Dept Phone No
Transport Management Technology	MMT	okafor,chukwu	(080) 662-1345
Information Management Technology	IMT	Amaobi,Uwaleke	(071) 666-5554
Electrical Engineering	EEE	Chukwuma,Admas	(080) 965-3423
Heath Technology	HET	Angela,Otowo	(080) 765-3425
Industrial Production Engineering	IPE	Jackson,Glaty	(081) 234-6531
Civil Engineering	CEE	Apkan,Eronini	(070) 534-2765
Mritime Management Technology	MMT	Nkechi,Okafor	(080) 675-4323
Project Management Technology	PMT	Maduka,Ugbo	(080) 765-3421
Managment Technology	MGT	Jonas,Ibru	(071) 087-6542
Geology and Mining	GOM	Amara,Ebube	(080) 765-3849
Textile Engineering	TXE	Magnus,Akpan	(080) 765-3249
Mathematics	MAT	Jude,Ekeocha	(080) 643-2678
Mechanical Engineering	MEE	Andrew,Benjamin	(070) 654-2332
Animal Science	ASC	James,Ukaga	(080) 234-5431
Dental Technology	DNT	Mary,Ekenta	(080) 245-3676
Agric Economics and Extension	AEE	Adaku,Okechukwu	(080) 621-4329
Chemical Engineering	CHE	Ajoku,Peter	(080) 634-5248

department report1

Generated By: Amadi Emmanuel 4/10/2012 5:27 AM

(Source: TIRMP App, 2012)

Table 4.21: Drill Down Summary of Academic Staff in Grade Level 4 and Above

Surname	First Name	Acad Staff Record: Staff Number	Department	Grade Level
Andrew	Benjamin	SP1134	Mechanical Engineering	4
Chukwuma	Admas	SP1114	Electrical Engineering	5
Ajoku	Peter	SP2224	Chemical Engineering	5
Apkan	Eronini	SP7771	Civil Engineering	7
Magnus	Akpan	SP2225	Textile Engineering	4
Amadi	Okafor	SP1214	Textile Engineering	4
Jonas	Ibru	SP4451	Management Technology	4
Amaobi	Uwaleke	SP1113	Information Management Technology	6
Ikechukwu	Okocha	SP4441	Project Management Technology	7
Maduka	Ugbo	SP2222	Project Management Technology	5
Nkechi	Okafor	SP3345	Mritime Management Technology	4
Nguzo	Chuka	SP3321	Mritime Management Technology	7
Adaku	Okechukwu	SP1454	Agric Economics and Extension	4
James	Ukaga	SP8881	Animal Science	7
Jude	Ekeocha	SP2221	Mathematics	5
Ikenna	GLadys	SP6661	Geology and Mining	7
Amara	Ebube	SP5551	Geology and Mining	4
Angela	Otowo	SP1136	Heath Technology	4
Mary	Ekenta	SP2223	Dental Technology	6
Jonathan	Okorobuchi	SP5551	Dental Technology	7

Staff in level 4 and above2

Generated By: Amadi Emmanuel 4/10/2012 5:42 AM

(Source: TIRMP App, 2012)

Table: 4.22: Drill Down Summary of Academic Staff Above 40years

Title	Surname	First Name	Acad. Staff Record: Staff Number	Date of Birth
Prof	Nguzo	Chuka	SP3321	04/10/1951
Prof	Apkan	Eronini	SP7771	02/07/1961
Prof	James	Ukaga	SP8881	01/07/1959
Dr	Mary	Ekenta	SP2223	01/07/1955
Mr	Ajoku	Peter	SP2224	04/07/1957

Acad Staff Record Report
Copyright (c) 2000-2012 salesforce.com, inc. All rights reserved.
Confidential Information - Do Not Distribute
Generated By: Amadi Emmanuel 4/10/2012 6:24 AM

(Source: TIRMP App, 2012)

4.13 Cost estimation for Setting up a Campus Area Network

4.13.1 Ground rules and assumptions for the Campus Area Network

- The project was preceded by a detailed study and proof of concept to show that is was possible to develop the hardware and install the software needed for a high speed Campus area network for internet access.
- Cost overrun is put into consideration because of changing technology
- There are data available to estimate the cost of the Linux licensed software, servers, access points and other hardware components
- No data is available to estimate the cost for software development
- The main goal of this project is to
    - ✓ setup a campus area network for high speed internet access
    - ✓ setup nine access point
    - ✓ Configuration of an Email/DNS and http server with appropriate software and settings.
    - ✓ Configuration of industrial routers, access points and switches on the network
    - ✓ Test the new system within the Federal University of Technology, Owerri

- ✓ Train 10 lecturers on the management of the campus area network

- The work break down structure (WBS) of the campus area network is as shown below:

  1. *Project Management*
  2. *Hardware*
     2.1 Servers
     2.2 Access points
     2.3 Router
     2.4 Switch
     2.5 Fibre cables
     2.6 Cat 5 cables
     2.7 Device racks
     2.8 Face plates, patch cables, tool kit, heat sink and other devices
  3. *Software*
     3.1 Server licensed software (Linux)
     3.2 Software installation and configuration
  4. *Testing*
  5. *Training and Support*
  6. *Reserves*

- Cost must be estimated by WBS and by month. The project coordinator will report progress on the project using earned value analysis.
- Cost will be provided in naira. Since the project length is less than one year, inflation will now be included
- The project will be managed by the Information Management Technology Department in collaboration with other engineering and science departments. Their will be a project manager which will be a Senior lecturer in the department, ten other lectures and ten 300level students will be assigned to the project. There will also be a consulting company to help in the supervision of the project. The consulting company will

delegate three experts to work with the internal staff members of the university. The lecturers and students will help manage various aspects of the project and provide their expertise in the area of software development, training and support.

