A Cloud Based Electronic Office Framework for Adama Science and Technology University

By: Gizatie Desalegn

A Thesis Submitted to the

Department of Computing

School of Electrical Engineering and Computing

Presented in Partial Fulfillment for the Degree of Master of Science in Software Engineering

Office of Graduate Studies

Adama Science and Technology University

Adama

January, 2018

A Cloud Based Electronic Office Framework for Adama Science and Technology University

By: Gizatie Desalegn

Name of Advisors: Dr. SelvaKumar S. and Dr. Mesfin Abebe

A Thesis Submitted to the

Department of Computing

School of Electrical Engineering and Computing

Presented in Partial Fulfillment for the Degree of Master of Science in Software Engineering

Office of Graduate Studies

Adama Science and Technology University

Adama

January, 2018

Declaration

I hereby declare that this MSc Thesis is my original work and has not been presented for a degree in any other university, and all sources of material used for this thesis have been duly acknowledged.

Name: ____

Signature: ____

This MSc Thesis has been submitted for examination with my approval as thesis advisor.

Name: ____

Signature: ____

Date of submission: __________________________

Approval of Board of Examiners

We, the undersigned, members of the Board of Examiners of the final open defense by ____________________________________ have read and evaluated his/her thesis entitled "____" and examined the candidate. This is, therefore, to certify that the thesis has been accepted in partial fulfillment of the requirement of the Degree of ……………………………………

______________________________ ______________________ ____________________

Supervisor /Advisor Signature Date

______________________________ ______________________ ____________________

Chairperson Signature Date

______________________________ ______________________ ____________________

Internal Examiner Signature Date

______________________________ ______________________ ____________________

External Examiner Signature Date

Acknowledgement

All praise and glory belong to Almighty God for the tremendous strength and courage. He gave me the persistency throughout my education. I have no words to thank Him enough for all His grace, guidance and unlimited pleasures and blessings.

It is my pleasure and duty to acknowledge all support i.e. the many people without whom this journey could not have been completed.

First, I would like to acknowledge the insight, generosity, patience, and encouragement of my advisors, Dr.Selvakumar S. and Dr. Mesfin Abebe., who have helped me keep my hope to finish this work.

Finally, I am especially grateful to all my family and friends for their continued patience and support, which has helped me, devote all my time to this work.

Table of Contents

List of Figures

List of Tables

List of Acronyms

Angular JS	Angular Java Script
APIs	Application Programming Interfaces
ASP	Active Server Page
ASTU	Adama Science and Technology University
BPMS	Business Process Management System
BSS	Business Support Service
CBEOF	Cloud Base Electronic Office Framework
CCRA	Cloud Computing Reference Architecture
CMS	Content Management System
CSS	Cascading Style Sheet
DMS	Document Management System
EA	Enterprise Architectures
E-GIF	Electronic Government Interoperability Frameworks
E-government	Electronic Government
EOF	Electronic Office Framework
E-Office	Electronic Office
E-Services	Electronic Services
G2B	Government to Business
G2C	Government to Citizen
G2E	Government to Employee
G2G	Government to Government

HTML	Hyper Text Markup Language
HTTPS	Hyper Text Transfer Protocol
IaaS	IaaS- Infrastructure as a Service
IAM	Identity and Access Management
IBM	International Business Machine
IBM CCRAs	International Business Machine Cloud Computing Reference Architectures
ICT	Information Communication Technology
ICTs	Information Communication Technologies
IDE	Integrated Development Environment
IIS	Internet Information Service
IT	Information Technology
JSP	JavaScript Object Notation
MVC	Model View Controller
NET	Network Enabled Technology
NIST	National Institute of Standards and Technology
OSS	Operation Support Service
PaaS	Platform as a Service
PKI	Public Key Infrastructure
RA	Reference Architecture
RF	Reference Framework
RM	Reference Model

SaaS	Software as a Service
SDK	Software Development Kit
SLA	Service Level Agreement
SMART	Specific Measurable Achievable Realistic Timely
SMS	Short Message Service
SOA	Service Oriented Architecture
SQL	Structured Query Language
SSH	Secure Shell
SSO	Single Sign-On
TH e-GIF	Thailand Electronic Government Interoperability Framework
UI	User Interface
UITM	University Technology MARA
US FEA	United States Federal Enterprise Architecture

Abstract

The government processes/workflows have necessity for accountability, transparency, and in the provision of services have long been recognized. The physical file transfer is permanent and requires continuous monitoring from office to office. As a result, many important decisions are being postponed due to the slow transfer of files and / or the unavailability of senior officers in the office, and it is difficult to manually prepare various institutional forms. To solve these problems, we suggest cloud-based electronic offices. The speedy need in such a scenario was to establish a system where by an authorized employee can find, update, and share the required forms, documents in the shortest conceivable time and eventually save them with appropriate references. Cloud computing has become an adaptive technology in many organizations. It is an important change in the way information technology is used. An effective cloud-based electronic office management system plays a crucial role in delivering services in a better way. The main goal of this research is to create a smart environment, a digital workplace solution, good communication, and collaboration between employers and university departments, and to add some cloud computing capabilities to the electronic office. The proposed system is intended to better drive the effective development of Smart society and to advance a step into an era of the paperless, replace the previous manual handling of documents with an efficient electronic system. The study use questionnaire, Interview, and observation data collection techniques and the developed prototype and proposed framework evaluate by different users, experts in software engineering and related fields. This study proposes a private cloud platform architecture for electronic office in ASTU Software as a Service. The cloud-based Electronic Office Framework (CBEOF) adapts the IBM Cloud Reference Architecture (IBMCRA), the NIST Cloud Reference Architecture, and the Microsoft Web Application Architecture to take advantage of all the considerations and capabilities of a cloud-based software architecture. The prototype is very interactive and user-friendly; it has prepared different form for services and performs various functions in the portal and in the back office. The prototype developed dynamic builder designer is based on the university attribute of the services.

Keywords: **Cloud computing, Cloud-based Architecture, Electronic office, Software architecture, Dynamic form builder, Digital workplace solution**

Chapter One: Introduction

1.1. Background of the study

In the area of government, Information and communication technologies (ICT) applications are encouraging to improve the delivery of services to citizens. Moreover, ICT play an increasingly important role in people's daily lives, revolutionize work and change business rules [1]. By applying information technology in government system, electronic Government (e-government) can be created.

To achieve the citizens goal anytime and anywhere, and eliminates the need to travel physically to government agents by implementing e-government to interact with computers, mobile, and tablets. Therefore, the use of ICT has changed the way of interaction between governments and citizens. With this in mind, e-government applications are being developed [2]. Currently, Web services and cloud computing are the most promising technologies for accessing and sharing information and resources everywhere. Cloud Computing has recently been used in almost every industry.

With the rapid growth of Internet users, the needs, and demands of citizens to complete their tasks with minimal time, many government agencies are offering e-government services through the website [3]. To improve government performance, be profitable, increase accountability, improve resource management, and provide better communication channels between citizens and the public sector by using Internet technology [4]. The non-Internet forms include telephone, fax, text messaging, etc. The short message services are among those that can be used as a network for citizens to communicate with the government. For example, In the Philippines, have SMS-based services use around Semi-Cabinet Agencies that allow citizens to obtain information about government officials and services [5]. Therefore, to define electronic office (E-office), it is a suit of digital communication systems that help employers collaborate with various offices in ASTU.

The paperless and transparent E-office system is designed to effectively connect the various departments and management offices, by taking into account the needs of employees and customers. Moreover, it's cloud-based, delivering all of the resources, application software, processing power, data storage, backup facilities, and development tools as a set of services over the Internet, allowing employers, customers, to use applications without installing them on their computers, and also providing access to save files from any computer with an internet connection [5].

Using web services and cloud computing technologies, the entire electronic office was stored in one place in the cloud and made available to authorized users from everywhere through extensive access to the network. The application can be providing and control by ASTU's ICT department, which saves costs, time, and increase the intelligent accessibility to university employers.

1.2. Motivation

According to our observation, the University Senior Management, Faculties, and Departments/ Units currently have weak relationships with each other or communicate in manual way. For this reason, there are various problems, especially the office tasks are not performed efficiently, effectively, and transparently. This can be solved by replacing the manual system of organizational units with the use of the dynamic form generator. Therefore, motivational research is to examine and identify the existing manual handling of files and documents in ASTU by using different methods and solving with an efficient electronic system. This solution includes the design of the cloud-based E-Office framework and the development of prototypes. The proposed system initiative aims to support the core processes in the office to deliver best-in-class services, transparency, accessibility, affordability, interactivity, and convenience within university employees and clients.

1.3. Statement of the Problem

Over the past twenty years, the use of ICT has radically changed the practices and procedures of almost all forms of corporate business and governance. In previous study, there are many problems related to office tasks like physically moving files and documents takes a long time and requires continuous supervision from desk to desk before senior officials make the final decision. As a result, many important decisions are delayed due to the slow movement of the files and / or the lack of availability in the office to eliminate these files [6]. In addition, the problems related to the interaction of employers with the government to use different services, there is a slow delivery service, lost files, less security, lack of availability and quality of services, and knowledge of (the existence of) services. In case of Case of The Blue House the documents were exposed to risk of concealment and fabrication and it was hard to collect public opinions on policies [7].

This study identified several issues that should be resolved with the proposed solution in the ASTU office tasks. The main issues are the repetitive and time-consuming processes / workflows such as document sharing, manual approvals, and routing, non-standard workflow, and repetitive form checks commonly used in ASTU offices. The internal communication system and the lack of coordination among different offices as well as the mismanagement of the knowledgebase are time consuming.

In each administrative and academic department, there are various problems associated with office task in routing of documents that are in efficient and ineffective, high storage costs, the manual way of preparing different form-based papers, difficult tracking of the document and it takes time to perform different task. In addition, there is a communication gap between the different departments and the staff as a result the university staffs are not satisfied. Quickly, in all cases paper documents, personal messengers are used for requesting decisions, manual document reception, and routing system is not efficient, complex tasks can take a lot of time. The limitation of existing studies not touched on E-office functionalities in cloud base.

Therefore, this study would focus with the development of cloud-based electronic office framework and prototype implementation that provides in managing the process of different documents, communicate in E-messaging, SMS and Email, E-forum discussions, apply dynamic form builder/generator for manual preparing of services form, using alerts and notification to notify different status, and using different engines like dispatcher and escalation rules to the office tasks in the ASTU which would be cost effective, accessible, transparent, create digital workplace/ smart environment, and deliverable. In addition, it creates commonplace or central information in which ASTU staff can access different information, share knowledge, and documents.

1.4. Research Questions

This study was to answer the following five research questions.

1. Could the cloud-based E-office model be a solution for integrating office tasks to efficiently use current office tasks for ASTU?

2. Is current office task process like the most frequently requested forms efficient and transparent?
3. How to solve the problems of the preparing forms manually?

1.5. Purpose of the study

The basic purpose of CBEOF is to create smart environment, digital workplace, and collaborations for ASTU customers. This will ultimately result in improvement of employers and user's potential to deliver quality service, to have good communication among employers, and different offices by implementing E-messaging, E-forum discussions, by managing institutional knowledgebase, implement escalation and despatcher engine, by integrating SMS, E-mail, alerts and notifications with their mobile data exchange, design workflow for institutional business process and different forms dynamically.

The design also used to create opportunities for the competitiveness and to make as Model University. In addition, the proposed system allows implementing cloud-based E-office system in ASTU, collaboration and to add cloud-computing features in the area of electronic office to design a cloud computing architecture for E-office. In addition, to improve the office task habits and to evaluate a cloud-based architecture by ASTU ICT professionals and other experts in software engineering and related fields.

1.6. Objectives

1.6.1. General Objective

The main objective of the study is to design a framework of cloud-based E-Office system for Adama Science and Technology University (ASTU) and to develop a prototype.

1.6.2. Specific Objective

The specific objectives of the study are the following:

- To examine the effectiveness of the current office tasks in ASTU.
- To analyze the internal communication channels among departments, faculties, and senior managerial office.
- To conduct an intensive literature review on cloud-based E-office and E-governance.
- To collect and prepare appropriate data from different sources to design a framework and develop a prototype implementation.

- To identify the best cloud computing platform and to conduct survey of potential use of cloud computing technologies for ASTU electronic office.
- To evaluate the framework and the prototype of the proposed system.

1.7. Scope and Limitations of the study

The main purpose of the study is to design a cloud-based electronic office for ASTU. For the design framework, a web-based prototype system was developed, which was deployed on a private cloud platform and would not have a complete implementation; because it needs more time, budget, and human power. The population of the study is composed of academic and administrative personnel who are concerned IT and cloud computing professionals. If several universities provide the electronic office, the framework will not take into account the integration of these electronic office providers that will be difficult to manage for cloud computing service providers. However, they can use the system with modification in this common place to make it accessible to other institutions.

The limitations of this study were the difficulty of responding quickly to interviews and questionnaires. The knowledge of cloud computing skills in the establishment was also the limit of the study and the lack of documentation on the practice of electronic office in cloud computing architecture. In addition, we used the Microsoft product for development purposes, because we are familiar with Microsoft products such as Visual Studio, and Microsoft Azure Cloud Service Provider Platform, but they are not free.

1.8. Research Methodology Overview

In order to achieve the overall work of this thesis, different methods and techniques were employed. Data-collection and analysis techniques, programming and technical tools, evaluation mechanisms for prototype developed system, and comprehensive document analysis as well as prototyping as a methodology. Primary data from ASTU directly and secondary data through documentary analysis were collected. For collecting data, the interview was used using a questionnaire that is due to this study using qualitative research approach. The questionnaires were distributed to ASTU employers and postgraduate students. The aim is to obtain qualification surveys from the ASTU employers in order to get a clear picture of the problem and identify possible solutions.

1.9. Significance of the application

The result of this research will contribute to the ongoing research in this domain area. It will be an input for different researchers, software developers, and students. The main beneficiaries of the results will be the university community as a whole. Likewise, it provides its own contribution to the beneficiaries such as government, university, developers, researchers, customers, and employees in ASTU by digitizing the different office task and it is routing, using different technologies like cloud providers on accepting cloud computing services. In addition, this study is significant because it adds both theoretical and practical knowledge and believes to bridge the gap of research work.

The direct beneficiaries and indirect benefits of the result are listed below -

Benefits for ASTU employers and software developers**:**

- Enables mobilization of electronic office application. The employers in the ASTU will access the application of electronic office anytime anywhere using their mobile phone, PDAs, PCs and other electronic devices.
- Used to create smart environment and digital workplace solution in which the employers should have good knowledge in technology and increase employee satisfaction.
- Encourage innovation by releasing time from unproductive processes and employee energy.
- This study mainly used by developers by understanding the proposed framework and prototype-developed system to implement the full system.

Benefits for ASTU and for Researchers:

- Save time, money, and increase efficiency.
- Contribute to create ICT society and increase competition.
- Enhance transparency - files of the university can be tracked and all at all times know their status.
- Increase accountability – to ensure data security, integrity and speed of decision-making is easier to monitor.
- Researchers from the education sector and software engineering fields can use the results of this thesis as an input to their study.
- Transform the institutional work, ethics, and encourage higher collaboration in the workplace.

- It will also help them design a better architecture and framework based on new technologies such as integrating cloud computing with SOA and web services together with other electronic office technologies.
- It will use to monitor the electronic office initiatives to make a decision to advance or rework initiatives by the decision makers in the office tasks.

1.10. Organization of the Thesis

This thesis work is organized in eight chapters. As we have discussed in this section, the first chapter presents introduction. The rest of this work organized as follows: Chapter 2 presents a review of literatures and related work on cloud-based e-office, e-governance frameworks; and the details of various related works that related to our work. Methodology of the work presented in Chapter 3. Chapter 4 discusses the selection of platforms and tools. The design of the proposed framework discussed in Chapter 5. Chapter 6 presents the development of the prototype. Chapter 7 presents the evaluation, discussion, and result. Finally, in chapter 8, conclusion remarks and future work discussed.

Chapter Two: Literature Review and Related Works

2.1. Introduction

In the first chapter, an introduction to the study was provided, including problem of the statement, purpose, objective, and significance of the study. This chapter deals with the documents review including different books and articles that are related to our work as well as used in this work in order to identify best practice and analyze the current situations of the area. First, it provides understanding overview of electronic office, the use of E-office applications in daily life, the need, and benefits of electronic office, and benchmarking of electronic office in other countries. Secondly, it explains about cloud computing; cloud computing service delivery and deployment models; cloud computing reference architecture, model, and framework. Lastly, works done by other researchers in the area that related to our study are discussed.

2.2. Overview on Electronic Office

Today, technologies are being developed and everything works with ICTs and office workers need to adapt to these ever-changing technologies and their problem-solving skills. Therefore, they need to build and develop their self-confidence in order to take advantage of the latest technologies, have the professional skills necessary to provide high quality office services, manage the infrastructure to provide office work and make quick decisions.

In general, office automation and workflow management, capture, storage, and control software are an electronic means of describing broad content and can assist in controlling content in conjunction with institutional processes and regulatory compliance [7]. It also aims to create a positive environment by eliminating the tedious number of documents and paper files, streamlining desktop workflow, shortening processing time for various form requirements, and then focusing on the management system. It also allows the university to improve the processing of its official letter / note and transform paper-based official documents and letters into a cloud-based electronic system. In addition, the design of most repetitive forms in the dynamic form generator, manage the institutional knowledgebase; Increase the communication culture through collaboration and messaging.

2.3. Need and Benefits of Electronic office

Nowadays, the responsibility for binding tax information is the responsibility of the officials of the organization. Given the large amount of information that passes through the various political, administrative, legislative, sales promotion, and governance functions that ease the organization, it is impossible to track and manage the control information manually. It would be extremely useful to use an integrated system of collection, dissemination, and use of all these elements of action and information with control information. The efficiency, productivity, and control that can be exercised in the implementation of any decision can be increased by using such a system. Benefits include improved efficiency and reduced response time, consistency, tracking and control of notes, reduced duplication, / wastage, increased workflow control, better tracking, reward, monitoring, and person independency [8].

2.4. Case of the Blue House (Korean Presidential Office)

Blue House needed a scientific record management for the low productivity of administrative work, such as records and maintenance, based on written documents and an electronic office system set up as part of innovative guidelines for national records management as a center of administrative works in government. The E-Office system, with its unique folder management function, has basic e-office functions, creating and saving records, organizing and maintaining records, registering standard management, opening, using records, managing records, and management of sensitive data [9].

2.5. Case of Seoul Metropolitan Government

In Seoul Metropolitan Government, implementation of "paperless" official documents have been achieved using the e-Document System since 2001, but due to the amendment of laws and regulations regarding the management of official record and administrative work, a system reconstruction was required. The specialized functions of Seoul e-Document System makes a high user friendly and accessible system, thus contributing to enhanced productivity and transparency, contributing much to improved productivity and to corruption prevention by automatically creating draft document, and by using attached document signature function, enhancing speed of work processing in office distant from main office [9].