- The project involves purchasing:
  - ✓ A two servers; an Email/DNS server and a http server. Each server cost ₦300,000
  - ✓ Access points, the cost rate per access point is ₦100,000. The project will require nine access points
  - ✓ Switches, the cost rate per switch is ₦100,000. The project will require twelve switches
  - ✓ Router, the cost rate per router is ₦200,000. The project will require twelve router
  - ✓ Linux licensed software at ₦60,000

- An estimate based on 5% multiplied by total hardware and software cost should be sufficient for the testing.
- Because there are several risks related to IT projects especially this type of project that is the first of its kind in the institution. Thus 20% of total estimate will be included as reserve.
- A computer model for the estimate is to be developed to make it easy to change several inputs, such as number of labour per hour for various activities or labour rates.
- The time duration required to setup the network is 6 months

4.13.2 Assumptions in Estimating the Cost for Each Work Break Down Structure (WBS)

1. *Project Management*

   Budget experts suggested using a labour rate of ₦1500/hr for the project manager, ₦1000/hr for the lecturers and ₦500/hr for students on their working of 72hours per month (6hrs, 3 times a week), full time.

The project will be completed in 3 months and will last for 5years before it will need a major upgrade.

1.1 Project manager

Total hours for the project manager is 72 hrs per month for 3 months

$72 \times 3 = 216 hrs$

1.2 lecturers

One of the lecturers will act as the assistant project manger and will work along side the project manager for 72 hrs a month. While the other nine lecturers will work for 24hrs a month

$(72 \times 3) + 9(24 \times 3) = 864 hrs$

1.3 Consultants

The three consultants will work 24hrs for two month

$3(24 \times 2) = 144 hrs$

1.4 Riggers

The riggers will work on contract for one month

1.5 Students

The ten students assigned to the project will work 24 hrs each per month

$10(24 \times 3) = 720 hrs$

2. *Hardware*

2.1 Servers

One server estimated at N200,000 based on recent market prize

2.2 Access points

One access point estimated at N100,000 based on recent market prize

2.3 Router

One router cost N200,000 based on recent market prize

2.4 Switch

One switch cost N100,000 based on recent market prize

2.5 Fibre cables

One drum of fibre cable cost N1000,000

2.6 Cat 5 cables

One carton of cat 5 cable cost N10,000

2.7 Device racks

Each rack cost N50,000

2.8 Other tools and devices

Other devices such as face plates patch cables and tool kits for the entire project will cost N500,000

3. *Software*

   3.1 Server licensed software (Linux)

   3.2 Software installation and configuration

4. *Testing*
5. *Training and Support*
6. *Reserves*

4.13.3 Calculating the Net Present Value, Return on Investment and Payback Period for the Campus area Network

In calculating the NPV, the following assumptions are used:

- Discount rate = 8%
- The project will be completed in 3months
- The project will be used for 5 years before it will need a major upgrade
- The project cost is in naira

Table 4.23: Net present value table for the Campus area network

Project year	**0**	**1**	**2**	**3**	**4**	**5**	**Total**
Cost	11,000,000	200,000	200,000	200,000	200,000	200,000	
Discount factor	1	0.93	0.86	0.79	0.74	0.68	
Discount cost	11,000,000	186,000	172,000	158,000	148,000	136,000	**11,800,000**
benefit	0	5,000,000	5,000,000	5,000,000	5,000,000	5,000,000	
Discount factor	1	0.93	0.86	0.79	0.74	0.68	
Discount benefit	0	4,650,000	4,300,000	3,950,000	3,700,000	3,400,000	**20,000,000**
Discount	(11,000,000)	4,464,000	4,128,000	3,792,000	3,552,000	3,264,000	**8,200,000**

benefit less discount cost							
Cumulative discount benefit less discount cost	(11,000,000)	6,536,000	2,408,000	1,384,000	4,936,000	8,200,000	

(Source: Author, 2012)

A. Calculating the Net Present value (NPV)

$$NPV = \sum_{t=0}^{n} \frac{A_t}{(1+r)^t}$$

NPV = total discounted benefit - total discounted cost

From the table above,

NPV = 20,000,000 – 11,800,000 = 8,200,000

Comment:

Since the NPV is positive, the project is worth engaging in.

B. Calculating the Return on investment (ROI)

$$ROI = \frac{total\ discounted\ benefit \quad - \quad total\ discounted\ \cos t}{total\ discounted \cos t}$$

$$ROI = \frac{20{,}000{,}000 - 11{,}800{,}000}{11{,}800{,}000} \times \frac{100}{1} = 69.5\%$$

Comment:

Since the ROI is positive, and the higher the ROI the better the project, we can say that the project is good.

C. Estimating the Payback period

This is the amount of time it will take to recoup, in the form of net cash inflows, the total money invested in the project.

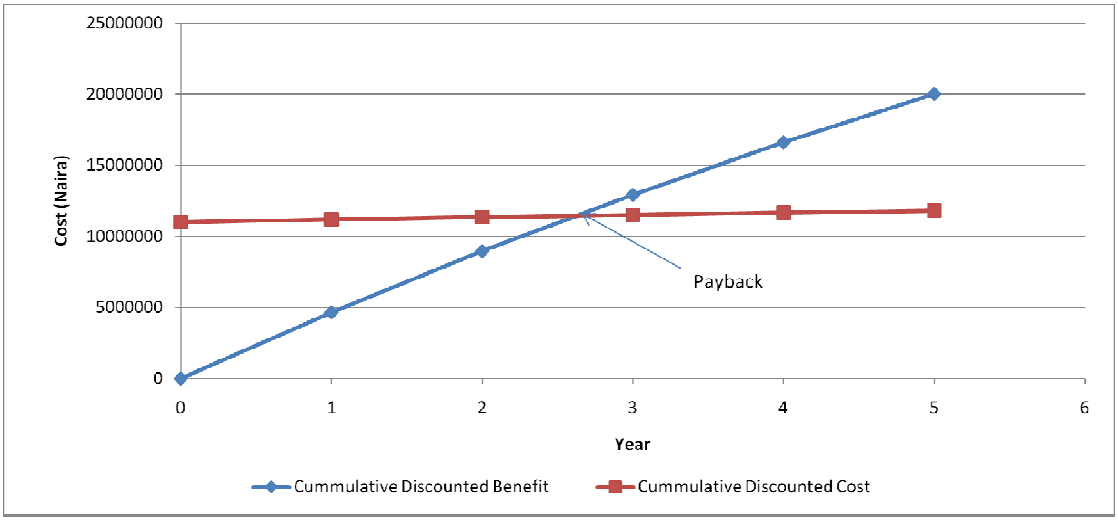


Figure 4.30: Graph illustrating pay back period (PBP) for the Campus area network cost estimation (the point of intersection is the PBP)

Comment:

Since the PBP occurs in the third year, as shown on the diagram at the point of intersection, the project is a good one.

4.13.4 Cost Estimate for the Campus area Network Design

Table 4.24: Campus Area Network cost estimate created January 12th, 2012

	Units/hr	Cost/unit/hr	Subtotal	WBS level 1 Total	% of Total
WBS Items					
1. Project Management				₦3,636,800	34.20%
1.1.Project manager	216	₦1500	₦324,000		
1.2.Lecturers	864	₦1000	₦864,000		
1.3.Students	720	₦500	₦360,800		
1.4.Contractors	144	₦2000	₦288,000		
1.5.Riggers			₦1,800,000		
2. Hardware				₦5,540,000	52.10%
2.1. Server	2	₦200,000	₦400,000		

2.2. Access point devices	9	₦100,000	₦900,000		
2.3. Switch device	12	₦100,000	₦1,200,000		
2.4. Router device	12	₦200,000	₦1,200,000		
2.5.Fibre cable			₦1,000,000		
2.6.Cat5 cable	4 cartons	₦10,000	₦40,000		
2.7.Device racks	6	₦50,000	₦300,000		
2.8.Other devices			₦500,000		
3. Software				₦340,000	3.20%
3.1. Licensed software	4	₦10,000	₦40,000		
3.2.Configuration			₦300,000		
4. Testing (10% of software development)			₦50,000	₦50,000	0.47%
5. Training and support				₦50,000	0.47%
5.1. Trainee cost			₦50,000	₦50,000	
6. Reserves (10% of total estimate)			₦966,680	₦966,680	9.09%
Total project cost estimate				₦10,633,480	

(Source: Author, 2012)

4.14 Cost Estimation for Developing the TIRMP App

4.14.1 Ground Rules and Assumptions for the TIRMP App

- The project was preceded by a detailed study and proof of concept to show that is was possible to develop the TIRMP app on the Force.com platform
- Cost overrun is put into consideration because of changing technology

- There are data available to estimate the cost of cloud space subscription
- The main goal of this project is to
  - ✓ Develop the TIRMP app that runs on the cloud
  - ✓ Test the new system within the Federal University of Technology, Owerri
  - ✓ Train 11 lecturers on the management of the TIRMP app

- The work break down structure (WBS) of the campus area network is as shown below:

1. *Developers*
   1.1 Senior Developer
   1.2 Developer
   1.3 Consultant
2. *Hardware*
   2.1 Laptop
   2.2 Glo NetPro modem
3. *Cloud subscription*
4. *Training and Support*

- Cost must be estimated by WBS and by month. The project coordinator will report progress on the project using earned value analysis.
- Cost will be provided in naira. Since the project length is less than one year, inflation will now be included
- The project will be managed by the Information Management Technology Department in collaboration with other engineering and science departments. Their will be a senior developer which will be a Senior lecturer in the department, six other lectures will be assigned to the project. There will also be a consulting company to help in the supervision of the project. The consulting company will delegate one expert to work with the internal staff members of the university remotely. The lecturers and students

will help manage various aspects of the project and provide their expertise in the area of application development, training and support.

- The project involves purchasing:
  - ✓ 5 laptop computer. Each computer cost ₦200,000
  - ✓ Glo NetPro internet modem, each costing ₦10,000. The project will require 5 internet modems
  - ✓ The project will also require subscription for internet access for 3 months at ₦5,000 per month
- A computer model for the estimate is to be developed to make it easy to change several inputs, such as number of labour per hour for various activities or labour rates.
- The time duration required to setup the TIRMP app is 3 months

4.14.2 Assumptions in Estimating the Cost for each Work Break down Structure (WBS)

1. *Developers*

   Budget experts suggested using a labour rate of ₦2000/hr for the senior developer, ₦1500/hr for the lecturers and ₦2000/hr for the. The project will be completed in 3 months and will last for 5years before it will need a major upgrade.

   1.1 Senior developer

   Total hours for the senior developer is 84 hrs per month (3hrs daily) for 3 months

   $84 \times 3 = 256 hrs$

   1.2 Developers

   The ten developers will work 84 hrs per month (3hrs daily) for 3 months

   $(84 \times 3) \times 4 = 1008 hrs$

   1.3 Consultants

   The consultant will work 84hrs per month (3hrs daily) for 3 months

   $84 \times 3 = 256 hrs$

2. *Hardware*

   2.1 Laptop

One laptop is estimated at N200,000 based on recent market prize. 5 laptops are required for the project

2.2 Glo NetPro modem is estimated at N10,000. 5 modems are required for the project

3. *Cloud space subscription*

   The cloud space subscription cost is based on the recent prizing on the Force.com platform.