2.6. The use of E-Office applications in daily office jobs

E-office for UITM: A survey analysis document related to the electronic office, the majority of respondents gave constructive comments and positive answers regarding the implementation of electronic office applications. They said that the electronic office could improve productivity, performance of daily work, skills, and knowledge of users, easy to access, reliable and easy to use. In addition, they suggested and recommended that implementation should be aligned, with good internet access and facilities, and adequate training be provided to users prior to the establishment of the electronic office. Finally, they expect the e-Office to be set up as quickly as possible [10].

2.7. Cloud Computing

In recent years, many IT professionals, business managers and researchers have begun to talk about a new phenomenon such as cloud computing. When invented is one of the first questions asked with the introduction of a new technology. In that, cloud computing has evolved through a series of phases that include grid and utility computing, provision of application services (ASP) and software as a service (SaaS). Each of these groups defined cloud computing differently according to their understanding of their offers.

"Cloud Computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction" [11].

The most recent and accepted standardized definition of cloud computing is the one by the National Institute of Standards and Technology (NIST) [12] [13]. The definition of NIST is concise, precise and differentiates between enabling characteristics and technologies (i.e., virtualization) and between main and derived characteristics (rapid elasticity versus massive scalability); However, the definition ignores the business model of cloud computing, which is the main driving force behind migration to the cloud. The NIST definition is relatively technical and covers all cloud delivery models (public, private, hybrid and community) and, services model (IaaS, PaaS and SaaS) as shown in Figure 2.1.

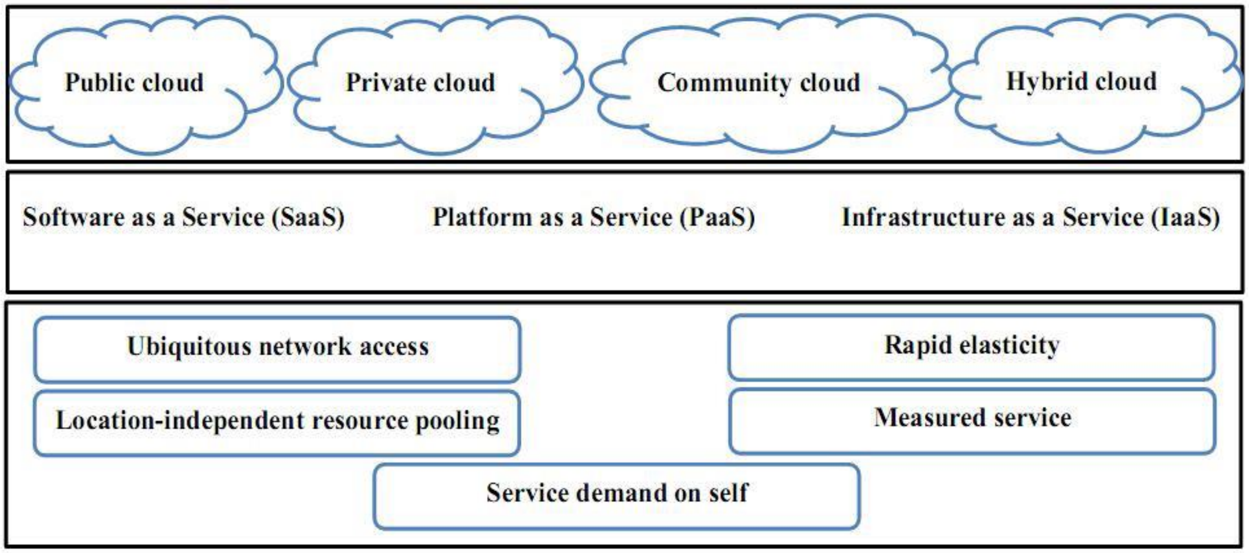

Public cloud
Private cloud
Community cloud
Hybrid cloud
Software as a Service (SaaS)
Platform as a Service (PaaS)
Infrastructure as a Service (IaaS)
Ubiquitous network access
Rapid elasticity
Location-independent resource pooling
Measured service
Service demand on self

Figure 2. 1 : NIST reference architecture [13]

2.7.1. Cloud Computing Service Model

Cloud computing uses the IT infrastructure as a service, and this service can vary from the rental of raw hardware to the use of third-party APIs. Cloud service models allow financial institutions to move from a capital-intensive approach to a more flexible business model that reduces operating costs. In practice, cloud service providers tend to offer services that can be classified into three categories: SaaS, PaaS and IaaS.

2.7.1.1. SaaS (Software as a Service)

In essence, SaaS is often used as a model where an application runs directly in the cloud and the user accesses it through the Internet. It refers to on-demand software, which is built on top of cloud's PaaS. As a result, Software that is implemented over the Internet with SaaS, a provider of an application's licenses to customers either as an on-demand service, through a subscription, on a "pay-per-use" model or (every time) no charge when there is the opportunity to generate income from transmissions other than the user, such as ads or sales of user lists [14]. Customer has no control over the underlying infrastructure or platform, except application parameters for specific user settings [13]. "SaaS, is a software delivery method that provides access to software and its functions remotely as a Web-based service" [15].

2.7.1.2. PaaS (Platform as a Service)

PaaS defined as a computing service provider platform that allows the creation and deployment of web based applications quickly, easily provides virtual environments for and developing, deploying, monitoring, and managing applications online without and with the cost of buying and managing the corresponding software tools or hardware [14]. In the case of PaaS, the cloud provider not only provides the hardware, but they also provide a toolkit and a number of supported programming languages (such as .NET, java or python) to build higher-level services [13].

2.7.1.3. IaaS (Infrastructure as a Service)

IaaS is the capability provided to the customer different raw like storage space, computing, or network resources, operating system, applications, or any software that they choose. In addition, customer can manage the software deployed but not able to control the distribution of the software to a specific hardware platform or change parameters of the underlying infrastructure [13].

2.7.2. Cloud Computing Deployment Models

The different infrastructure deployment models differ in their architecture, the location of the data center where the cloud is deployed and the requirements of the cloud provider's customers (for example, due to regulatory, legal or other requirements). Therefore, clouds categorized based upon the underlying infrastructure deployment model as Public, Private, Community, and Hybrid clouds.

2.7.2.1. Public clouds

A public cloud is one in which the cloud infrastructure and computing resources are made available to the public over a public network and owned by an organization selling cloud services, and assists a diverse pool of clients [16]. A cloud service provider owns a public cloud’s physical infrastructure. Such a cloud runs applications from different customers who pay for their resource utilization on a utility computing basis, share this infrastructure, most often hosted away from customer premises, and they provide a way to reduce customer risk, even temporary extension to enterprise infrastructure, and cost by providing a flexible.

2.7.2.2. Private clouds

A private cloud managed by the organization that consumes the services or a third party, and gives a single cloud consumer’s organization the exclusive access to computational resources and usage of the infrastructure. Moreover, the cloud may be outsourced to a hosting company (i.e. outsourced private clouds), or hosted on the organization's premises (i.e. on-site private clouds) [16]. Private clouds offer highest level of security and privacy (not available to general public), improved reliability, performance and flexibility, and have total control but are expensive and need additional skill. Private cloud used to manage the secure data by the ASTU rather than third- party Cloud services provider.

2.7.2.3. Community clouds

Community cloud is the cloud whose infrastructure shared by several organizations and supports a specific community that has shared concerns and may be managed by organizations collectively or by a third-party Cloud services provider [17]. Community cloud used with cost effective, good security comparatively than the public cloud and shared among organizations.

2.7.2.4. Hybrid clouds

A hybrid cloud consists of two or more clouds, like on-site private, on-site community, off-site private, off-site community, or public clouds that remain as independent entities, but are interconnected by standardized or patented technologies to allow the transfer of data and applications [16]. While public clouds allow companies to outsource parts of their infrastructure to cloud service providers, they lose control over resources and the distribution / management of codes and data. In some cases, this is not desired by the respective company.

2.7.3. Cloud computing for ASTU Office Tasks

It is providing the E-Office software as a service. The E-Office cloud model allows easy creation setup for ASTU offices by selecting modules or subsets of e-office product suite according to their need. The model makes e-office deployment aims for easier and faster process for Adama Science and Technology University to start using e-office services. In addition, to integrate the Microsoft Azure service provider platform cloud services with electronic office services to use by ASTU community.

2.7.4. The Definition of Cloud Reference Model, Architecture, and Framework

In software engineering, a reference model (RM) is conceptual, abstract, technology independent framework, which represents the relationships between them and a set of domain concepts. It is usually used by domain experts who are working independently toward a standard [12]. When reference model concepts are arranged in a specific order (pattern) to provide a specific solution for recurrent problem, the generated architecture called a reference architecture (RA) [18]. While it is important to use the correct terminology when describing architectures, models, and frameworks, formality absent from several works that have been recently published in the domain of Cloud Computing. Together, the set of RMs and RAs create a reference framework (RF) [19]. For example, both NIST Cloud Computing Reference Architectures and IBM are more reference frameworks than reference architectures.

2.7.5. Comparison between Cloud Computing Reference Frameworks

As discussed in the previous subsection, various reference models, frameworks and architectures for cloud computing are being developed or are in development. The question is how to choose the framework or model, how these frameworks differ, and best suits our needs. The most obvious distinction between current reference frameworks, and models is that some of them are generic, while others are focused on specific areas. NIST has recently compared five specific architectures and six generic reference architectures in cloud computing [19]. The analysis of the current framework of reference shows that these frameworks were first, decomposed into architectural styles and principles, which are a set of rules and guidelines that guarantee that best practices are applied methodically; and, secondly, reference models consisting of architectural elements and their relationships.

There are many similarities between IBM CCRAs and NIST [19]. Figures 2.2 and 2.3 show the IBM Cloud Computing Reference Architecture and the NIST Cloud Computing Reference Architecture, respectively. We have used both of the cloud architecture ideas and concepts to design the proposed framework especially the relationship of cloud service model with electronic office. While some architectural components are the same, both NIST and IBM CCRAs are comprehensive frameworks, and others have different wordings or are implicitly contained within other architectural components.

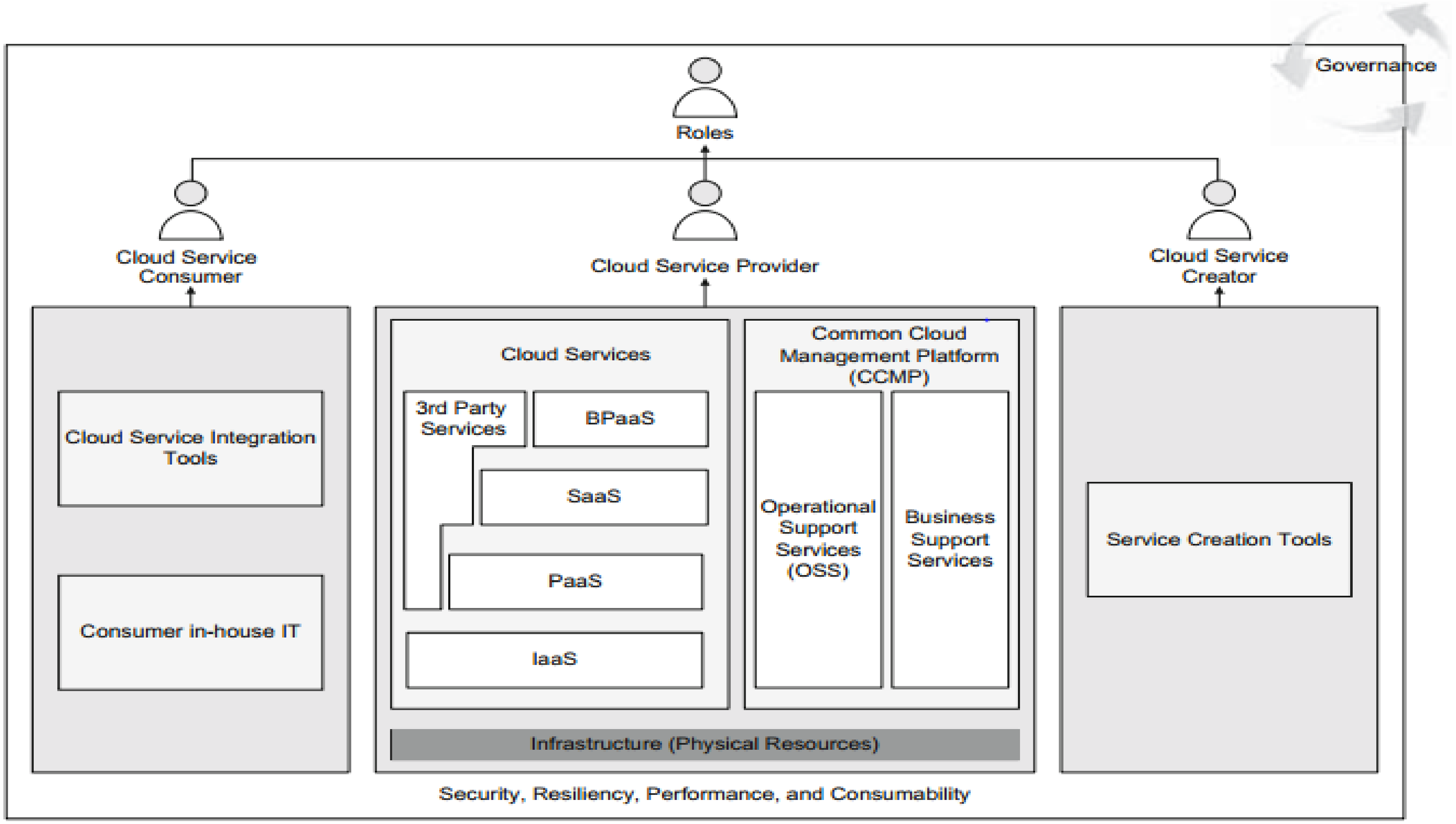

Governance
Roles
Cloud Service Consumer
Cloud Service Provider
Cloud Service Creator
Cloud Service Integration Tools
Consumer in-house IT
Cloud Services
3rd Party Services
BPaaS
SaaS
PaaS
IaaS
Common Cloud Management Platform (CCMP)
Operational Support Services (OSS)
Business Support Services
Infrastructure (Physical Resources)
Service Creation Tools
Security, Resiliency, Performance, and Consumability

Figure 2. 2: An overview of the IBM Cloud Computing Reference Architecture [19] [20]

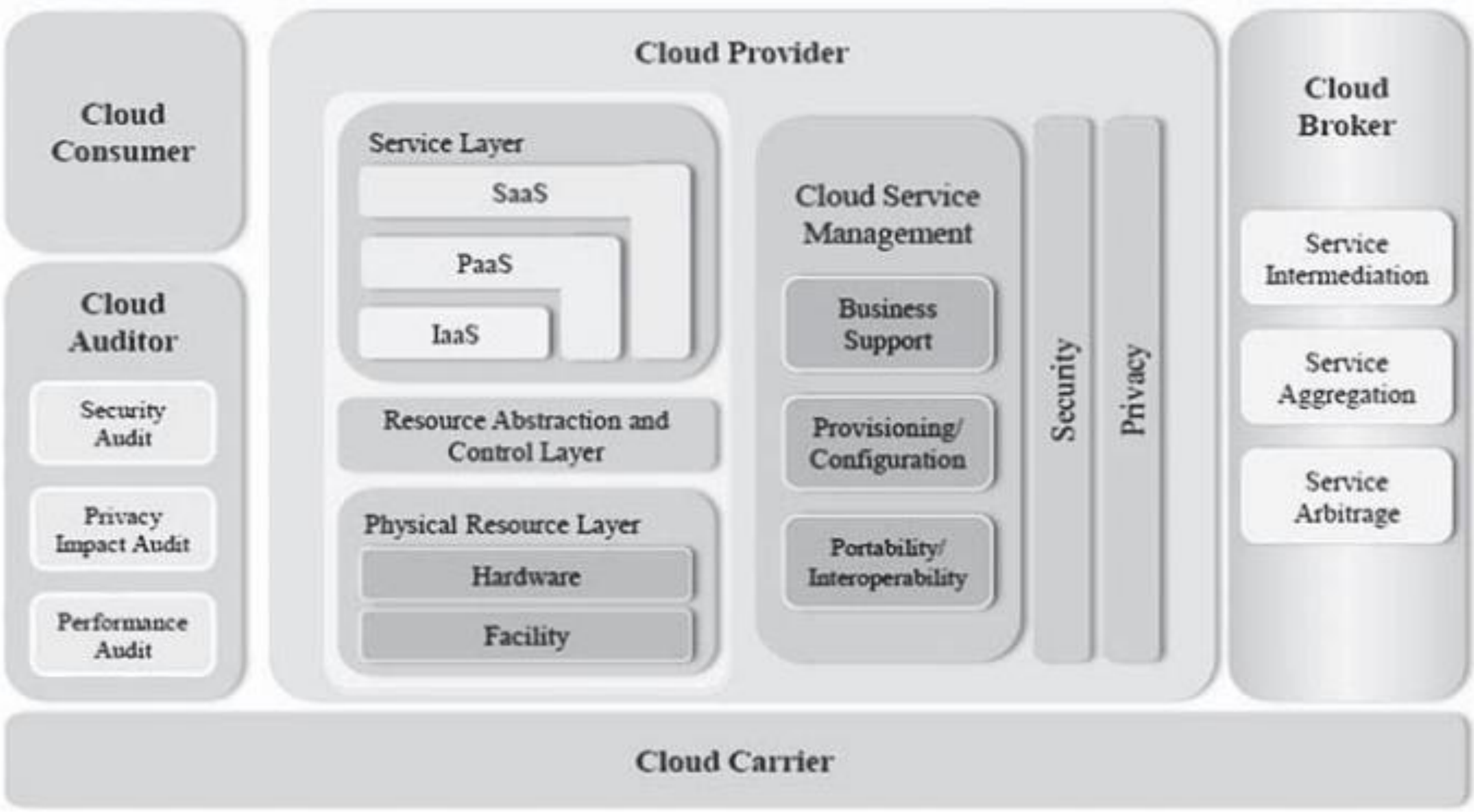


Figure 2. 3: NIST Cloud Computing Reference Architecture overview [11] [19]

2.8. Related Works

Many researchers around the world are exploring the possibilities of cloud computing, formulating new models, architectures, and cloud-based frameworks that help drive information and communication technology to implement different solutions in different sectors, such as Education, Finance, Healthcare, etc. In recent years, there have been many publications on various applications based on cloud computing. However, this section discusses some work on architecture and cloud-computing systems related to e-governance and e-office, in particular the framework for communication between government and citizens, institutions, businesses, co-operation and employers, and others Government authorities. Understand the need and benefit of cloud computing technologies in electronic office.

Rastogi, Dr.Ashish [21] , The study proposed a model-based approach for the implementation of cloud computing frameworks in e-government, to reflect the way in which the government functions in terms of the government, its relationship with its citizens, institutions, companies and the cooperation with other governments and the discussion. It reflects how to overcome the problem of e-government in developing countries.