4. *Training and Support: training will run simultaneously with development. After the development the staffs members that participated in the development process will go ahead to train other staff members at various schools. Training will last for two day; 4hrs each day simultaneously at 6 schools/faculties.*

   $8 \times 6 \times 2 = 96 hrs$

4.14.3 Calculating the Net Present Value, Payback Period and ROI for the Result Management Portal Project

- Discount rate = 8%
- The project will be completed in 3months
- The project will be used for 5 years before it will need a major upgrade
- The project cost is in naira

Table 4.25: Net Present Value Calculation for the TIRMP app

Project year	**0**	**1**	**2**	**3**	**4**	**5**	**Total**
Cost	4,000,000	200,000	200,000	200,000	200,000	200,000	
Discount factor	1	0.93	0.86	0.79	0.74	0.68	
Discount cost	4,000,000	186,000	172,000	158,000	148,000	136,000	**4,800,000**
Benefit	0	2,000,000	2,000,000	2,000,000	2,000,000	2,000,000	
Discount factor	1	0.93	0.86	0.79	0.74	0.68	
Discount	0	1,860,000	1,720,000	1,580,000	1,480,000	1,360,000	**8,000,000**

benefit							
Discount benefit – discount cost	(4,000,000)	1,674,000	1,548,000	1,422,000	1,332,000	1,224,000	**3,200,000**
Cumulative discount benefit – cost	(4,000,000)	(2,326,000)	(778,000)	644,000	1,976,000	3,200,000	

(Source: Author, 2012)

A. Calculating the Net Present value (NPV)

$$NPV = \sum_{t=0}^{n} \frac{A_t}{(1+r)^t}$$

NPV = total discounted benefit - total discounted cost

From the table above,

NPV = 8,000,000 – 4,800,000 = 3,200,000

Comment:

Since the NPV is positive, the project is worth engaging in.

B. Calculating the Return on investment (ROI)

$$ROI = \frac{total\ discounted\ benefit \quad - \quad total\ discounted\ \cos t}{total\ discounted \cos t}$$

$$ROI = \frac{8{,}000{,}000 - 4{,}800{,}000}{4{,}800{,}000} \times \frac{100}{1} = 66.7\%$$

Comment:

Since the ROI is positive, and the higher the ROI the better the project, we can say that the project is good.

C. Estimating the Payback period

This is the amount of time it will take to recoup, in the form of net cash inflows, the total money invested in the project.

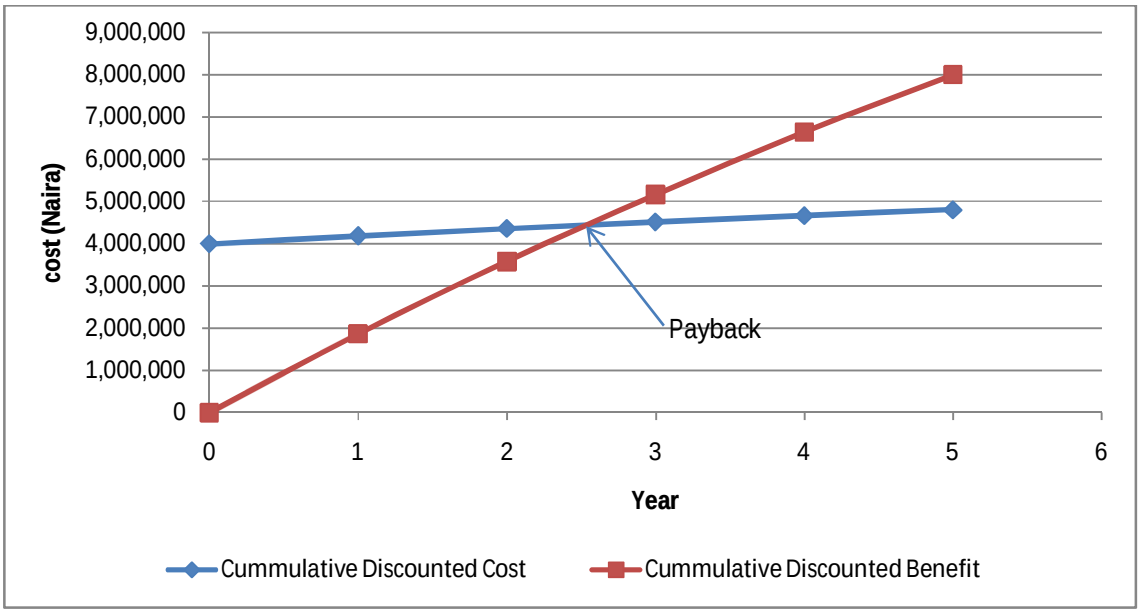


Figure 5.2: Graph illustrating pay back period (PBP) for the TIRMP app development (The point of intersection is the PBP)

Comment:

Since the PBP occurs in the third year, as shown on the diagram at the point of intersection, the project is a good one.

4.14.4 Cost Estimate for the TIRMP App

Table 4.26: Cost estimate for the TIRMP app created 12th January, 2011

	Units/hr	Cost/unit/hr	Subtotal	WBS level 1 Total	% of Total
WBS Items					
1. Developer				₦2,536,000	58.81%
6.1.Senior developer	256	₦2000	₦512,000		
6.2.Developers	1008	₦1500	₦1,512,000		
6.3.Consultant	256	₦2000	₦512,000		

7. Hardware				₦1,075,000	24.93%
7.1. Laptop computers	5	₦200,000	₦1,000,000		
7.2. Glo NetPro modem	5	₦10,000	₦50,000		
7.3. NetPro subscription	5	₦5,000	₦25,000		
8. Cloud subscription for one year				₦117,000	2.71%
9. Training and support	96	₦2,000		₦192,000	4.45%
Total project cost estimate				₦3,920,000	

(Source: Author, 2012)

4.15 Executive Cost Summary for Deploying and Running the TIRMP App

The following summary is required for the design, testing, training and deployment of the Campus area network and the design of the TIRMP app that runs on the cloud. The cost includes a one year subscription.

Table 4.27: Table showing the total cost of setting up a TIRMP app on a CAN with one year cloud and internet subscription

Table 4.27: Executive Cost Summary of the TIRMP App Design and Implementation

Item	Cost (Naira)
Campus area network design	11,000,000
TIRMP App development	4,000,000
Total	**15,000,000**

(Source: Author, 2011)

CHAPTER FIVE

5.0 CONCLUSION AND RECOMMENDATION

5.1 Conclusion

The TIRMP provides a platform on which an institution can manage its resources on a virtual environment with less physical infrastructure and scalable bandwidth utilization. The TIRMP focused on the **Personnel records** of an institution**.**

The personnel are the staff members of the institution both academic and non academic staff operating at different levels. This interface can be adopted to suit any higher institution with little or no adjustments on the design interface. The Forec.com cloud computing platform was used as the cloud service provider.

The work provides a model of an improved Campus Area Network (CAN) for optimised internet access with nine access point within the campus environment. It also provides a model database structure of key personnel information using ERD diagram and how these information is keyed onto the cloud platform using tables and fields. Other relationships, roles and security features are also configured on the platform. The Personnel data structure of the Federal University of Technology, Owerri, Nigeria was deployed on the platform for the testing process.

The TIRMP app eliminates the challenge of managing personnel manually with hardcopy documents. It provides a real time system that captures personnel information, update records and generate summaries and ultimately can be managed by the institution. A lot of cost is saved with the TIRMP in use.

The following are summaries of the basic function of the TIRMP app

- Track all the personnel records in the institution both academic and non academic
- Track staff publications, books, research works at various levels.

- Track teaching and professional experience of staffs and their growth in the institution for the purpose of appraisal for promotion.
- Allow for the appraisal of staff members at various levels
- Track available vacancy within the institution and provide a list of possible candidates to fill those vacancies
- Keep track of employment status of staff members provide such information to the staff members in real time.
- Inform staff members of necessary steps to take as regards their growth within the institution.
- Automate the posting of vacancies to the institutions website.

5.2 Problems Encountered

Implementing the project has its pitfalls. One major issue encountered was the implementation of more robust security and trigger system on the platform. This is largely due to time constraint.

Another challenge encountered is the testing of the various modules of the project during development as the developer version on the force.com platform was used. This is basically due to the challenge of funds available for subscription on the platform for development.

One key ingredient required for development on the cloud is the availability of internet access with considerable speed. Due to the challenge of internet connectivity speed within the state (Imo state), the developmental process was slow as development on the cloud require high speed and stable internet connectivity.

5.3 Recommendations

Having carried out this work thus far, some recommendations are necessary to ensure its deployment and use. Recommendations are also required for further research and this work and on cloud computing in general.

Cloud computing is a new paradigm in information technology and as such there is a need for researched to delve into this field of study to develop it further. For the development of the TIRMP, the following are my recommendation:

1. The development of cloud computing app should be done as a team as ideas sheared brings out the best from an application
2. This work should be expanded to include all the four modules of the TIRM app as presented at the beginning of the work
3. The rollup summary, security features and roles should be further modified and enhanced
4. In subscribing for cloud space, all the institution in a particular country can come together under a unified body and subscribe for the unlimited package as this will greatly reduce running cost. The unlimited version has no limit to number of apps or storage space.
5. In developing the app internet connectivity should be up and running to best performance in the development process.
6. The maintenance of the system should be done monthly and new module added as the case may be
7. Relationship with the cloud service provider should be cordial in the event of any major system upgrade.
8. The TIRMP should undergo a major system upgrade after five years due to changing technology in the information technology industry.

In developing the campus area network, the following are my recommendations:

1. A viable consulting company should be invited to collaborate with the institutions staff and students
2. The equipment to be used should be ordered directly from the manufactures with the correct specification.
3. Proper configuration and testing should be done before the system is opened for use.
4. A fibre link should be used from any of the telecommunication company. In the event of not being able to get a fibre link, a dedicated line should be used for high speed performance
5. A monthly routine maintenance should be carried out on the network, and the network should undergo a major upgrade after every five years because of changing technology in the information technology industry.

REFERENCES

AccessGrid. Available at: < http://www.accessgrid.org> [Accessed November 2006].

Amazon. 2009. *About Amazon Web Services*. Amazon Web Services Homepage. Available at: <http://aws.amazon.com/>. [Accessed 25 February 2010]

Andrew S. T. 2007. *Computer Networks*. Fourth ed. Pearson Education: Vrije Universiteit Amsterdam. pp. 1-2.

Anette, S.. *Utility Computing*. IBM Technology update. Available at: http://www- 05.ibm.com/nl/events/presentations/Utility_Computing.pdf [Accessed 15 August 2011].

APACGrid. Available at: < http://www.apac.edu.au> [Accessed November 2006].

Armbrust, M., et al., 2009. *Above the clouds: A berkeley view of cloud computing*. EECS Department. University of California: Berkeley, Tech. Rep. UCB/EECS.

ATIS committee PRQC. 2007. *Network topology*. ATIS Telecom Glossary. Alliance for Telecommunications Industry Solutions. [Accessed 10 October 2010].

Available at: <http://www.mcmcse.com/cisco/guides/hierarchical_model.shtml> [Accessed 11th December 2011].

Behrouz, A. F., Sophia, C. F. 2007. *Data communication and networking*. Fourth ed. Forouzan Networking Series. McGraw-Hill: New York. pp 3-90

Bicsi, B. 2002. *Network Design Basics for Cabling Professionals. City:* McGraw-Hill Professional.

Boehm B. 1988. '*A Spiral Model for Software Development and Enhancement.* IEEE Computer. 21:5. pp 61–72.

Buyya, R., et al. 2009. *Cloud Computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility.* Future Generation Computer Systems, 25, 599-616.