It also discusses how to transform traditional governance into electronic governance based on the cloud and how users can easily use electronic documents and improve government services in education, health care and social empowerment. The critical issue discussed for developing countries is the infrastructure needed to implement electronic services. Use of cloud computing for better electronic governance in developing countries and proposal of a framework based on models to implement cloud computing. The main topics discussed were infrastructure development, service implementation, accessibility, security, trust and privacy, transparency, responsibility, capacity for change, permanent availability and conservation, cost structures. Apply cloud computing for better electronic governance in developing countries and propose a framework based on models to implement cloud computing and solve problems. The proposed model transformed traditional computing into cloud computing by introducing the concept of cloud computing in government offices, performing organizational evaluations, designing a cloud computing prototype, evaluating it, establishing an implementation strategy and continuously improving it.

G.Sahoo, K.Mukherjee [22], the proposed study was Cloud Computing Future Framework and Intelligent Computer Web Service by E-Governance. Adequate use of resources that are scarce day by day. Superior technology compared to the traditional architecture of server-based electronic governance. The main problem is that government services cannot easily obtain citizens. The purpose of the document was, therefore, to design the intelligent computational service model and the proposed cloud computing framework for electronic government. Address electronic government services for all categories of users, from rural citizens to citizens of the city. The researcher used the Hadoop model. In addition, it is the central part to communicate to users, access the different web services of electronic government with a laptop or sophisticated desktop, the thin clients / cell phones are unreachable for a large number of users in a country like India.

K.Veljanovska, V.Zdravevska [23], in April 2013 to analyze the proposed e-government based on cloud computing, and the possibility of using cloud computing for e-government in the Republic of Macedonia. The key points of the study are e- government in cloud computing, which uses type of cloud service delivery and model. In addition, the e- government service categories such as G2G, G2E, G2B, and G2C. By conducting this study, the Republic of Macedonia would be the leader in Europe through e-services offered by the state. In addition, it has the advantage of increasing productivity, changing the nature of work and boosting the economy, sharing information with citizens and businesses, supporting budget management and decision-makers, reducing efforts to provide services and those of them provided resources to effectively use services. With the introduction of cloud computing through e-government, all government services are virtualized.

S.Suchaiya, S.Keretho [24], in 2014, research will focus on the analysis of national e-government interoperability frameworks. Many countries have been actively involved in the development of interoperability for the electronic exchange of data and transactions between government agencies in order to provide better public services to their citizens. Many of these countries have set national policy frameworks, often referred to as "electronic government interoperability frameworks" (electronic GIFs). However, most of these e-GIF frameworks have not adopted the holistic concept of Enterprise Architectures (EA). The document proposes a benchmarking methodology to propose further improvements to EA-based interoperability frameworks to better promote the effective development of smart and connected e-government services.

The purpose of this systematic comparison is to identify opportunities and make suggestions for further improving the framework and methodology of e-government in general. They choose to compare themselves to the US FEA because the United States performs best in planning, budgeting, implementing, monitoring, and measuring international development programs. Online administration, its remarkable features, and generate ideas for future improvements. TH e-GIF, which has been developed since 2005, is a national policy framework and methodology for promoting the collaborative and development of e-government services for smart and common public services for citizens and businesses. A strong lead agency, including membership and stakeholder engagement, an effective agent-overlapping and collaborative platform of all these key stakeholders should be established, maintained, and kept in constant interaction. Current business processes (as they are) are analyzed and redefined to achieve improvement goals / visions.

Future (proposed) processes should be simplified; more efficient, safer, new processes need to be reviewed, refined, and accepted by key stakeholders. Through harmonized data elements and document simplification, technical development teams can successfully create data models, electronic documents, and messages to better serve citizens and businesses. Service functions, called application architectures, must also be designed and accepted. It provides a plan for describing services and features. This plan covers the various subsystems and components of software solutions, their relationships, and interactions with the core processes of the business for government business users and agencies.

Architects use guidelines, standards, and principles to define the direction of the future and to suggest the implementation. In addition, the implementation includes the development and use of central facilities and infrastructures. Local authorities to support the development of their system architectures can use central facilities. There is no clear measuring methodology in the TH e-GIF; therefore, a framework and measurement methodology, including risk management, should be developed and used as part of the new version of TH e-GIF.

To achieve greater effectiveness of TH-GIF, the Cabinet should adopt, legalize, or enact a policy or regulation for all government agencies to adopt the Enterprise Architecture concept and e-GIF TH for all projects with the performance measurement.

M.Kumar, R.Bhatt and K.Singh Vaisla [25], in 2015, the main objective of this document is to create or implement the central e-governance framework that is beneficial or profitable for the

Indian government. Through this model, all government organizations and agencies will effectively and conveniently exchange data and information. The purpose of this model is the use of ICT infrastructure, providing services between architectural models of e-governance and providing central connectivity to the various government departments of the Government of India. The aim of the proposed model is a faster and more reliable exchange of information, integration between ministries and the best communication between all departments.

The central architecture framework for electronic governance is the interactive model that integrates India's center, state, block center, and entire gram panchayat, providing a better information delivery service between the central and state levels. Under this approach, communication between the center and the state will be flexible, cost-effective, and provide the best connectivity for all Indian government departments. The purpose of this framework is to centralize the entire IT department in an organized framework and for e-governance, reducing the initiative's integration method.

This architectural framework for the e-governance system covers cities and rural areas, making it easy to connect the user nationwide and integrate the national e-governance plan with the ICT infrastructure and e-governance process, electronic authentication. This framework provides useful coordination between the center's office to display and increase the district, block, or gram panchayat in a cost-effective manner and reduce the integration problem. All departments in India communicate effectively with a single framework and reduce communication costs.

To summarize the related research, all the above researcher has not touched on the problem of the electronic office platform to allow them to routing different documents, and how to solve the manual preparing of forms that most frequently uses in office work becomes. None of the research develops the architecture and framework to run and access cloud-based electronic office services anywhere. To this end, we proposed to develop a cloud-based electronic office framework that enhances communication between the document-sharing offices, and basic processing, preparing automated form preparing, messaging and collaboration, institutional knowledge sharing in Adama Science and University of Technology. In addition, several researches have been conducted to enhance customer satisfaction, improve customer services and customer service provision. These researchers list several problems and propose solutions for that problem as we have seen above. As we have seen from the papers there is a gap like selection of cloud platform, service provider, the framework or architecture to solve the problems related to office tasks or to communicate government with employee (G2E).

Table 2. 1 : Comparison of the related paper and the proposed cloud based framework

Title	***Achievement***	***Gap Analysis***
A Model based Approach to Implement Cloud Computing in E-Governance. ***Rastogi, Dr.Ashish***	To design the model of cloud-based E-governance and to implement E-services.	• The researcher did not consider on cloud framework and Cloud computing platform selection.
Cloud Computing: Future Framework for e-Governance. ***G.Sahoo, K.Mukherjee***	Design the model of intelligent computational service and cloud computing framework of E-governance.	• The researcher does not consider how users access E-governance web services from the intelligent systems.
E-Government based on cloud computing. *K.Veljanovska,* *V.Zdravevska*	Analyze the possibility of using the cloud computing for the E-government in Republic of Macedonia.	• The researcher does not consider the cloud platform selection and the services deliver for users.
Central Architecture Framework for e-Governance System in India using ICT infrastructure. ***M.Kumar, R.Bhatt and K.Singh***	Model most of government agencies and organizations will interact effectively and conveniently share the data and information.	• Not specified cloud computing architecture, model, and platform. • A proposed cloud computing architecture, and service delivery model.
Proposed Framework: Cloud based E-office Framework for ASTU: (own work)	Design the E-office framework for ASTU and prepare the prototype developed the service request and service fulfillment.	• Most service delivery infrastructures are not implemented. Therefore, it takes longer time to evaluate and finalize all functionalities of the system.

Chapter Three: Research Methodology

3.1. Introduction

The Cloud-based e-governance and E-office frameworks were studied to propose the highly secured and scalable cloud-based E-office framework for ASTU to transform the current manual and semi-computerized documentation to highly sophisticated electronic offices at ASTU. We started the study by conducting interview and questionnaire with University Higher Managerial Office, Faculty Deans, Administrative and Academic Departments, Research Dean, ICT Professionals, and Teachers in ASTU to collect data about the current information on the process of office task specifically on how to send/receive, manage and routing of different documents and letters, different services form based request, and approval/rejecting processing, collaboration and messaging among different employers of the institution and related to institutional knowledgebase repository.

Based on these steps, a design of cloud-based e-office framework and its prototype were developed to address the problem in (Figure 3.1).

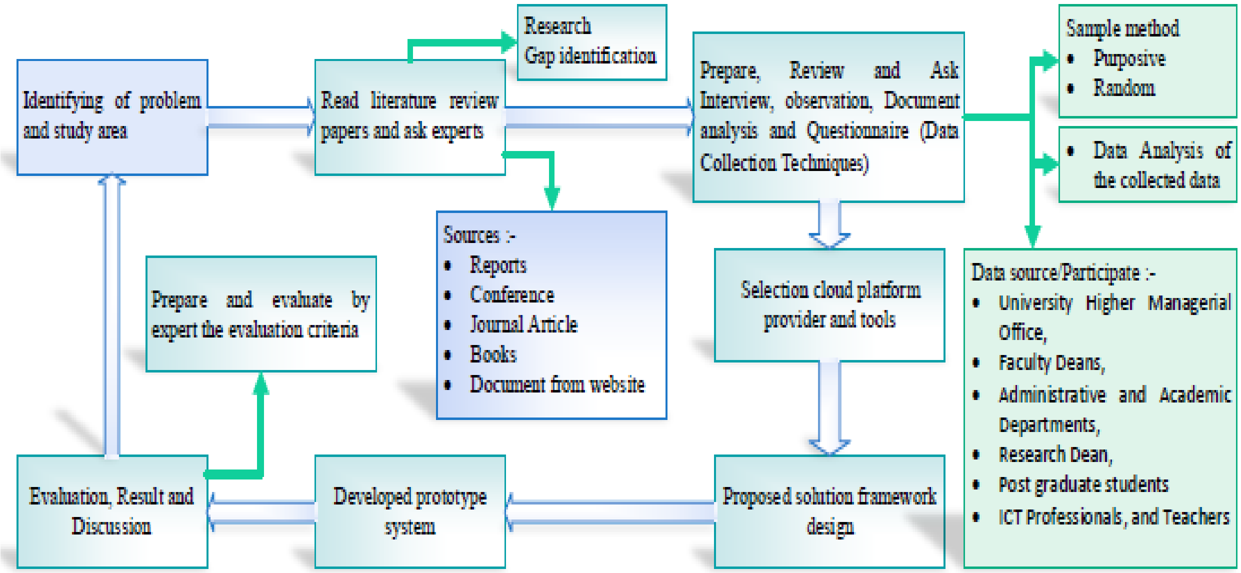

Identifying of problem and study area
Read literature review papers and ask experts
Research Gap identification
Prepare, Review and Ask Interview, observation, Document analysis and Questionnaire (Data Collection Techniques)
Sample method
• Purposive
• Random
• Data Analysis of the collected data
Sources :-
• Reports
• Conference
• Journal Article
• Books
• Document from website
Prepare and evaluate by expert the evaluation criteria
Selection cloud platform provider and tools
Data source/Participate :-
• University Higher Managerial Office,
• Faculty Deans,
• Administrative and Academic Departments,
• Research Dean,
• Post graduate students
• ICT Professionals, and Teachers
Evaluation, Result and Discussion
Developed prototype system
Proposed solution framework design

Figure 3. 1 : Research Methodology

3.2. Research Design

For any research studies, there are four commonly used approaches, which are quantitative, qualitative, design oriented and mixed method [26]. The quantitative research approach uses standardized measures, numerical values, have large sample size, and analyze data using statistical program. It is applicable to phenomena that can be expressed in terms of quantity or number [27]. On the other hand, qualitative research approach is focused on describing situations, phenomenon, problem, or event; tends to be more in-depth and have smaller sample size [28]. A mixed methods research design is a procedure to analyze, collect, and mix research and quantitative and qualitative methods in a single study to understand a research problem [29].

Design-Oriented Research creates an artifact by artificial object and its basic objective is producing new knowledge during this process. It is mostly focus on knowledge flows and process steps that is problem awareness, suggestion, development, evaluation, and conclusion; eventually aims to get result or output [30].

In this study, a combination of qualitative and design oriented research method were used for data manipulation and to evaluate the functionality of the prototype. To demonstrate the usefulness of the proposed e-office framework for ASTU and their employers, the design-oriented research methodology was used in this study. Design-oriented research method is a problem-solving model [31]. The five steps of design-oriented method followed in this work are identification of problems and requirement, framework design, framework evaluation, revaluation and improvement of framework, communication, and discussion of research.

3.2.1. Identification of Problem and Requirement

Currently, various documents and forms-based inquiries are distributed, which is messaging and collaboration between employers from one office level to another office level than different Departments / Units, Academic School / Faculty, Senior management office and other offices of this form on request, and workflow not electronically available. Moreover, there is no central institutional knowledgebase, no effective way of working together, and the messaging channel. In general, the tasks of the university office such as communication, the exchange of documents and various inquiries are not effective and it takes a lot of time. To solve these problems, the requirement or data collection was the basic step to design the proposed framework after identifying the problem.

In this study, the required data and user needs were collected using a variety of methods, including interviews questions in Appendix A, with Department Heads and Faculty Deans, ICT professionals, Research Dean, and Teachers. Observation with the study area and distribution of questionnaires with questions about office tasks and problems encountered. Then the collected data was organized and analyzed to use for the design of the framework.

3.2.2. Framework Design

The design of the E-Office framework focused on the elements or components from the questionnaire and interview in Appendix A and B, an of the framework to achieve the goals given under the given constraints and limitations. To design the proposed framework, reference architectures, models, frameworks, a web application architecture for the Microsoft platform, and other Internet resources were used.

3.2.3. Framework Evaluation

In the software development process, the framework can be evaluated at the early stage or later stage. This is used to compare and identify gaps in various alternatives during early design stages. There is different software evaluation architecture that includes experience-based, simulation-based or prototyping and scenario based [32]. In this study, the early assessment phase and prototyping are good, as the cost of remedying an error during the requirements was determined. The evaluation and improvement of the e-Office framework for quality attributes set as targets has been reviewed with the help of software development experts, systems development companies, and selected cloud technology experts, as presented in Chapter 7.

3.2.4. Re-Evaluation and Improvement of Framework

After the performance review, the skill set and users to add features and functionality to improve the proposed framework acquired the functionality and its workability, comments and suggestions.

3.2.5. Communication and Discussion of Research

Communication addresses the problem that can be solved and how the system works, communication plays an essential role. To answer these and other questions, help, and guidance have been prepared for the end users. While the quality characteristics of this work depend on the software architecture, the architecture should be clearly presented to all users. The discussion of this work focuses on quality frameworks to be secure, scalable, interoperable, portable, and reliable.

3.3. Process Model

In software development, the process model describes the order of activities performed in the software project and the relative order of those activities. Several process models include Prototyping, Unified Process, Incremental, and Agile. We can rarely combine process models or use together in software development process [33]. In this study, prototyping model were used to achieve the overall objectives.

Prototyping one of the development processes models to improve the design and execution of software projects by developing executable software systems (prototypes) for experimental purposes. It is very suitable to gain experience in new application areas and to support the development of evolutionary software [34].

Therefore, prototyping is a process model that provides a solution to the problem, and it is the practice to create an early version of a system that shows certain features of the actual system. There are two basic types of prototyping, which are throwaway, and evolutionary prototyping. In throwaway prototyping, the various parts of the system are developed and then ignored after demonstrating the basic functions that the system will perform. Therefore, the final system to be delivered will be different from the prototype.

However, evolutionary prototyping technology implements the prototype of the system, demonstrating its basic functions, and will ultimately be part of the final system that can be delivered. Therefore, we used the evolutionary prototyping method for the proposed study.

3.4. Data Source

The data used as input for this work were collected from the following sources:

- The primary data source is to use interviews and questionnaires in ASTU, which include Senior Executives, Faculty, and Heads of Administrative and Academic Departments, Teachers, Research Dean, Post-Graduate Students, and ICT Professionals working at the university.
- The second data source in this work is a comprehensive literature search and document analysis from the Internet. Therefore, the Internet is one of the data sources, as explained in the methodology section in the first chapter.
- The other data source in this work was the observation. During this study, observations were used as a means to gather the required data from office tasks, specifically to know the most

repetitive and paper-intensive process, delays in transporting documents and letters and understand coordination across departments, and too many manual approvals and checks.

3.5. Sampling Techniques

The sample selection includes the selection of individuals from Post-Graduate Students, Departments / Units, ICT Professionals, a Research Office, some Senior officers, and a Faculty from a defined area of study. There are different sampling methods in qualitative research to study the research participants. This includes purposive sampling, snowball sampling, quota sampling, random sampling, convenience sampling [35].

In this study, purposive and random sampling technique was used to collect the qualitative data. Therefore, data collection was started by identifying the participants using purposive sampling technique. Before distributing the questionnaire and the interview, the structure of the university must be identified to determine the total number of Departments / Units, Persons in each V/Presidential office.

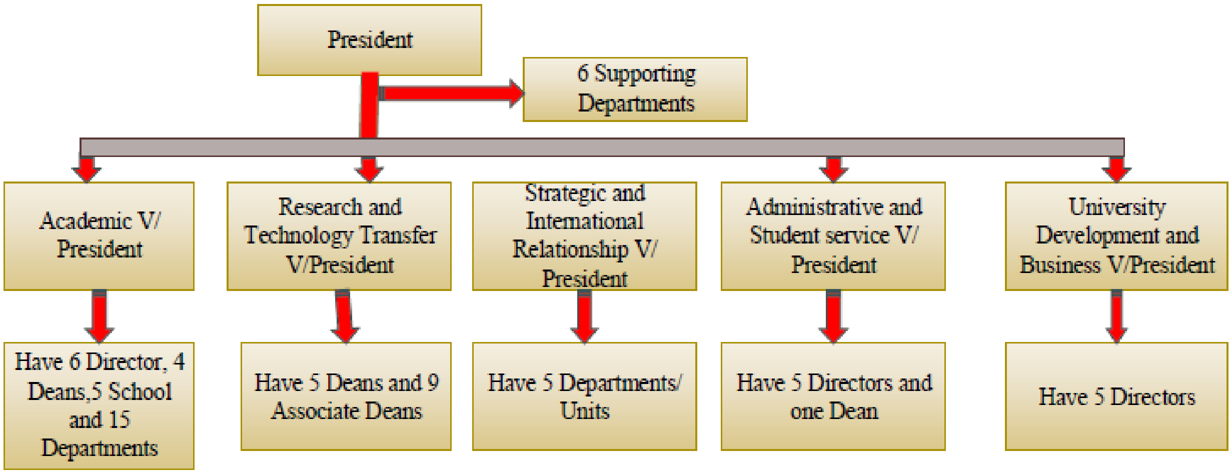

President
6 Supporting Departments
Academic V/ President
Research and Technology Transfer V/President
Strategic and International Relationship V/ President
Administrative and Student service V/ President
University Development and Business V/President
Have 6 Director, 4 Deans,5 School and 15 Departments
Have 5 Deans and 9 Associate Deans
Have 5 Departments/ Units
Have 5 Directors and one Dean
Have 5 Directors

Figure 3. 2 : Organization Structure of ASTU

As shown in Figure 3.2 above, the organizational structure of ASTU under the President have six other supporting departments of the university. In addition, under the President has five V/President like: - Academic V/President have 6 Directors, 4 Deans, 5 School and 15 Department and supporting units and Research and Technology Transfer V/President have 5 Deans and 9 Associates Deans and Strategic and International Relationship V/President have 5 Divisions / Teams and University Development and Business V/President have 5 directors and Administrative and Student services V/President has 5 Directors and 1 Dean.