Caroline, R. 2010. *About Google Apps*. Available at: <http://www.google.com/apps/intl/en/business/index.html.> [Accessed 12 April 2010]

Centre of Medicare and Medicaid Services 2008 SELECTING A DEVELOPMENT APPROACH

Chee, S. Y., el al. 2009. *Utility grids*. [pdf] Available at:

<http://www.buyya.com/papers/HandbookCN_Utility_Grids.pdf> [Accessed 3 December 2011].

ChinaGrid. Available at: < http://www.chinagrid.edu.cn> [Accessed November 2006].

Cloud Computing, Conference on Object Oriented Programming Systems Languages and Applications. Orlando, Florida, USA.

Computing Curricula. 2005. *The Joint Task Force for Computing Curricula 2005*. Available at: <http://www.acm.org/education/curric_vols/CC2005-March06Final.pdf> [accessed 16th august 2011].

CRM. *Trends and Technologies In The CRM World*. Available at: <http://www.allthingscrm.com/cloud-computing/the-importance-of-cloud-computing.html> [Accessed 16 August 2011].

Dave, E., et al. 2002. *Networking Complete*. Third ed. Willey publication Sybex: San Francisco.

David G. 2002. *Theory and problems of Software Engineering*. Shuams outline series, Computing and Information Sciences Department Kansas State University: The McGraw-Hill Companies, Inc. pp 17-20

Deriving entity-relationship and relational model diagrams. Available at: <http://ficara.biz/pdf/DerivingERandRMDiagrams.pdf> [Accessed 5 December 2011].

Dialogic Corporation. *Introduction to Cloud Computing*. [pdf] White Paper. Dialogic Corporation 9800 Cavendish Blvd., 5th floor Montreal, Quebec CANADA H4M 2V9. Available at: <http://www.dialogic.com/products/docs/whitepapers/12023-cloud-computing-wp.pdf> [Accessed 22 September 2011].

Encyclopedia Americana International edition 2007. Vol 7. Gronlier Incorporated International Headquartes: Dombury Connecticut D6816, pp.472-490.

EU Data Mining Grid. Available at: < http://www.datamininggrid.org> [Accessed November 2006].

Foster, I., Zhao, Y., Raicu, I. and Lu, S. 2008. *Cloud Computing and Grid Computing 360-Degree Compared*. University of Chicago pp.10-56.

Gregory F. Pfister. 1998. *In Search of Clusters*, Second Ed. Prentice Hall

GriPhyN. Available at: < http://www.griphyn.org> [Accessed November 2006].

Groth, D. and Toby, S. 2005. Network+ Study Guid. Fourth ed. Sybex, Inc. http://www.edrawsoft.com/Hierarchical-Network-Design.php [Accessed 11th December 2011]

https://www.cms.gov/SystemLifeCycleFramework/downloads/SelectingDevelopmentApproach.pdf [accessed 20 February 2012]

Human resource management manual http://eprints.mdx.ac.uk/2682/1/Muller_-_HRM_under_institutional_constraints.pdf. [Accessed 29 August 2011].

Ian S. 2000. *Software engineering*. 6th edition.: McGraw-Hill

Ian, F. and Carl, K. 2003. *The Grid 2: Blueprint for a New Computing Infrastructure*.

Ian, F. and Carl, K. 2003. *The Grid 2: Blueprint for a New Computing Infrastructure*.

Morgan Kaufmann.

Ian. F., Carl K. and Steven T., 2001. *The Anatomy of the Grid: Enabling Scalable Virtual Organizations.* The International Journal of High Performance Computing Applications. Vol. 15, No. 3. pp. 200-222.

Jasmine, A. 2009. *A brief history of cloud computing.* Available at: http://www.undertheradarblog.com/blog/a-brief-history-of-cloud-computing/ [accessed 15 August 2011].

Kambil, A. 2009. *A head in the clouds*. *Journal of Business Strategy*, **30** (4) pp. 58-59. Available at: <http://www.emeraldinsight.com/Insight/viewContentItem.do;jsessionid=6DE7620223924615A02DDCAE5A9356DD?contentType=Article&hdAction=lnkpdf&contentId=1810662&history=true> [Accessed 22 September 2011].

Klaus, K., Rajkumar, B., and Muthucumaru M., 2002. *A taxonomy and survey of grid resource management systems for distributed computing Software*. Practice and Experience. Vol. 32, No. 2. pp. 135-164.

Kroenke, D. 2009. *Database processing: fundamentals, design, and implementation*. 97th edition. Person's publication. pp 919-999.

LHCGrid. Available at: < http://www.cern.ch/lcg> [Accessed November 2006].

LINUX SOCKET PART 14 Advanced TCP/IP - The TCP/IP Stack & OSI Layer Available at: <http://www.tenouk.com/Module42.html> [Accessed 9 December 2011].

Local Area Network (LAN) Overview. 1996. BICSI LAN Design Manual - CD-ROM, Issue 1. [Accessed 18 December 2011].

Luit InfoTech '*What is cloud computing*' available at:

<http://www.luitinfotech.com/kc/what-is-cloud-computing.pdf> [Accessed 4 August 2011].

Mario Cannataro and Domenico Talia, The Knowledge Grid, *Communications of the ACM*, vol. 46, no. 1, 2003, pp. 89-93.

Narasimhan, B. (2009) Cloud Computing Saving – Real or Imaginary? *Appirio.* Available

at: <http://blog.appirio.com/2009/04/cloud-computing-savings-realor/html.> [Accessed 15 February 2010].

NetSolve/GridSolve. Available at: < http://icl.cs.utk.edu/netsolve> [Accessed November 2006].

Networking components and device [pdf] Available at: <http://www.pearsonhighered.com/samplechapter/0789732556.pdf> [Accessed 18 December 2011].

Nvidia. 2010. *NVIDIA Tesla GPUs Power World's Fastest Supercomputer* Available at: <http://pressroom.nvidia.com/easyir/customrel.do?> [Accessed 3 November 2011].

O'Brien, J. A. and Marakas, G. M. 2008. *Management Information Systems.* New York: McGraw-Hill Irwin.

P. Patel, A. Ranabahu and A. Sheth, 2009. *Service Level Agreement in*
Parnas D. 1979. *Designing Software for Ease of Extension and Contraction*. IEEE Transactions on Software Engineering (TOSE) 5:3. pp 128–138.

Petaflop Sequoia Supercomputer - United States., 2009. Available at: <http://www-03.ibm.com/press/us/en/pressrelease/26599.wss> [Accessed November 3 2011].

Peter R. and Carlos C. 2007. *Database systems design, implementation and management*. Seventh ed. Course technology a division of Thomas learning, Inc: Canada. pp28-146.

Phil, C., Chris M. 2010. *Force.com Platform Fundamentals an Introduction to Custom Application Development in the Cloud.* Version 5. Available at: http://www.developerforce.com/events/fundamentals/registration.php [Accessed 10 September 2011].

Proulx, S. R., Promislow, D. E. L. and Phillips, P. C. 2005.*Network thinking in ecology and evolution*. Trends in Ecology and Evolution. pp 345–353.

Rajkumar B. 1999. *High Performance Cluster Computing: Architectures and Systems*, Volume 1. Prentice Hall.

Rajkumar, B. and Srikumar, V., 2004. *The Gridbus Toolkit for Service Oriented Grid and Utility Computing: An Overview and Status Report. Proceedings of the First IEEE International Workshop on Grid Economics and Business Models (GECON 2004, April 23, 2004, Seoul, Korea).* ISBN 0-7803-8525-X, IEEE Press: New Jersey. pp19-36.

Rajkumar, B., Toni, C., and Hai J., 2001 *Single System Image.* The International Journal of High Performance Computing Applications. Vol. 15, No. 2. pp. 124-135.

Relational database management system. Available at: <http://searchsqlserver.techtarget.com/definition/relational-database-management-system > [accessed 5 December 2011].

Resource management http://en.wikipedia.org/wiki/Resource_management [accessed 29th

August 2011].

Richard P. F. Available at: <http://www.goodreads.com/author/quotes/1429989.Richard_P_Feynman> [Accessed 18 September 2012].

Rittinghouse, J. and Ransome, J., 2010. *Cloud Computing*. Boca Raton, FL: CRC Press.

Robert, S., and Geoff, M. 2005. *Beginning MySql*. Published by wiley publishing, inc. Available at: <www.wiley.com> [Accessed 10 December 2011].

Russell, K. 2002. *QuickStudy: System Development Life Cycle*. Available at: <http://www.computerworld.com/s/article/71151/System_Development_Life_Cycle> [Accessed 21 December 2011].

Sam, J. 2009. Randon *rats about tech stuff (cloud computing, intellectual property, security, etc).* Available at: http://samj.net/2009/03/cloud-computing-types-public-cloud.html [Accessed 3 December 2011].

Solution White paper of Dialogic Corporation. 2010. *Introduction to cloud computing*. Available at: http://www.dialogic.com/Solutions/Cloud-Communications/Learn/~/media/products/docs/whitepapers/12023-cloud-computing-wp.pdf [Accessed 5 September 2011]

Sven, G., Jim, P., and Sharad S., 2003. *Making the Utility Data Center a Power Station for the Enterprise Grid. HP Labs Technical Report*: Palo Alto. Available at: <http://www.hpl.hp.com/techreports/2003/> [Accessed November 2011].

System Design. Available at: <http://en.wikipedia.org/wiki/Systems_design> [Accessed 6 November 2011].

System development life cycle. Available at: <http://en.wikipedia.org/wiki/Systems_Development_Life_Cycle> [Accessed 21 December 2011].

TechTarget. *Searchcloudcomputi*ng. Available at: <http://searchcloudcomputing.techtarget.com/feature/Top-10-cloud-computing-providers-of-2011 > [Accessed 3 December 2011].

TeraGrid. Available at: http://www.teragrid.org> [Accessed November 2006].
The five major components of data communication. 2010. Available at: <http://www.divinekonection.info/articles/The-Five-Main-Components-of-Data-Communication-a49.html> [Accessed 8 December 2011].

The internet diagram. Available at: <http://i.i.com.com/cnwk.1d/i/tr/contentPics/Cisco-Self-Defending-Network--1.jpg> [Accessed 18 December 2011].

The open system interconnect model. Available at: <http://www.infocellar.com/networks/osi-model.htm> [Accessed 8 December 2011].

The role of human resource development and management http://www2.aau.org/wghe/gender/toolkit/Toolkit-module3.pdf [accessed 29th august 2011].

Thomas ERL, 2010. *The Prentice Hall Service-Oriented Computing Series*.

Thomas ERL. *The Prentice Hall Service-Oriented Computing Series*. Available at: <http://www.soabooks.com/>. [Accessed 3 December 2011].

Todd, L. 2007. *CCNA: Cisco Certified Network Associate Study Guid*. Sixth Edition. Wiley Publishing Inc: Indianapolis, Indiana. Pp 44-92.

Tom J., 2009. *IBM to build new monster supercomputer*. TechWorld. Available at: http://www.networkworld.com/news/2009/020409-ibm-to-build-new-monster.html > [Accessed 3 November 2011].

Types of networks. Available at: http://www.wifinotes.com/types-of-networks.html [Accessed 11 December 2011].

Vaquero, L. J., Caceres, M., Lindner, and Rodero-Merino., 2009. *A Break in the Clouds: Towards a Cloud Definition*. ACM SIGCOMM Computer Communication Review, 50–55.