The reason questionnaire distributed sampling from 6 Higher managerial officers and supporting Department is 5, 10 Administrative Units is 6, 5 School Deans is 3, 15 Academic Department Heads is 5 and from 5 research deans taken 1.

The sampling was selected by personal contact for the interview and the questionnaire based on the appropriateness of the study; the knowledge they have, and the technology, and ICT professionals such as School of Electrical Engineering and Computing, Post Graduate Students, and Computing and ICT staff. In addition to this sample selection, they are also used for the other departments and officers, as most of them have a similar status to our research interview and questionnaire. In general, 55 taken total number for questionnaire that selected in selected of each department, and from total population.

As shown from Table 3.1, from question number 1-5 Higher Managerial officers, Directors and Deans selected randomly and question number 6-8 selected based on purposive sampling. Stratified sampling calculator used to determine the sampling size from total population [36]. Higher managerial office =1 %, Administrative Units= 1%, School Deans= 0.5%, Academic Department Heads= 1.5%, Research Deans= 0.5%, ICT Professionals=4.1%, Teachers=41%, and Post Graduate Students =5.6%.

Table 3. 1 : Sampling Population Size

No	***Sample Population***	***Total No. Departments/ Population***	***Distributed Questionnaire***	***Returned Questionnaire***	***Returned in (%)***
1	Higher managerial officers and supporting department	6	2	2	100
2	Administrative Units	10	4	3	75
3	School/Faculty Deans	5	1	1	100
4	Academic Department Heads	15	4	4	100
5	Research Dean	5	1	1	100
6	ICT professionals	40	10	9	90
7	Teachers	400	18	15	83.3
8	Post Graduate Students	55	15	12	80
	Total	536	55	47	91

As we have seen from the above Table 3.1 the response rate from the respondent around 91 %. Therefore, there is unreturned questionnaire from different respondents.

3.6. Data Collection Techniques

Research differs in a number of ways, but they have some similarities. One of the common aspects is the need to collect data. The interviewed experts came from ICT professionals, Computing and Electrical Engineering staff. This allows us to correctly capture the required data and simplify the process of data collection.

This data collection task involves three phases:

(1) In depth interviews interviewed a total of seven ASTU staff members, two of ICT professionals, two of the academic and administrative departments, three of teachers on their office task ideas, their indispensability, efficiency and effectiveness, log file and status of the transaction, current situations, barriers and solutions to provide it continuously. About cloud computing include in the interview questions.

(2) The questionnaire addressed a number of issues related to ASTU office tasks for different employers and postgraduate students.

(3) Observation tasks in Appendix C: Observation on Services, observed based on workplace study on ASTU office tasks in different offices like in Computer Science Engineering Department, Registrar, Academic V/President, and ICT office for 4 days. In addition, we are part of a higher education institution and we observe the customer satisfaction on current manual system as a way to manage and process of various documents, current communication systems between departments, faculty and senior officials and the repetitive form based inquiry, its efficiency, and effectiveness of the current system, technology adaptation and the knowledgebase. This is done through different techniques, but getting the right information was extremely difficult, as observed.

Using the questionnaire, we examined the work efficiency and effectiveness of the current manual system, the type of delivery services, the problems they face with the current system, and the possible solutions to the current problem and employers interest. It would be E-office in Cloud-based implemented. As well as their use of ICT and other technologies in their activities specifically related to the office task. Then we have added questions to interest them when the office task is cloud-based and is supported by current technologies to handle various requests, and communication between Department, Faculty, and senior officers using various communication tools such as E-messaging, E-forum to conduct discussion. We collect data about these concepts because they help us to build the foundation for the design framework.

3.7. Data Analysis

Questionnaire and interview are a structured technique used to collect primary data to decide how to collect replies, questionnaire design, and analyze data after identifying the population and sample [37] [38] [2].

The techniques are focused on the importance of creating a smart office environment for the ASTU community, the efficiency, and effectiveness of the current manual system and how to forward the paper documents, currently how they are deployed, and what efforts have been made to make it available to you one after another. In addition, how the institution handles a variety of request, form-based tasks related to technology, collaboration, and knowledgebase environment.

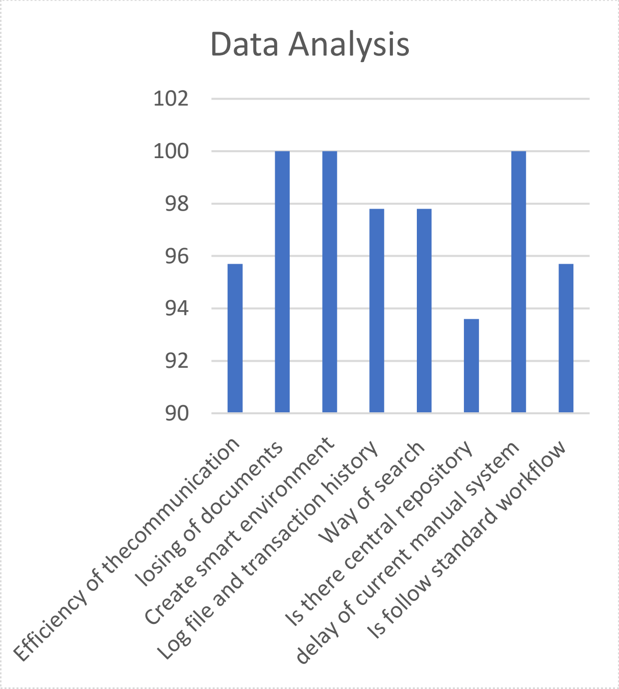
To know how many the university employers and post graduate students responded. The total number of in the Appendix B: Questionnaire Questions, distributed 55 and 47 returned from respondents.

Table 3. 2 : Response rate of the questionnaire

No	***Questionnaire List***	***Yes***	***No***	***No, in (%)***
1	Efficiency of the current communication system.	2	45	95.7
2	The losing rate of different documents and letters.	47	0	100
3	Is the paperless office system working place create smart environment?	47	0	100
4	Is there any log file that can hold the transaction history on your side to know the status of delivery for your request?	1	46	97.8
5	Does the current system support an easy way to search or find documents and letters?	1	46	97.8
6	Is there any automated office management system in your university to route or send/receive documents and to process form based request?	2	45	95.7
7	Do you believe to improve your effectiveness and efficiency, if E-office management system is implemented in your university?	46	1	97.8
8	Is there any central repository of institutional knowledge base and best practices?	3	44	93.6

9	In average, is there any delay of the current manual system in movement of file level and to give decision-making?	47	0	100
10	Can you say the current manual system is following a standard workflow the given task?	2	45	95.7
11	Is there any way to report/complain if a task is delayed or does not performed in time?	3	44	93.6

The ASTU community found rough understanding and the situation regarding the current manual system, its needs, and the associated problems. As they explained, to manage the various documents from one office level to another office, widely distributed by e-mail, him/herself, colleague / friend, telephone, and personal messenger. In almost all cases, paper documents are used for requesting decisions from the other media of their supervisor, such as telephone and e-mail, to require decisions based on the sensitivity of the decision. The use of e-mail for official communication is minimal, only 4.3% believe that the current communication system is efficient in their university.



The problems of the current communication system have also become ineffective, time consuming, and expensive, with security issues, delaying communication in the case of a personal messenger, and some people may not be interested or need to read emails, various tasks that are not perform on time, high paper used and it has the unavailability or absence of senior executives. In addition, most respondents are interested in an intelligent environment or a paperless workplace by automating the workflow / business process and implementing online communication such as E-messaging.

In general, the current manual system for analyzing data from the Appendix B: Questionnaire Questions, they believe around 97.8%, the current manual handling is not an easy way to search documents, there is no log file containing the transaction history, and there is difficulty to know the status of the delivery request, there is no automation office management system, but there are similar systems in the university such as tracking maintenance system, ICT support service system, and campus e-mail service.

And the respondents believe that around 97.8%, if the E-office management system implement improves the efficiency and effectiveness of office work such as: effective storage and accessibility, efficient document tracking features, it can easily manage, share and disseminate various documents with colleagues from a centrally stored repository, improve the delivery of files, reduce delays, and have good communication between employers from one level to another.

At the office level, the type of communication becomes "SMART", which reduces the decentralization of information, reduces the workload and wastage of resources, facilitates tasks and easily responds to decisions and increases employer satisfaction, no messengers can lose their time and energy, have transaction history, to have automatic notifications to know the status, creating an intelligent environment, saves time and avoids non-standardized work, no lost documents, most of the tasks will be supported by electronic technology, makes use of business intelligence feature of the E-office and users IT skill will be upgraded.

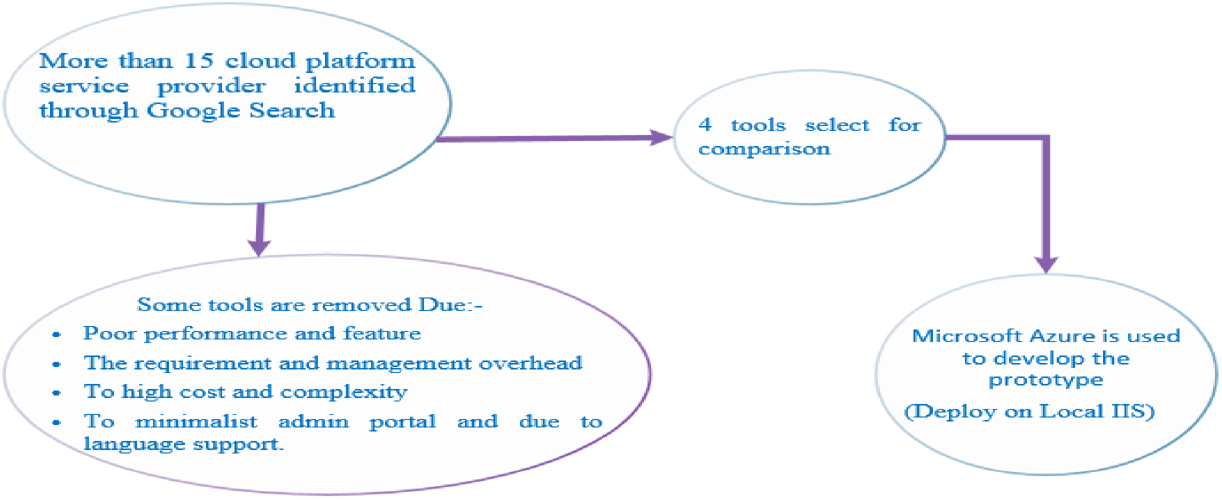
According to the Appendix A: Interview Questions Appendix B: Questionnaire Questions Appendix C: Observation on Services respectively, the majority of respondents stated that, on average, there is no central repository of the university. To manage the documents takes within two days or until a week, and preparing of forms, and there is no way to report or file a complaint, but some of them claim that it is possible to express their claim verbally, request the manager and finally the respondents recommends to implement e-Office system. Based on the finding, customers are not satisfied on services.

As the Appendix A: Interview Questions Appendix C: Observation on Services respectively, indicates the number of documents moving from one office to another depends on the department, but in average, it can process up to twenty letters and documents in one day. In addition, we have good information about the institutional structure and how they can request the repetitive process: such as overtime requests, notification of absence, annual leave, transfer of material, purchase of needed material, travel handling, file request and other forms request and how various services are handled.

Chapter 4: Selection of Cloud Platform Service Provider and Tools

4.1. Introduction

This chapter provides an overview of the four-selected cloud platform service providers and in detail the Microsoft Azure platform for this work. To develop the prototype of the proposed framework, we started to find the best cloud platform as a provider service. As a result, we have many tools available to do an advanced search, and we have just selected four tools for comparison and analysis to use the best for our purposes. The chapter also contains a selection of tools to design the proposed framework and to implement its prototype for the purpose of concept verification and evaluation purposes.


More than 15 cloud platform service provider identified through Google Search
4 tools select for comparison
Some tools are removed Due:-
• Poor performance and feature
• The requirement and management overhead
• To high cost and complexity
• To minimalist admin portal and due to language support.
Microsoft Azure is used to develop the prototype
(Deploy on Local IIS)

4.2. Analysis and Comparison of Cloud Application Development and Hosting Platforms

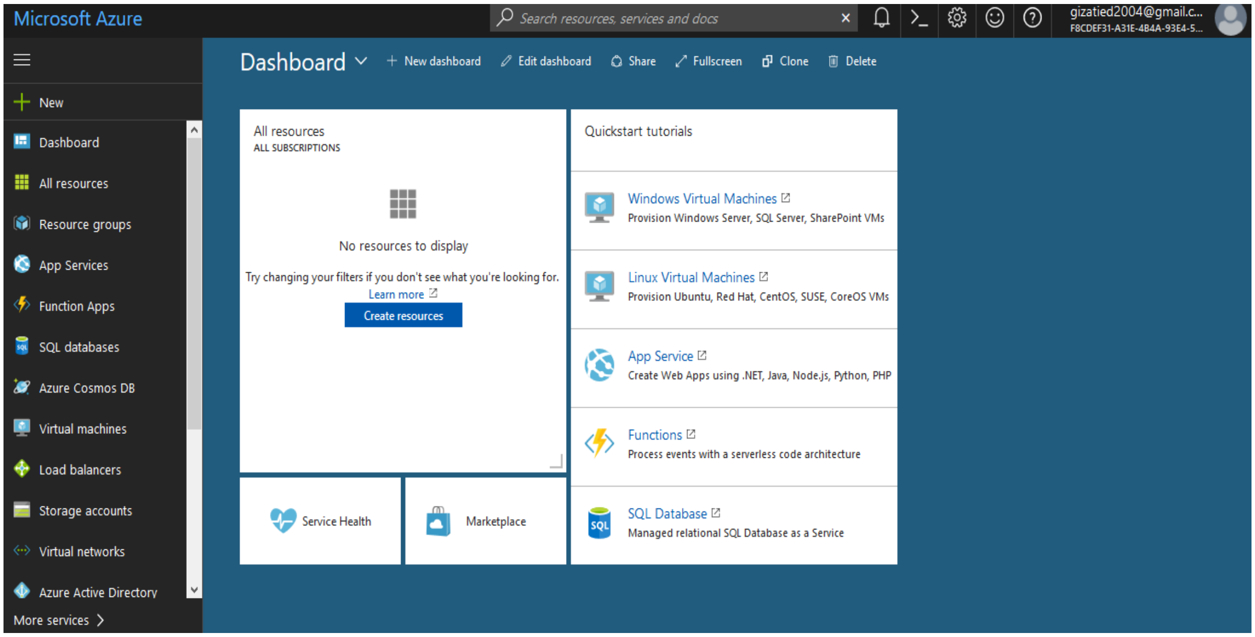
PaaS is a cloud designed for software developers to develop web and mobile applications. In this study, we focus on the platforms used for web development and hosting. There are many tools for application developers in the cloud environment, and we try to describe the most popular platform providers and their development and hosting tools.

We identified about fifteen tools, and we declined some tools based on their adaptability for our proposed work, we selected the six tools, such as Microsoft Azure, Amazon Web Services, Google App Engine, OpenShift, Engine Yard, and Appfog. From these tools, we selected four tools for comparison and analysis based on familiarity, and most dominate used with our study.

When comparing the platform providers and their tools described below, we used seven criteria, such as: Provider, GUI support, middleware technologies, programming language, and software framework, native core services as part of the offering, pricing model, and tool availability. The list of cloud application development and hosting tools compared in this work is as shown in table 4.1.

4.2.1. Microsoft – Azure

Microsoft's Azure platform is a group of cloud technologies, currently being engineered and under development to enable usage of massive computing power packed within Microsoft's data centers around the world. The main components of Azure platform are .Net services, Windows Azure, SQL storage, and live services. In addition, Azure is Microsoft's cloud platform that offers the combination of IaaS and PaaS at one to suit the needs of customers, which is a foundation for running applications and storing data in the cloud. It enables developers to use any language, framework, or tool to build applications [39] [40].



No resources to display

Try changing your filters if you don't see what you're looking for.

Windows Virtual Machines: Provision Windows Server, SQL Server, SharePoint VMs

Linux Virtual Machines: Provision Ubuntu, Red Hat, CentOS, SUSE, CoreOS VMs

App Service: Create Web Apps using .NET, Java, Node.js, Python, PHP

Functions: Process events with a serverless code architecture

SQL Database: Managed relational SQL Database as a Service

Figure 4. 1 : Microsoft Azure Portal

4.2.2. Google - App Engine

The App Engine is Google's platform designed for web applications and supports different languages such as Java, Python, Go, and PHP allows for integration to other technologies such as

Hadoop, MongoDB and others. It uses a sandbox model, which isolates processes thereby reducing the risk that a rogue process on a physical server will disrupt operations of other processes on that server. Google App Engine like other platform manages many of the implementation details about maintaining virtual servers, operating systems, and development tools. Nevertheless, it is restricted to four programming languages, which is its limitation.

App Engine support of more languages will appear in the near future and is the SDK which completely emulates Google’s server environment. This is useful because customers can create their applications locally, in real time, without uploading them. Then it comes the web based admin interface. This interface contains all the necessary tools to manage and debug user applications and isolates processes thereby reducing the risk that a rogue process on a physical server will disrupt operations of other processes. Finally, the Data store API, which is the Google’s scalable persistence layer [40] [41] .

4.2.3. RedHat – OpenShift

OpenShift is a cloud development Platform as a Service (PaaS) hosted by Red Hat. It is an open source cloud-based user-friendly platform used to create, test, and run applications, and finally deploy them on cloud. It is capable of managing applications written in different languages, such as Node.js, Ruby, Python, Perl, and Java. One of the key features of OpenShift is it is extensible, which helps the users support the application written in other languages. In addition, which offers an application development environment for developers and provides two options for developers to interact with OpenShift such as through a web console or the command line client tools as well as developers can interact with OpenShift through an integrated development environment. Therefore, this platform is highly customizable and offered in three forms: OpenShift Origin (the open source application hosting platform), OpenShift Online (a public cloud-based hosting service), and OpenShift Enterprise (a private PaaS that runs in your data center) [42] [40].

4.2.4. Engine Yard

Engine Yard is a PaaS designed for Web application developers using Ruby on Rails, PHP, and Node.js who want the advantages of cloud computing without all the operations management responsibility. Moreover, it provides a set of services on top of Amazon AWS. In fact, Engine Yard runs its platform in the Amazon cloud, so the value of the PaaS rests more with orchestration and management than with providing software components and takes care of key operations tasks such as managing snapshots, performing backups, administering databases, managing clusters, and load balancing [40].

Table 4. 1 : Comparison among the four cloud-application development and hosting tools

Features	***Azure***	***App Engine***	***OpenShift***	***Engine Yard***
Provider	Microsoft	Google	Red Hat	Engine Yard
GUI Support	Windows Azure portal and command line	Command line interface	Web console interface and command (RHC client tool)	Portal and Command Line tool
Middle ware	Tomcat java 7	--------	Zander server Tomcat and J Boss	Tomcat, Ha Proxy, passenger
Frame work	Cake PHP, Django, etc.	Django, Webapp2	Drupal(PHP), Switch yard(java0.8), vert. x(java2.1), Django	Grape, MERB, Rack, Rails
Language support	.NET, Java, PHP, Node.js, Python, and Ruby	Java, Python, Go and PHP	Java, Ruby, PHP, Python and Perl	Java, Node, PHP, Ruby
Service	No native, databases, messaging queues and caching	Storage and development services	Data storage, Messaging, backup development, and mobile services	MySQL, Postgre SQL
Pricing	Based on size of instances running	Based on usage, monthly bills or free usage	Based on the number and size of Gears, (monthly bill/ fixed usage or free usage)	Based on a "pay as you go"
Availability	Open source: http://azure.microsoft.com	Open source: http://www.cloud.google.com/appengine	Open source: http://www.redhat.com	Open source: https://www.engineyard.com

Based on the above comparison and the analysis of Table 4.1, we used Microsoft Azure for our study to implement the prototype of the proposed architecture. Microsoft Azure automates basic systems management tasks such as provisioning, configuring, and scaling virtual servers. In addition, Microsoft Azure provides a wide range of environments (UI, command line, and .Net), flexibility, and access to various programming languages, including .Net will benefit us greatly.

Microsoft Azure provides agility and an open cloud platform that helps to grow the business with greater efficiency and responsiveness to change. With Azure, possible to quickly upgrade, scale or shrink as needed, and avoid high capital costs - you only pay for what you use.

Azure also works seamlessly with other Microsoft software and services, including Windows Server, SQL Server, Exchange, and SharePoint, to secure and protect the organization most important information. By comparing the Microsoft Windows Azure. It has a variety of languages and databases that allow the developer to make their applications easy and understandable. We can manage our code using a web portal.

The main criteria for choosing Microsoft Azure are;

Firstly, it provides the developer with more deployment options than the other three platforms and more sets of application programming interfaces (APIs).

Secondly, is that we are familiar with Microsoft products such as the Microsoft’s Visual Studio (VS) Integrated Development Environment (IDE), Angular JS, and SQL storage. Therefore, for most popular cloud service providers uses Microsoft Azure.

Third, the upcoming releases of Azure are going to support applications written in languages such as Python and PHP.

Fourth, it resembles existing Windows environment a lot. In general, the migrating applications to cloud is easy. This partially stems from the fact that Azure’s services exploited by an application whether it run locally or in the cloud. The developer can deploy locally using web deployments on IIS.

In addition, Microsoft has features like -

- High **availability** (99.9%) of the application and the data.
- **Open** and **flexible** cloud platform that enables to rapid creation, deployment, and management of applications in a global network of Microsoft managed datacenters.
- Possible to integrate public, private cloud applications with the existing IT environment and quick deployment of the application to new customers etc.…

4.3. Selection of Tools

***Programming language:* -** We used the .Net framework to develop the prototype for the following reason: -

- We are more familiar with Visual Studio than other languages and it is an integrated development environment (IDE) from Microsoft.
- Used to develop computer programs for Microsoft Windows as well as websites, web applications, web services, and mobile applications, uses Microsoft software development platforms such as Windows APIs, Windows Forms, Windows Presentation Foundation, Windows Store, and Microsoft Silverlight [43].
- It can generate both native and managed code, and includes code that supports IntelliSense (the code completion component) and code refactoring. The integrated debugger works both as a source-level debugger and as a debugger at the machine level, supporting 36 different programming languages [44].

AngularJS: - is a relatively new JavaScript framework developed by Google to enable front-end development. It is a complete solution for fast front-end development and familiar with the language as easy as possible. There are many frameworks and plugins available. Therefore, it can sometimes be difficult to scour the entire noise to find useful tools. These are the basic principles that lead AngularJS to creating an efficient, performance-oriented, and maintainable front-end codebase.

In general, various tools used to achieve this study. Some of the tools are Windows 10 operating system and Microsoft Office 2016 for the preparation of the document, Adobe Photoshop for creating images and other diagrams (Microsoft Visio 2016) used. As mentioned above, Microsoft Office, Excel, and Visio 2016 used because, used to create simple or complicated diagrams. The architecture implementation based on Microsoft Azure Platform and using the ASP.net based Microsoft Visual Studio MVC architecture. The prototype used to demonstrate the architecture implemented with Angular JS for the UI implementation and Microsoft Visual Studio (IDE) of ASP.net for the backend implementation has been developed and deployed on the IIS8 server.

Chapter Five: Design of Proposed Framework

5.1. Introduction

This chapter describes the cloud framework based on the proposed framework. In order to design this framework for the provision of the software as a service, we have added ideas or concepts from IBM CCRA [45], NIST Cloud Reference Architecture [46], Microsoft Web Application Architecture [47], and available literatures on software architecture approaches for cloud-based systems.

In addition, from the Interview, Questionnaire, and Observation data analysis or result data from respondents are used as input data to determine and identify the E-office components in the framework and architecture.

The CCRA reference architecture categorizes cloud business models and corresponding architectures based on seven cloud customization patterns. It identifies common architectural patterns for each cloud-adapted pattern that describes the technology that underlines every type of cloud implementation. The reason why we used these three reference architectures is:

1) It shows a way to construct the framework across all delivery models by providing the steps we follow and the time required to complete the components required save up. 2) It meets the required cloud requirements such as elasticity, self-service and flexible sourcing. 3) It uses comprehensive architectural principles to ensure design scalability for efficiency, security, service management and virtualization, as well as reducing errors throughout the development process.

In addition, we have used graphical elements or symbol representations of icons for cloud applications [16], which presents a collection of patterns to describe cloud computing, its service models, its deployment models and its architectural patterns in general. For this reason, they have cited three types of architectural patterns for cloud applications, such as elasticity patterns, availability patterns, and multi-tenancy patterns.

Moreover, we want to introduce a cloud-based e-office framework with two major parts: frontend and backend: which are usually connected via the internet. The first part presents the delivery institution (ASTU) and employer's infrastructures such as computers, laptops and the network used to access the cloud-based E-office management software as a service. The first part consists of the visible interface and applications (for example, web browsers like Chrome, IE, Torch, Opera, or Mozilla Firefox) that are needed to access the E-office software as a service from the cloud providers. The second part will illustrate the basic components and services of the ASTU E-office management system, which is the cloud part of the framework. These include various servers, middleware, storage and virtual machines, programming languages, platforms, E-office software as a service itself, security, tools, integration, and deployment models.

5.2. Design Goals and Constraints

The main goal of the cloud-based E-office framework design is to provide E-office software as a service that meets the ASTU community's related to office task. As a result, the framework should be flexible and easy to use, with increased productivity, a well-defined interface, and user-friendliness.

Constraints are a limiting factor and severely restrict options for making design decisions. In software architecture design, there are two types of technical and non-technical constraints or business constraints [48]. In this study, the following technical and non-technical constraints were considered.

Technical constraints include:

Programming language*:* - we have used .NET framework language to develop the prototype of the framework.

Operating system or platforms supported: - This study works properly on windows operating systems with the support of IIS local deployment.

Non-technical constraints include:

Schedule*:* - this is the time frame for final delivery of the thesis work, so we have stuck with our schedule during this work.

Software and Cloud platform service provider licensing restrictions*:* - We are restricted to using freely available software and tools to perform a work that lacks full features like Microsoft Azure platform.

Internet-based*:* - all cloud-based applications are working properly on dedicated internet connection; Due to this, slow internet connection may affect the accessibility or availability of the system. In addition, it may be affect for deployment and development purpose.

5.3. General Cloud Framework for the CBEOF

The central function of designing an electronic office framework based on the cloud will be easier to sending / receiving, forwarding/routing, and approving / rejecting documents from one office level to another office level to create an institutional repository, to design dynamic form generator for most repetitive form based request instead using manual preparation of forms, time saving and non-standard work avoidance, most tasks are compatible with electronic technology to provide effective storage and accessibility, efficient document tracking capabilities and the electronic office business intelligence function for all ASTU employers. In addition, the process of designing a cloud-based electronic office framework will be progressive. As a result, CBEOF design, according to the current state of E-Office, cloud-computing expertise, and Microsoft Azure, can learn about the following generic cloud framework, shown in Figure 5.1. It contains the components listed below.

Physical resource pool: These include hardware resources such as network, storage devices, and virtual servers. These are linked by network virtualization with the intention of creating cloud-centric university knowledge-sharing networks and sharing various documents and letters as well as various institutional forms.

Infrastructure as a service (IaaS): The delivery university (ASTU) and end users can share the full hardware resources by virtualizing physical resources. Therefore, virtualization technology increases user demand by dynamically allocating resources to active demand, thus avoiding the use of fixed storage space. It is fully managed by the cloud provider and manages the resources and environment of physical infrastructures such as servers, storage, and networking to the virtual machine.

Platform as a service (PaaS): It contains the Microsoft Azure platform, which will control the development environment to enable creation and hosting of applications and programs with the ability to access private cloud resources and platforms.

Software as a Service (SaaS): End users order software as a service provided by the platform service providers over the network. The service providers collect payments based on used services. The SaaS's cloud based electronic office in ASTU have various services such as service delivery infrastructure, electronic office service such as: electronic annual leave, tour, material purchase request, send / receive and forward/route documents and letters, manage institutional knowledgebase, dynamic form generator, E-forum discussions and messaging, dispatching and escalation engines, design workflow, alerts and notification, SMS and E-mail exchange can be provided at this level.

Microsoft azure: As we have described in chapter four, it is Microsoft Azure offering. This platform used to create and deploy cloud-based electronic office system. On the E-office provider (ASTU) center, there are individual level, university / department level, and system level electronic office components, which will adapt different approaches.

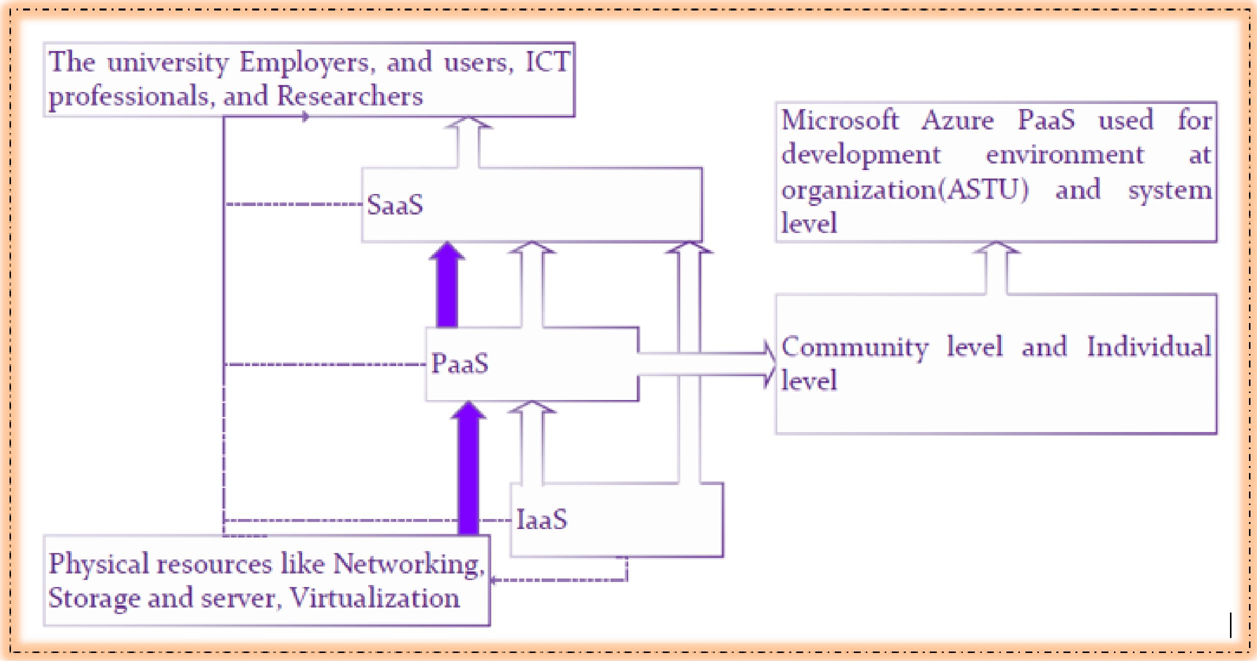

The university Employers, and users, ICT professionals, and Researchers
Microsoft Azure PaaS used for development environment at organization(ASTU) and system level
SaaS
PaaS
Community level and Individual level
IaaS
Physical resources like Networking, Storage and server, Virtualization

Figure 5. 1 : General Cloud Framework of E-office for ASTU

5.4. The Conceptual Framework of Cloud Based Electronic Office

Due to several factors, it is true that in their current situation, using ICT in Ethiopian Higher Education Institutions is not efficient, especially in office tasks.

Therefore, in this study, we proposed a cloud-based electronic office framework. To get the layer first, we analyzed the cloud taxonomy and got a classification that relates to cloud computing like service type, middleware, and others. However, we do not know the relationship between classifications; we reviewed existing cloud computing architectures, including the reference architecture for cloud computing, the IBM cloud computing reference architecture, the NIST cloud computing reference architecture, and the Microsoft Web Application Architecture.

The study proposes the following architecture to implement cloud computing as a tool and access the electronic office for ASTU. In addition, it is designed to realize the electronic office components, design workflow, integrate with other systems, use tools, and provide a worldwide electronic office to perform various service in the University.

Cloud-based E-office framework in ASTU includes the following features.

Application independence: to ensure that access to resources is not dependent on a single application.

Platform Independence: to ensure that access to resources is not limited to specific hardware platforms.

Long Term Access: to ensure that resources can be obtained and retrieved over a long period of time.

Architectural Integrity: to ensure that the architecture for IT development is robust and can evolve in the future.

The institutional records and standards should be followed as well as authentication and role-based access in accordance with the institution's security standards.

Figure 5.2 shows in below, the framework for using the Microsoft web application and the cloud computing architecture. The framework is follows the overall Microsoft architecture for web application in which users and end-user devices interact with electronic office applications for the ASTU community that uses the cloud. After the users have requested in the presentation and the application layer, request/response from the core services and the data layer based on WebAPI. Similar to that, the data retrieved from data layer to user interface request. But,

Figure 5.5 shows that, the conceptual system architecture of the service requesters and service fulfillment (service respondents) via cloud and to communicate the service delivery infrastructure must be prepare firstly.

In addition, **Figure 5.3** shows in the below that the architecture of the proposed system. Moreover, the main difference between the framework and the architecture of the proposed system is that the framework is the general concept that contains the various tools, e-office components, integration, security, and other external systems to integrate with proposed system. Nevertheless, the architecture of the system is concern with detail components of the electronic system.

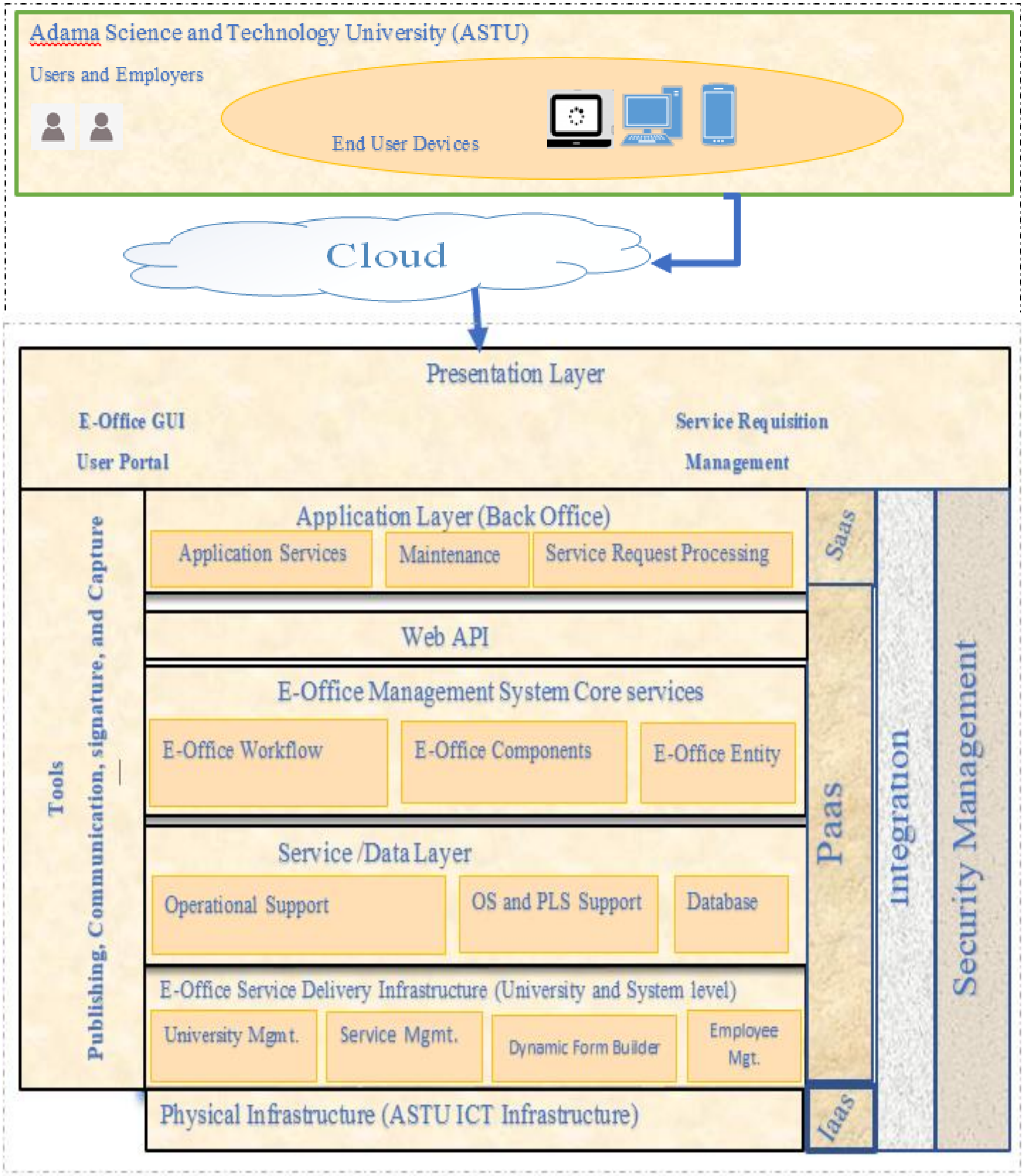


Figure 5. 2 : Conceptual Framework of CBEOF for ASTU

Figure 5.2 is own work, it consists of four layers managing the E-office components and the infrastructure for the provision of services. In the CBEOF, users / employers can request E-office services from the cloud provider. The details of the CBEOF layers are described below.