Washington may 18 2011 technology. 2011. Available at: <http://washingtontechnology.com/articles/2011/05/18/agg-top-50-cloud-providers.aspx> [Accessed 3 December 2011].

What Are The Five Major Components Of Data Communication System? Available at: <http://computer-networks.blurtit.com/q963897.html> [Accessed 8 December 2011].

Wikipedia . 2011. *Relational database management system*. Available at: <http://en.wikipedia.org/wiki/Relational_database_management_system> [accessed 5 December 2011].

Wikipedia, 2011. *Abacus*. Computing and abacus Available at: http://en.wikipedia.org/wiki/Abacus[Accessed [20 August 2011].

Wikipedia, 2011. *Cloud computing*. Available at: <http://en.wikipedia.org/wiki/Cloud_computing> [Accessed 15 August 2011].

Wikipedia, 2011. *Computing*. Available at: <http://en.wikipedia.org/wiki/Computing> [accessed 20 August 2011].

Wikipedia, 2011. *History of computing* Available at: <http://en.wikipedia.org/wiki/History_of_computing>

Wikipedia. 2011. *Entity relationship mode*. Available at: <http://en.wikipedia.org/wiki/Entity-relationship_model> [Accessed 5 December 2011].

Wikipedia. 2011. Internetworking. Available at: <http://en.wikipedia.org/wiki/Internetworking> [Accessed 8 December 2011].

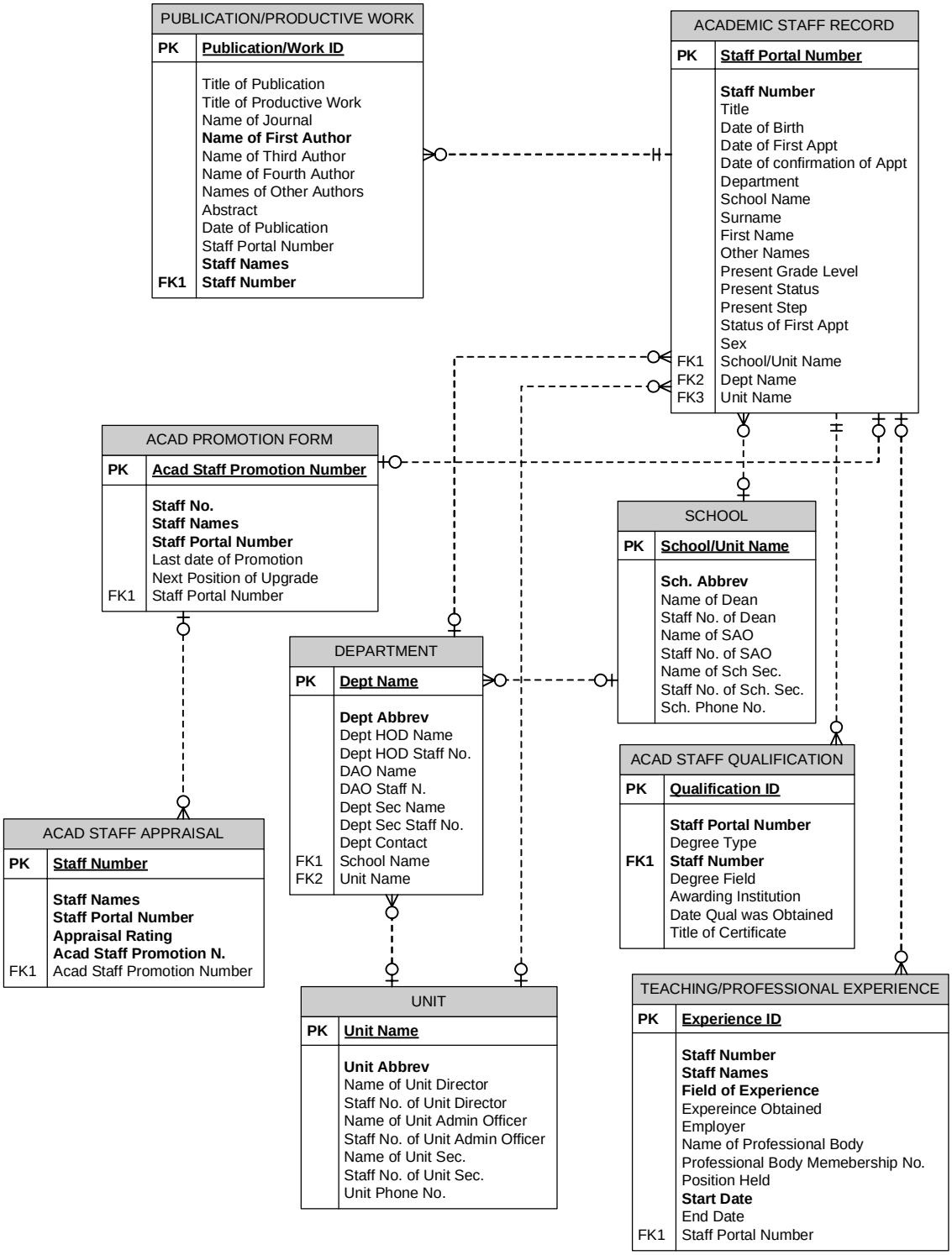
Wikipedia. 2011. *Network topology*. Available at: <http://en.wikipedia.org/wiki/Network_topology> [Accessed 11 December 2011].

Wikipedia. 2011. *Supercomputers.* Available at: <http://en.wikipedia.org/wiki/Supercomputer> [Accessed 3 December, 2011].

Wikipedia. 2011. *Telecommunications networks*. Available at: <http://en.wikipedia.org/wiki/Telecommunications_network> [Accessed 18 December 2011].

APPENDIX A

ERD DIAGRAM FOR THE ACADEMIC STAFF CATEGORY



APPENDIX B

ERD DIAGRAM FOR THE ADMINISTRATIVE STAFF CATEGORY