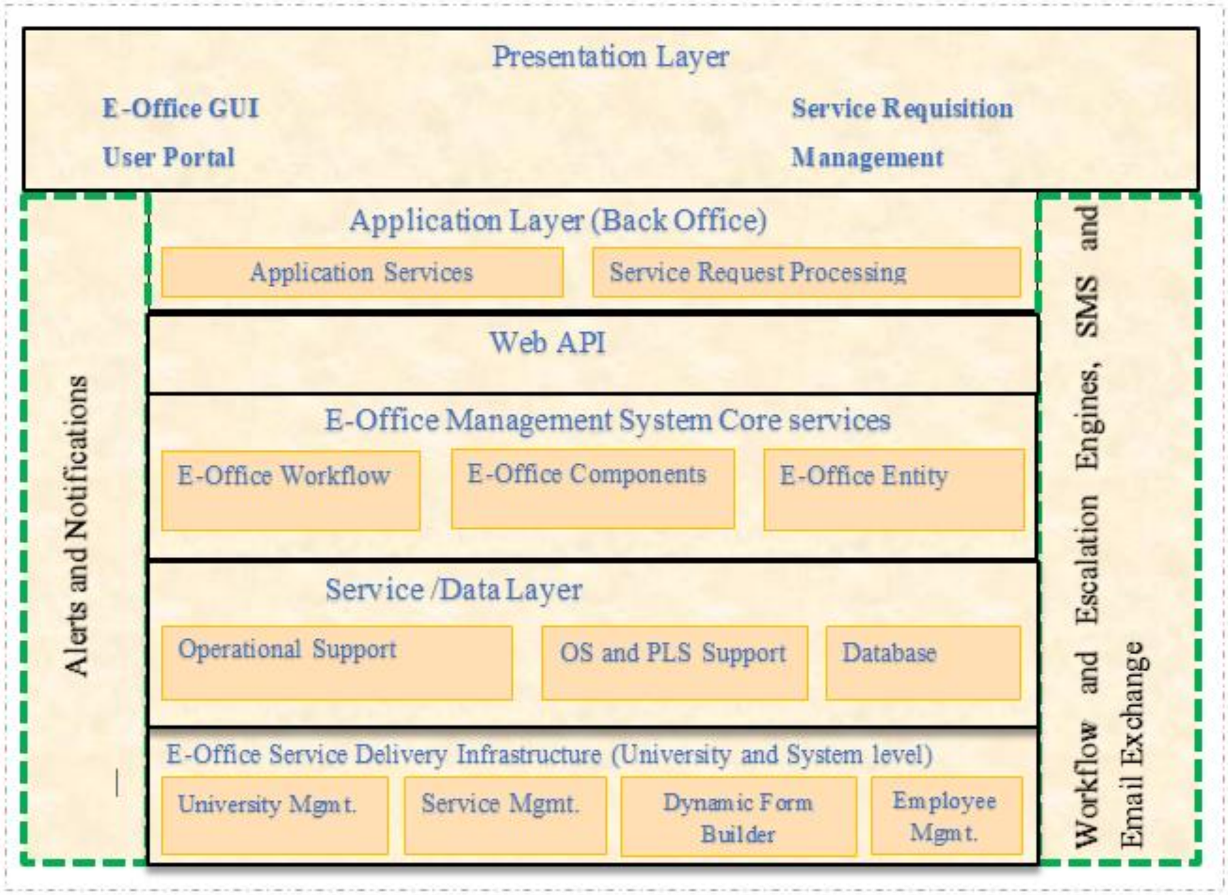


Figure 5. 3 : Conceptual E-Office Architecture for ASTU

As it is shown in Figure 5.3, the architecture provides detail information about the electronic office components. Moreover, the architecture is the integration of e-Office components such as workflow and escalation engines, SMS and email exchanges, service delivery infrastructure, data layer components, core services that interact with UI components using WebAPI. The figure focuses on the components of the system without integration with other external systems.

5.4.1. Users and End User Devices Layer

The device and user layer include

Users: - employees / users of the Adama Science and Technology University with access rights and use of the system.

End-user devices: - End-users communicate with the cloud-based E-office system using browser-enabled devices such as handheld computers, laptops, desktops, tablets, and mobile phones using protocols such as SSH, http / https.

5.4.2. Presentation Layer

The presentation layer consists of components that implement and display the user interface and manage user interaction with the system. For E-office, the application graphical user interface, E-mail client, and web browser (including mobile devices) are used for user interaction with the system at the presentation level. User interface components provide a way for users to interact with the electronic office application. They render and format data for users, and they also collect and validate data entered by the user. In the presentation layer of the proposed system like: -

***User Portal**: -* Provides the functions and features to authenticate and identify users and to provide a convenient, intuitive, personalized, and customizable web interface for facilitating access to information and services that are of primary importance and interest to users. In the E-office framework, it is an optional platform on which e-office components can be delivered.

Service Requisition Management**:** - The portal side of the institution, an employer can request and process their request in the required service management system on the back-office page. The university employers are requesting various services of the university from the e-office system and are submitting their request.

5.4.3. Application Services Layer

The main application services of e-office components from data analysis of the chapter 3 include:

Document Management Systems - is the service delivery, it is the capture and management of documents within the institution and is designed to provide different offices with the administration, creation, and processing to support documents by providing a centralized repository and workflow process that encapsulates business rules and metadata. The focus of a DMS is primarily on the storage and retrieval of stand-alone electronic resources in their native format.

Collaboration and messaging**:** - As shown in Appendix A and B the interview and questionnaire respectively, most employers use paper-based communication in a dominant form, then it is inefficient, and there is a lack of transparency, an increasing dependence on real-time collaboration tools. Therefore, we suggest solving the problem by implementing and using SMS, E-mail, a shared calendar, joint events (conferences, workshops, etc.), discussion forums, and instant messaging.

Dashboard*:* - Dashboard in the E-office system provides a proactive management information system in the form of visual alerts provides information on the schedule of critical activities and covers various file statistics, correspondences and monitoring of other important information.

Management of reports: - The reports present the analysis of the day-to-day activities of the institution, such as archives, correspondences in various dimensions such as the period of time, the Senior Administrative Office, Department / Unit, Faculty / School, and person.

Processing of forms: As shown in Appendix A and B the interview and questionnaire respectively in the institution, they have different forms and the automation solution focuses specifically on automating or preparing a dynamic form generator for the most repetitive paper-based forms of the institution. ***Knowledge base management system***: - Electronic knowledgebase management system that allows the university to store, manage and deploy all kinds of documents, and ***Business process management system***: - BPMS automate core business processes from "end to end" of the university.

Maintenance**: -** In the application service layer of the cloud, maintenance of the proposed system components should be required. SaaS maintenance is software that is remotely hosted and delivered over the Internet. Instead of buying the software license directly, you rent the software monthly or annually.

Service Request Processing (Fulfillment)*:* - On the side of the portal, employers can request different services and send them to the appropriate office, and then the administrative office of the employers/users of the application layer or the assigned workers can process the application/request and respond for the applicant.

5.4.4. Web API

The web API, as its name suggests, is an API on the web that accessed through the HTTP protocol. It is a concept and not a technology. We can build web API using different technologies like .NET, etc. It is an interface to communicate the user's HTTP request to the central service layer or the back-end implementation and the return HTTP response. So, we have chosen to use the web API because for the development of the prototype we have .Net Framework 6.0, it is compatible with the HTTP protocol, and we are familiar with ASP.NET, and MVC architecture.

In general, the web API used to communicate the HTTP request, the HTTP response data of the web application and the mobile application with the main services of the proposed system.

5.4.5. Electronic Office Management System Core Services Layer

The main application in the core services layer from data analysis of the chapter 3 includes **-**

E-Office workflow: - workflow components used when an electronic office needs to complete a series of information processing tasks that depend on the information content. When designing workflow components, we must consider the options that are available to manage the workflow. Many electronic office processes involve several steps that must be performed in the correct order.

E-Office subcomponents in core services**: -** It is the main layer of the platform for electronic office, which includes all the engines required by the application services.

CMS Engine: allows you to capture, create a process, publish, and edit the contents of the university, as well as maintenance from a central location. ***Report engine***: as we discussed in the application layer and used to extract data from various sources and display it in reports with a user-defined format. ***Rules engine***: to send / receive or automatically dispatch different documents and application forms of the university, first it offers the possibility to register, define, classify and manage all the rules, as well as verify the consistency of the definitions of rules. Then based on the rule it dispatches the requests. ***Workflow engine***: defines a process, the rules that govern the process decisions and routes the information. ***Search engine***: used to locate information or data from structured or unstructured databases of E-Office and other systems. ***Messaging engine***: used to enable E-mail, SMS, instant messages, discussion forums and other forms of messages and collaborations. ***Forms engine***: it is a framework for the dynamic creation of forms and the processing of complex forms of institutional forms. ***Metadata engine***: summarizes the basic information about the data.

E-Office Entity**: -** E-office units are used to transfer data between the E-office components. The data represents entities of the real electronic office, such as the form generator, the University structuring, knowledgebase, Employee, Service, and the messaging service. The internal use of the electronic office application usually takes place. Alternatively, they can be implemented with custom object-oriented classes that represent the real entities in which the application should work.

5.4.6. Service /Data Layer

The service layer centralizes the general data access functionality, making the application easier to configure and maintain. The e-office data repository, which includes databases, web content, and data from a business and external applications, and provides core services such as development environment such as operating system and programming language, operational support, management system databases, middleware, and other platforms.

5.4.7. E-Office Service Delivery Infrastructure

It provides the basic infrastructure for the delivery of services or communication between the service requester and the fulfillment of the service. These services must be prepared of the modules at the organization/university and system level, such as organization structuring (ASTU), services, and employees, dispatching of different processes, managing forum-discussions, managing the knowledgebase, and generating dynamic forms to prepare the institutional forms.

5.4.8. Cloud Service Delivery Components

The cloud service delivery components for the e-Office include the followings:

***Software as a service*: -**SaaS layer is the second layer of the architecture below the presentation layer or user interface layer, which allows access to the hosted electronic office system and any third-party application software to run on the cloud platform. Services provided in the PaaS layer can be utilized by sending request messages through Web APIs. Therefore, the institution employers and users can access electronic office system and other third-party software through a web interface.

***Platform as a service*: -** PaaS layer is the third layer of the architecture, which is deployed on a cloud provider that can support a requirement of the designed platform. It consists two sub-layers called electronic office core service layer and service/data layer.

***Infrastructure as a service*: -** IaaS layer is the fourth layer of the architecture in which the cloud-based electronic office system is implemented. It contains the physical and virtualized infrastructure including the institution ICT Infrastructures.

5.4.9. Tools

The proposed framework from data analysis of the chapter 3 includes -

Communication tools: As shown in Appendix A questionnaire, there is an ICT support tool to complain from other offices to ICT technicians but need additional tools to communicate from one office to another. The tools such as instant messaging, discussions in electronic forums, SMS and E-mail exchange.

Signature tool: - The signature tool uses an electronic or digital signature tool by integrating the software with the hardware part (signal by mouse or pen) for electronic documents.

5.4.10. Integration

The E-office framework provides a seamless integration between the subcomponents and the systems currently used by the institution and other systems at the national level. The information value chain was designed to create a seamless workflow between modules, reduced errors, and omissions. The integration interface should ensure that the E-office system follows the Ethiopian Government Interoperability Framework (eGIF).

5.4.11. Security Management

Security is the ability of a system to prevent accidental actions outside its intended destination, which remains the big problem. Rightly, it allows access to resources and services based on assigned responsibilities. You should implement security and control at the process and data levels.

Cloud computing has many security mechanisms to protect the data and the entire system holding and processing it. Outside of these mechanisms, the following are very common and are used in almost all cloud service providers. Also, for your own security methods [49]. The mechanisms are Firewall in hardware and software, Encryption, Hashing, Digital Signature, Public Key Infrastructure (PKI), Identity and Access Management (IAM), and Single Sign-On (SSO). In this study the IAM mechanism and Firewall was used.

IAM controls and tracks user identities and access privileges for electronic office, the knowledge-sharing environment. It includes components and policies such as authentication, authorization, user management, and credential management that are already provisioned for

IaaS environment. In addition, remote management, resource management, SLA (Service Level Agreement) management and billing management are used as a management mechanism in cloud computing.

These mechanisms typically provide unified APIs and can be offered as single results, custom applications, merged into different output sets or multi-function applications [50].

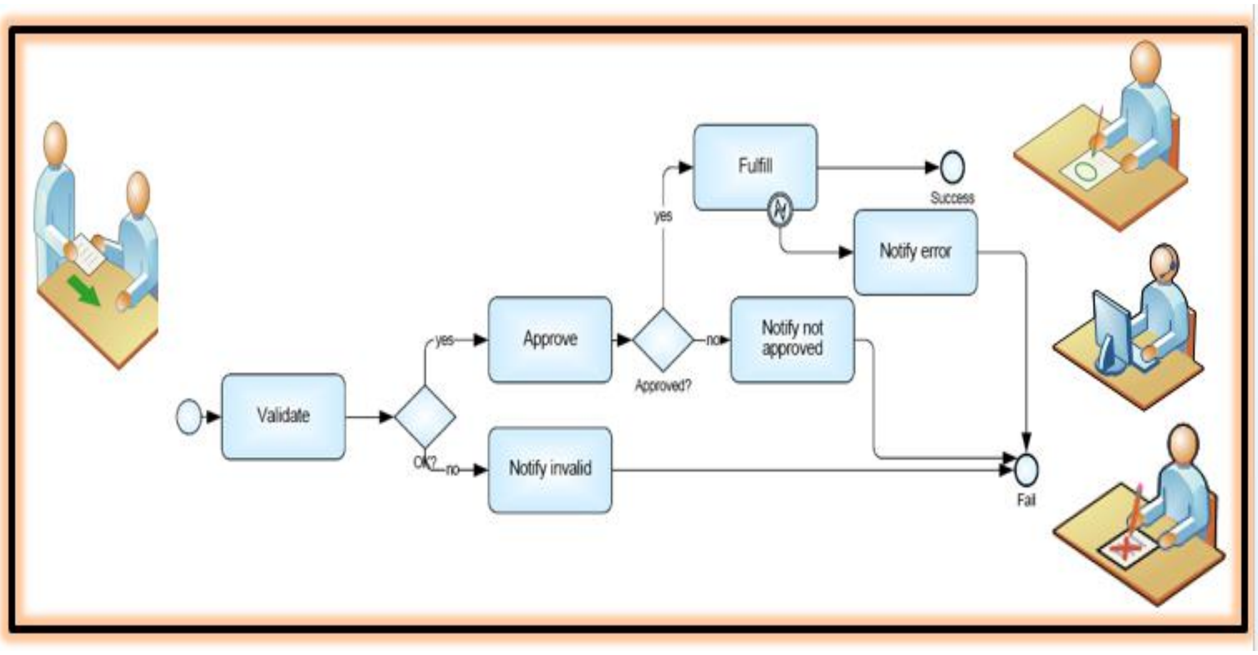


Figure 5. 4 : Conceptual Service Request Authentication Process of CBEOF for ASTU

Figure 5.4 is own work, shows that there is service request then it validates the system requester whether it is authorized or not and if it is not valid the system notifies invalid message and if the requester is valid it approves and continues to the next path. After the requester approves it fulfills the given service and the system produces success message. Otherwise, it notifies invalid message.

5.4.12. Conceptual System Architecture of CBEOF

The system architecture shows that the relationship among service requisition with request processing via internet. Before that the service delivery infrastructure must be prepared in order to perform the service requisition and service processing. The data analysis in chapter 3 used as input for E-office components in the figure 5.5 below.

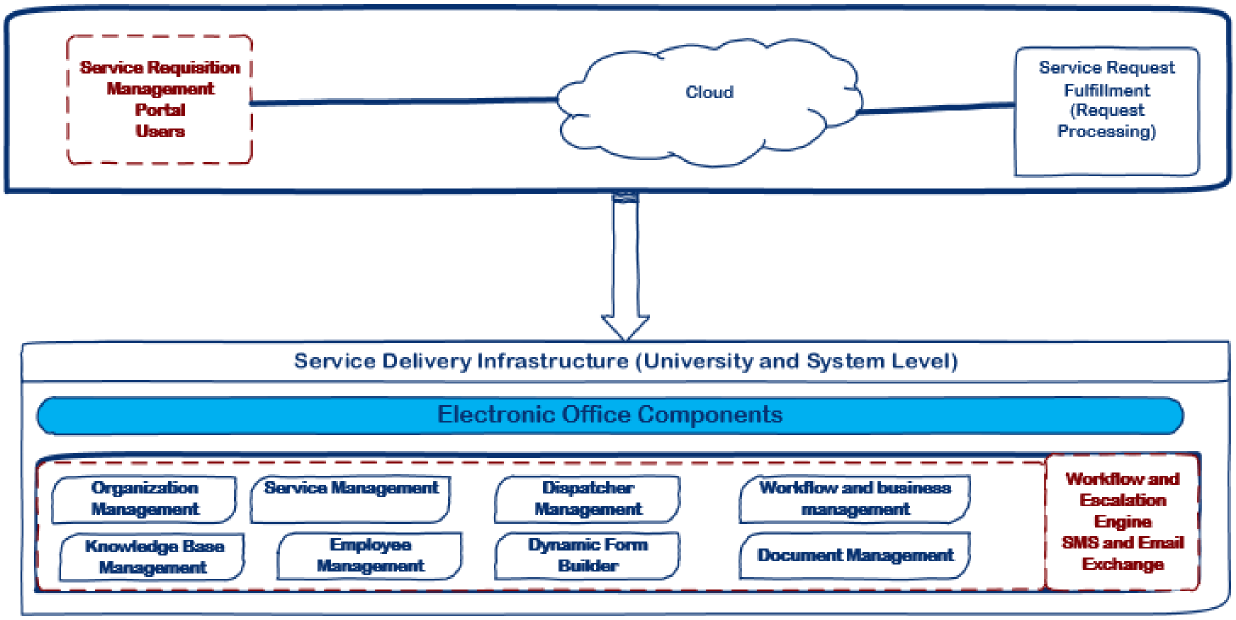

Service Requisition Management Portal Users
Cloud
Service Request Fulfillment (Request Processing)
Service Delivery Infrastructure (University and System Level)
Electronic Office Components
Organization Management
Service Management
Dispatcher Management
Workflow and business management
Knowledge Base Management
Employee Management
Dynamic Form Builder
Document Management
Workflow and Escalation Engine SMS and Email Exchange

Figure 5. 5 : Conceptual System Architecture of CBEOF for ASTU

Figure 5.5 is own work, showing the prototype implementation for service delivery infrastructures originate from the data analysis or use as data input such as Organization Structuring (ASTU), Employee/customer, service, design dynamic form generator. Preparing / designing the application form for the selected service for the employee / user based on the privilege. After filling out the information and sending it to the responsible department.

Maybe the verification of the request approved / rejected by the privileged user performed. The status can be transmitted using automatic alerts and notifications. If the service request is delayed or not implemented on the baseline, it automatically switches to senior management. Moreover, designs business processes and workflows for every service.

Chapter Six: Development of the prototype

6.1. Introduction

In this chapter, we provide an overview of the prototypical implementation of the proposed architecture, which was developed in chapter five. To this end, we focus on the implementation of some parts of the electronic office management system. This is the development of the front-end web application interface, an intelligent way to create a digital workplace; design services form dynamically, and manage services, Departments / Units, and Teams / Divisions. And web-based interfaces from Microsoft Azure providers of the system, the functionality of the link between each menu, the portability or the mobile device and tablet interface to search, and the desktop or laptop design of the developed prototype and other key functions to demonstrate the concept.

In this work, we apply the MVC mode, Angular JS, Bootstrap, and other libraries used for the development of the user interface and ASP.Net for back-end implementation. Windows Azure cloud platform selected as application platform for implementation and creation, SQL server as a data storage in local and use SQL Azure database in Microsoft Azure.

6.2. Development Environment and Tools

Figure 6.1 in below, Illustrates Overview development view of the proposed framework. In the CBEOF, development environment for the cloud-based electronic office web application is configured; the university employers and users can access the system using different windows operating systems and web browsers. The UI of a cloud-based e-office system was created using Angular JS, HTML, and CSS for front or UI development and the editor used Jet Brains Web Storm 2016.3.1. Visual Studio IDE 2015 (ASP.net and MVC) architecture for backend or core service implementation and C # as the main language.

The integration of this CBEOF system with other systems of the university and national systems related to the proposed framework can be operated using JSON or other programming technology. For the purpose of managing and storing electronic office processes, transactions, and operations, we have used SQL Management Studio 2014, Azure storage service, and IIS8 is local server to deploy the application and it is other alternative of cloud deployment. The development platform is Window Azure platform (Platform as a Service).

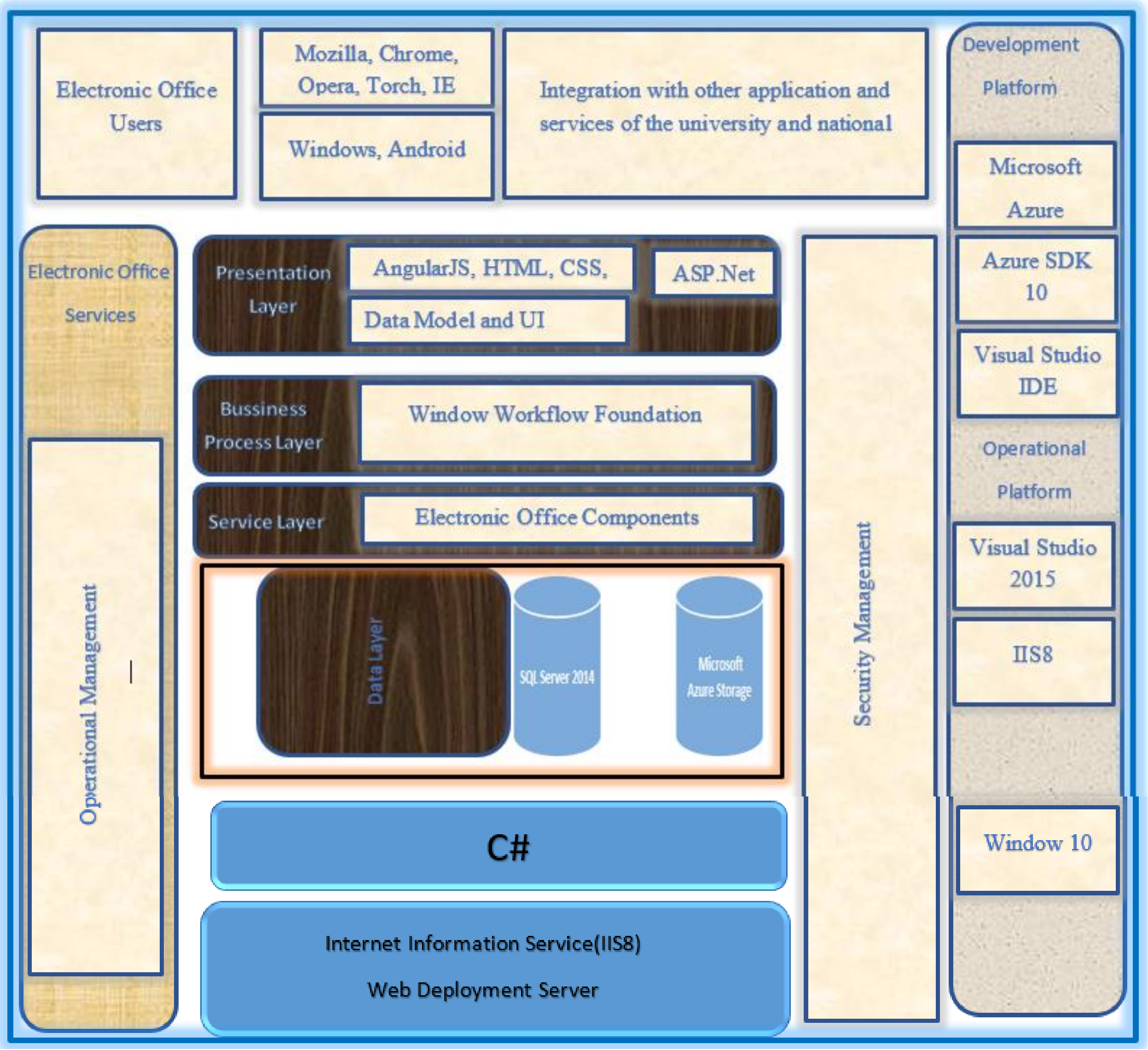


Figure 6. 1 : Overview of developmental view of CBEOF (Source: adapted from [45])

6.3. The Proposed Framework Developed System Function

Based on the design architecture defined in the previous chapter, a demonstration web application was developed in the electronic office management system to show how the framework and architecture can be used in the actual application.

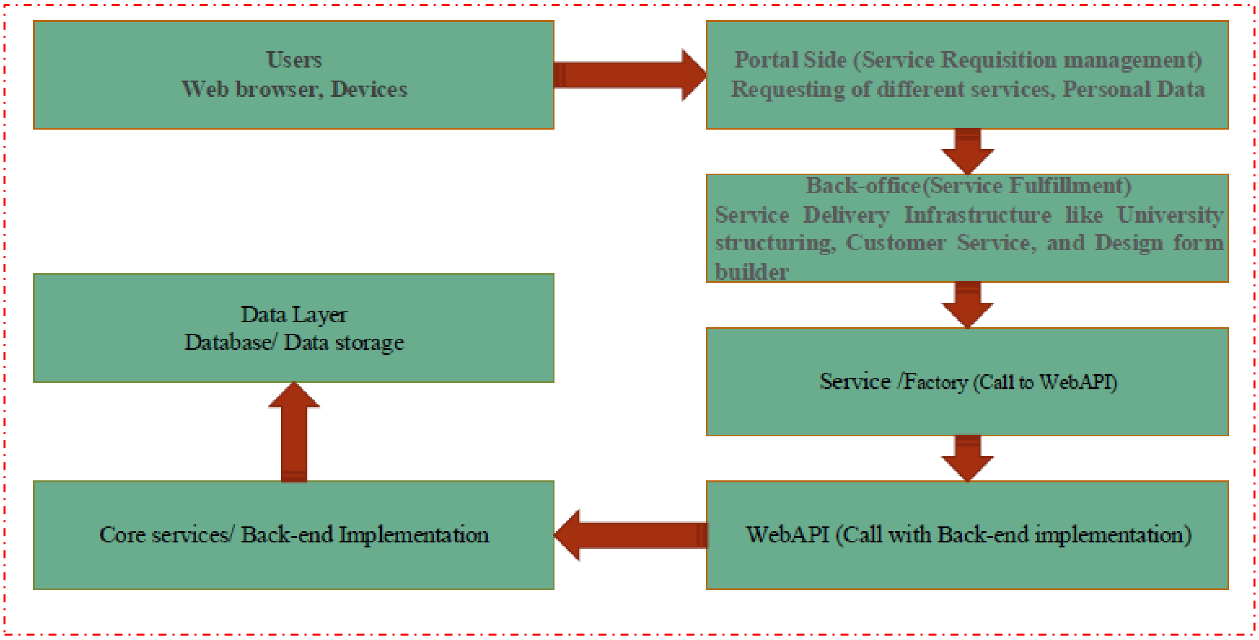

Users
Web browser, Devices
Portal Side (Service Requisition management)
Requesting of different services, Personal Data
Back-office(Service Fulfillment)
Service Delivery Infrastructure like University structuring, Customer Service, and Design form builder
Data Layer
Database/ Data storage
Service /Factory (Call to WebAPI)
Core services/ Back-end Implementation
WebAPI (Call with Back-end implementation)

Figure 6. 2 : The Overall process of user request

Figure 6.2 is own work, shows the prototypical implementation of the proposed system: -

Firstly, prepares the infrastructure for providing services such as organization structuring (ASTU), customer services, and customers.

Second, design the dynamic creation of services form by setting the number of attributes used and then designing the dynamic form generator.

Thirdly, on the portal page, you will receive the form for the specified service, fill in all the required information, and send the request. In addition, there are other functions such as registering employees, logging in, and managing personal information like to know the personal requests, active, closed, and replay messages for their request.

Fourth, the requesting services are processed in the back-office process by assigning the employee / user of the university, such as approval, replay messages for customer requests, and forwarding, to another office.

Implement the prototype architecture and develop implementation details, interaction-oriented architectures as an architectural description concept. Every service can be identified by using code, and create based on service category to filter efficiently, effectively, and manage easily.

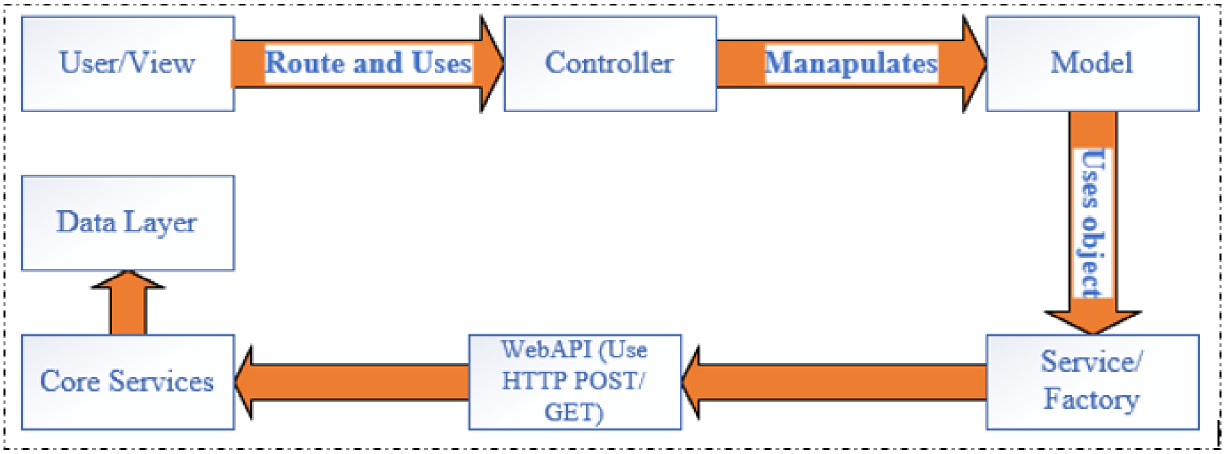


Figure 6. 3 : MVC architecture (adapted from [51])

Interaction-oriented software architecture consists of separating user interaction from data abstraction and processing business data, and splits the system into three main partitions. [51]:

- Data module / models: provides data abstraction, all business logic, or the core services of electronic office and interacts with the data layer.
- Controller module: respond to user action and direct the flow of the electronic office.
- View module: it is responsible for the visual presentation of the data output, it also provides an interface for the user input. Format and present the data from the model to a user.

In general, Figure 6.3 above shows that the MVC architecture to the view of user interaction up to data layers from each user interface. When the user clicks an action, by creating AngularJS module, then forwards/routing path, and create global variable to store service data, call the config base paths, the controller creates and uses objects, communicate with service / factory then connect to HTTP interceptor in order to communicate with WebAPI through the URL, core services, and data layer. All communication is done in two ways. Then data is send request and return responds.

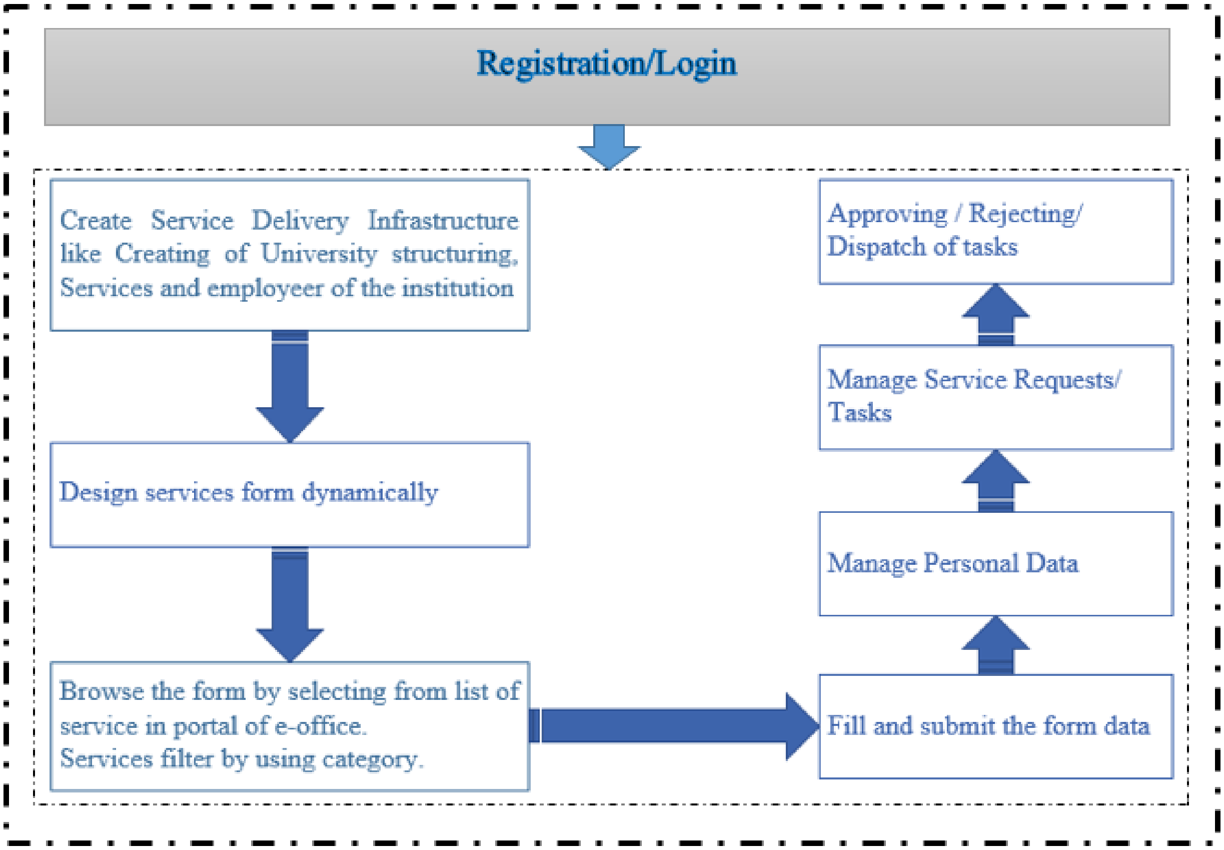

Registration/Login
Create Service Delivery Infrastructure like Creating of University structuring, Services and employeer of the institution
Design services form dynamically
Browse the form by selecting from list of service in portal of e-office.
Services filter by using category.
Fill and submit the form data
Manage Personal Data
Manage Service Requests/ Tasks
Approving / Rejecting/ Dispatch of tasks

Figure 6. 4: Overall Prototype Implementation Process of User Interface

As we have seen from figure 6.4, the overall prototype implementation process: first by preparing the service delivery infrastructure like university structuring, services and employer of the university, then by selecting one service from the list of services then choose whether service request or suggestion to design form dynamically.

After design the form then go to E-Office portal of ASTU and filter the services based on category and select one service the render the form, fill and submit the data. In addition, in portal side there are other functionalities like manage personal data- Track my file, knowing my active tasks, closed tasks, messages and manage by account. For the service request in portal side possible to manage in back office of the requests by approving/rejecting and forwarded to other offices. In general, the prototype implementation has registration and login functionalities.

6.4. Prototype DataBase Design

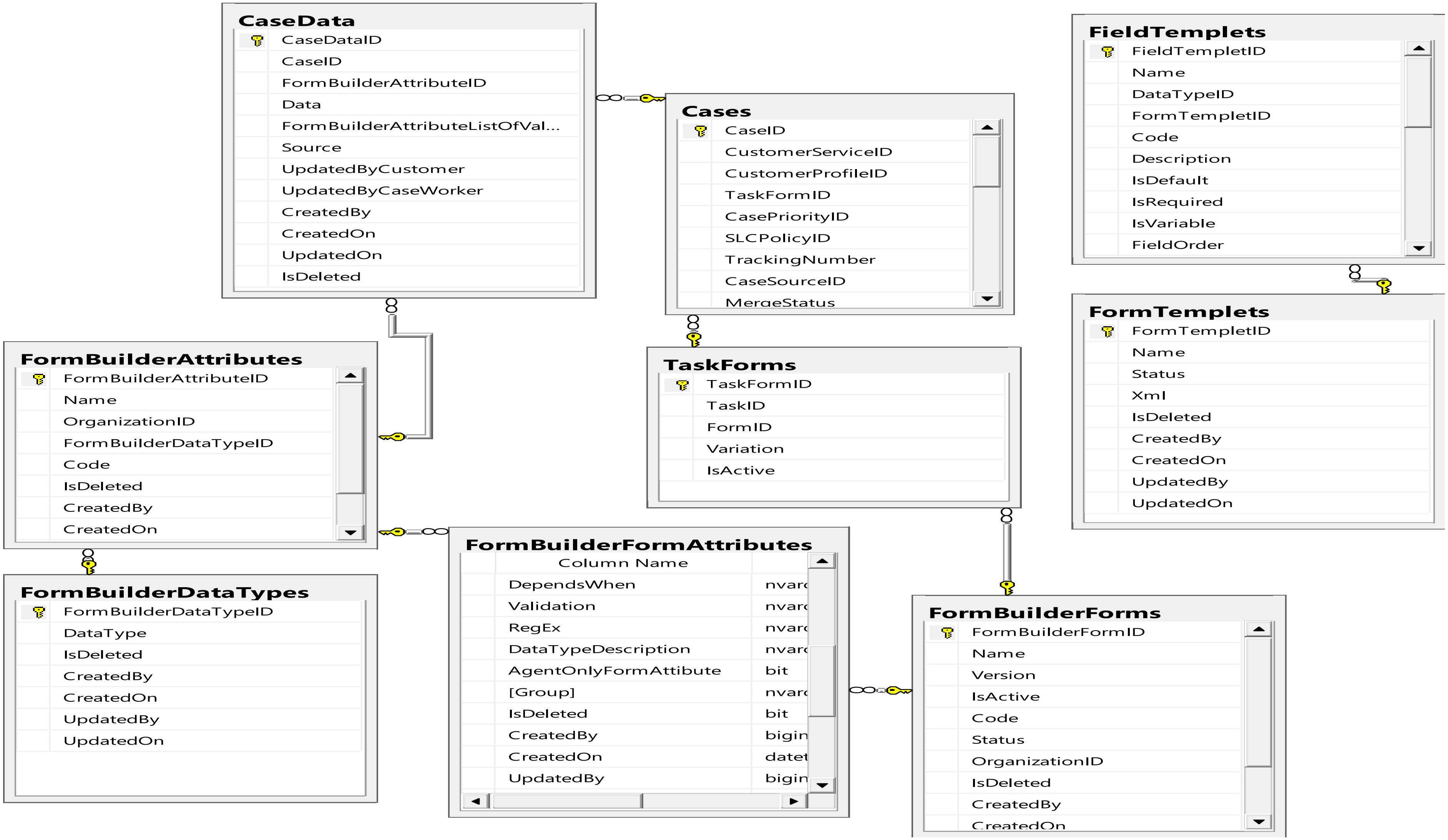
As shown in Figure 6. 5: Sample Database Design below, we tried to show the prototype of the database design for the dynamic form builder and service request data, but not the other table include because the figure is not visible for reviewers. To define the following sample table list to design the dynamic form builder and cases/service request:

Form Builder Data Type: - to store a list of data types such as string, number, date, text, rich text, single-value list, multivalve list, tree, dynamic list, file, and UI element. **Form template**: - to save the only default form with subject and content fields when creating the service. Therefore, each service has a default form. **Field attribute**: - to save the standard fields / attributes and to have a relationship with the form template and the data type. **Form Builder Attributes**: - Saves the organization attribute (ASTU) and has a relationship with the form builder's data type table. Each attribute identifies itself by assigning its own code and name.

Form Builder Form Attributes: - Stores the list of attributes and has a relationship with the form builder data type and form creation tables. **Form Builder Form**: - saves the list of form names for the selected service. This can be done after we design the form and submit the attribute list. **Form Builder Form Attributes**: - Stores the list of forms with the specified attribute and identifies the attributes based on the code.

Service request/Cases: - To identify whom the requestor is, for which service and to assign the tracking number for each case/customer request. Therefore, the case indicates that the request comes from customer / employee. **Service request/Case data**: - After identification for each case / request and save the data for a specific attribute. There are many tables that are not included in the following database design: Employee, Organization (ASTU), Department / Unit, Team / Department, Service, and assignment.

In general, to explain the relationship of the tables, first created the organization structuring (ASTU), created under the university, services also created as the university and assign to the corresponding department. Then create form for selected service. To create the form, first prepare the data types, the form template to the default name of the form with other related information, the field attribute to the default attribute, and the form builder validation to validate the fields.



CaseData

CaseDataID
CaseID
FormBuilderAttributeID
Data
FormBuilderAttributeListOfVal...
Source
UpdatedByCustomer
UpdatedByCaseWorker
CreatedBy
CreatedOn
UpdatedOn
IsDeleted

Cases

CaseID
CustomerServiceID
CustomerProfileID
TaskFormID
CasePriorityID
SLCPolicyID
TrackingNumber
CaseSourceID
MergeStatus

FieldTemplets

FieldTempletID
Name
DataTypeID
FormTempletID
Code
Description
IsDefault
IsRequired
IsVariable
FieldOrder

FormTemplets

FormTempletID
Name
Status
Xml
IsDeleted
CreatedBy
CreatedOn
UpdatedBy
UpdatedOn

FormBuilderAttributes

FormBuilderAttributeID
Name
OrganizationID
FormBuilderDataTypeID
Code
IsDeleted
CreatedBy
CreatedOn

TaskForms

TaskFormID
TaskID
FormID
Variation
IsActive

FormBuilderDataTypes

FormBuilderDataTypeID
DataType
IsDeleted
CreatedBy
CreatedOn
UpdatedBy
UpdatedOn

FormBuilderFormAttributes

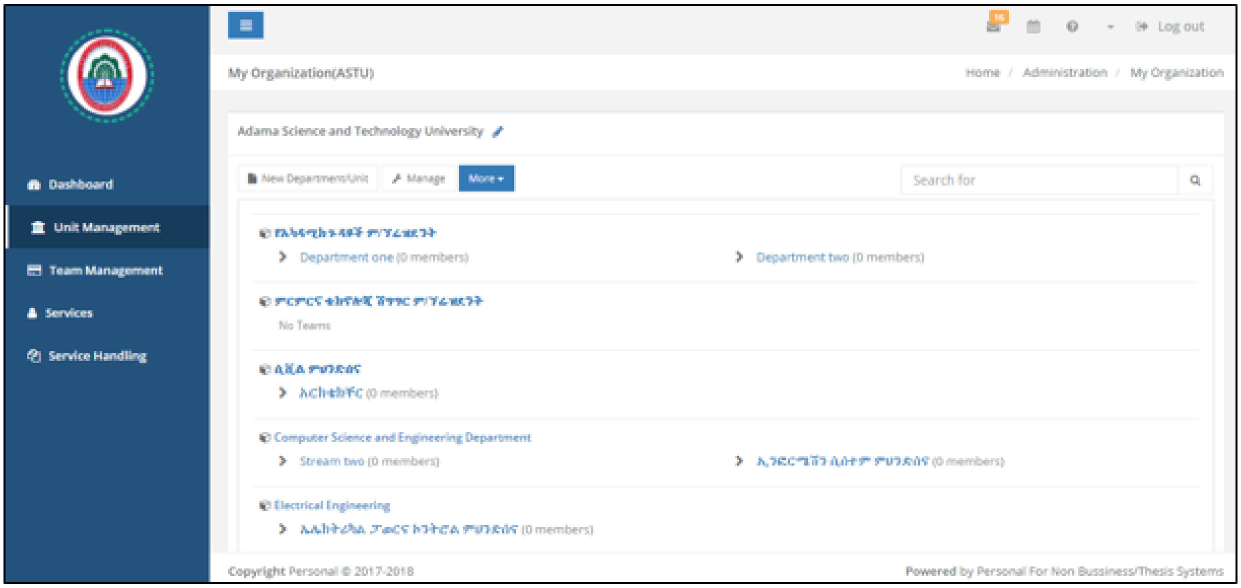
Column Name	
DependsWhen	nvarc
Validation	nvarc
RegEx	nvarc
DataTypeDescription	nvarc
AgentOnlyFormAttibute	bit
[Group]	nvarc
IsDeleted	bit
CreatedBy	bigin
CreatedOn	datet
UpdatedBy	bigin

FormBuilderForms

FormBuilderFormID
Name
Version
IsActive
Code
Status
OrganizationID
IsDeleted
CreatedBy
CreatedOn

Figure 6. 5: Sample Database Design

6.5. User interface with description



My Organization(ASTU)

Home / Administration / My Organization

Adama Science and Technology University

የአካዳሚክ ጉዳዮች ም/ፕሬዚዳንት

- Department one (0 members)
- Department two (0 members)

ምርምርና ቴክኖሎጂ ሽግግር ም/ፕሬዚዳንት

No Teams

ሲቪል ምህንድስና

- አርክቴክቸር (0 members)

Computer Science and Engineering Department

- Stream two (0 members)
- ኢንፎርሜሽን ሲስተም ምህንድስና (0 members)

Electrical Engineering

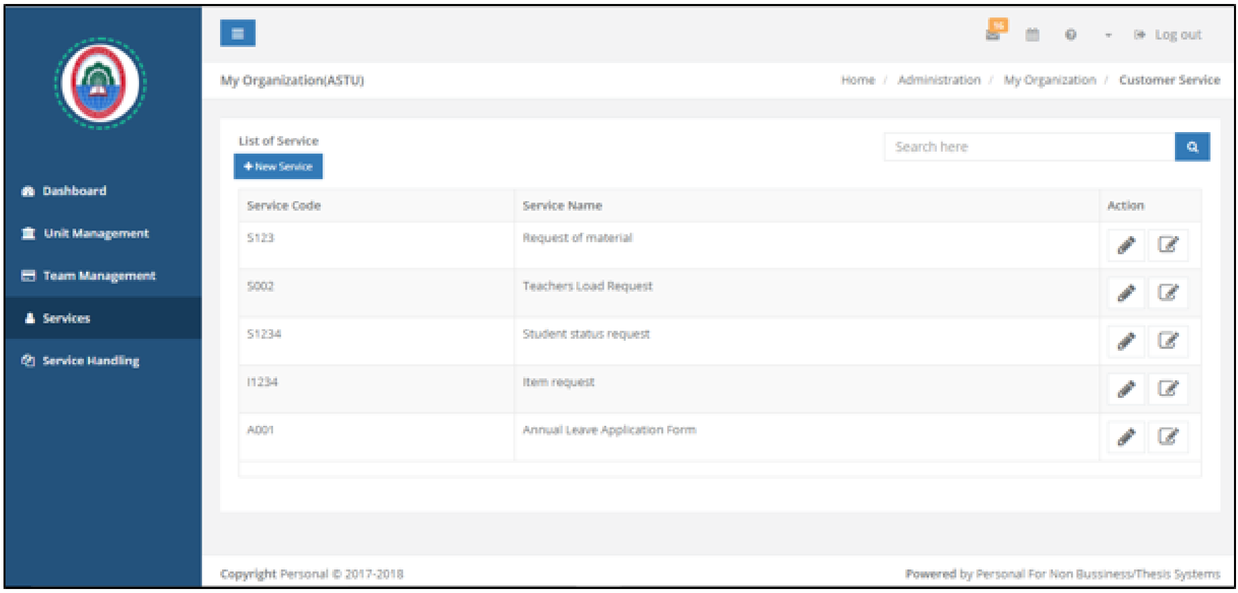
- ኤሌክትሪካል ፓወርና ኮንትሮል ምህንድስና (0 members)

Copyright Personal © 2017-2018

Powered by Personal For Non Bussiness/Thesis Systems

Figure 6. 6 : Manage the organization structuring (ASTU)

As shown from figure 6.6, indicates that to manage the organization structuring (ASTU) like creating of the Unit/Department and Teams/Divisions.



My Organization(ASTU)

Home / Administration / My Organization / Customer Service

List of Service

Service Code	Service Name	Action
S123	Request of material	
S002	Teachers Load Request	
S1234	Student status request	
I1234	Item request	
A001	Annual Leave Application Form	

Copyright Personal © 2017-2018

Powered by Personal For Non Bussiness/Thesis Systems

Figure 6. 7 : Service list of the organization (ASTU)

As shown from figure 6.7, after preparing the organization structuring (ASTU), then creating of services of the organization (ASTU) is next step by design and update the default service form.

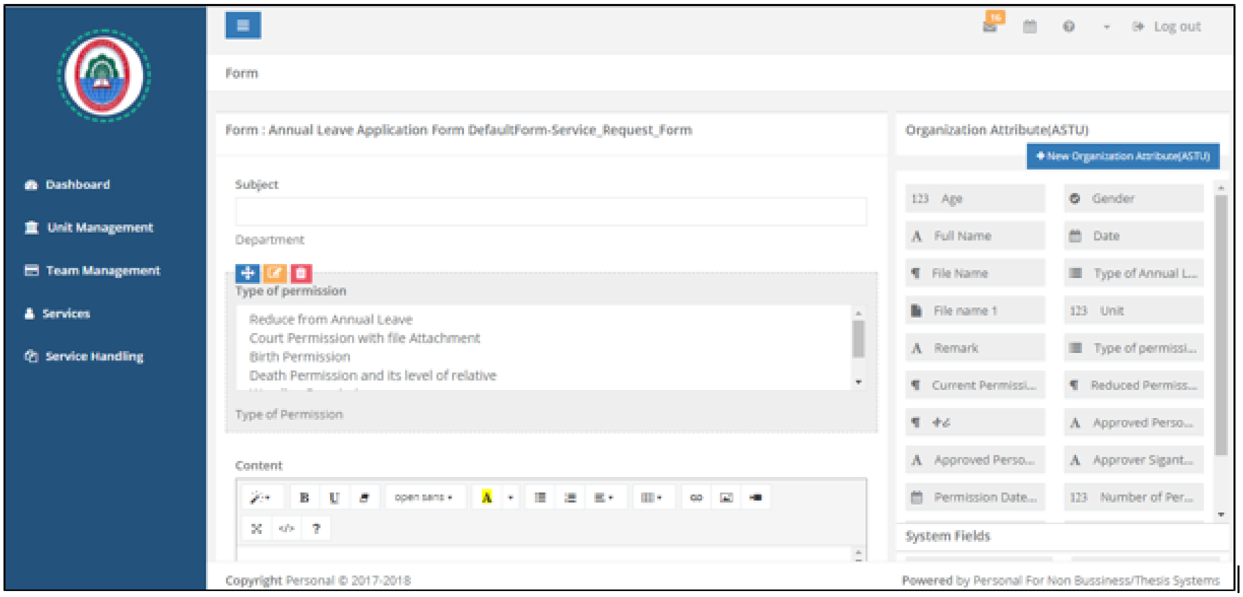


Figure 6. 8 : Dynamic Form Builder/Generator

As we have shown in Figure 6.8, dynamic form builder starting by creating the organization attribute, designing the services form, and then save the form for the selected service.

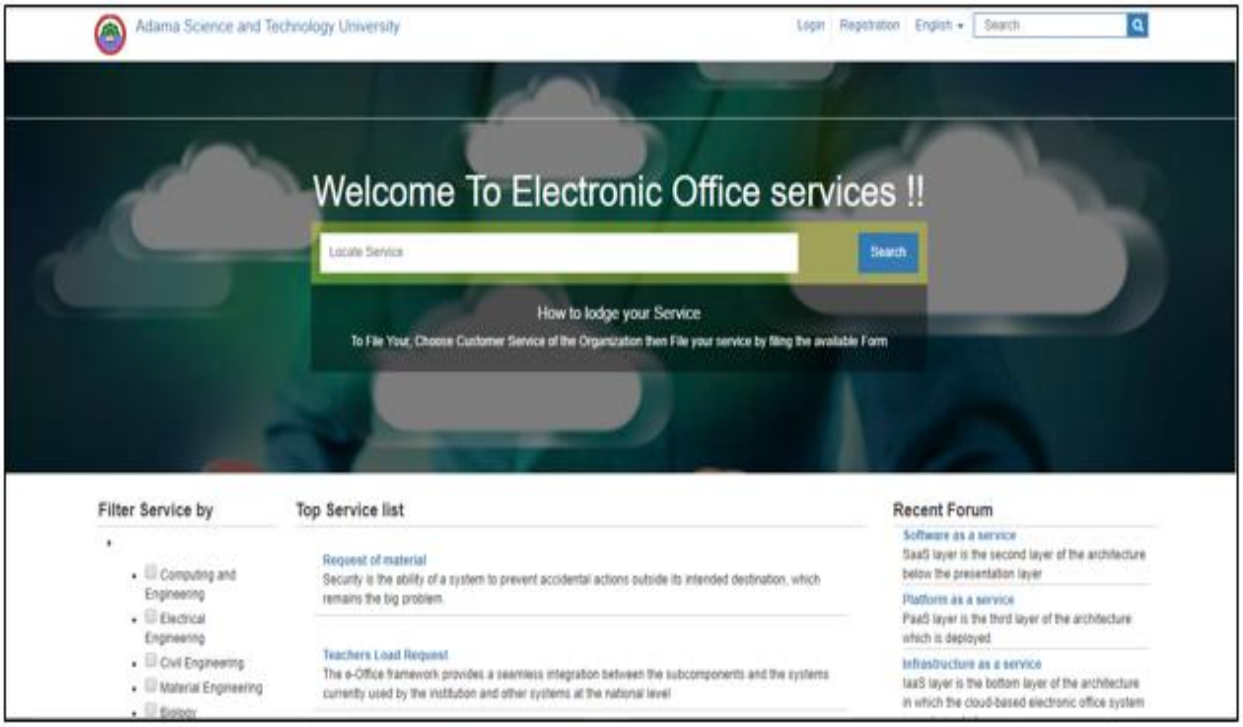
The following code shows the view, the controller, and the service / factory code for designing the dynamic form generator.

```
<div class="col-xs-12 col-lg-4 g-lg-lst-2">
  <div class="ibox">
    <div class="ibox-title">
      <h5 translate> Organization Attribute(ASTU)</h5>
      <button ng-click="formCtrl.createAttribute()" class="btn btn-primary btn-sm pull-right">
        <i class="fa fa-plus"></i>
        <span translate>New Organization Attribute</span>
      </button>
    </div>
    <div class="ibox-content col-xs-12">
      <div class="row field-cnt" ng-if="formCtrl.organizationAttributes.length">
        <div class="col-xs-6 field-btn-cnt" ng-repeat="field in formCtrl.organizationAttributes">
          <button class="field-btn " ng-click="formCtrl.addFormAttribute(field)">
            <attribute-icon type="{{field.DataType}}"></attribute-icon>
            <span
```

```
class="name">{{field.Name}}</span>
     </button>
    </div>
   </div>
   <div class="row field-cnt" ng-
if="formCtrl.uiAttributes.length">
    <div class="col-xs-6" ng-
repeat="fields in formCtrl.uiAttributes">
     <button class="field-btn " ng-
click="formCtrl.addFormAttribute(fields
)">
      <attribute-icon
type="{{fields.DataType}}"></attribute-
icon>
      <span
class="name">{{fields.Name}}</span></
button>
    </div>
   </div>
   <div class="row b-v p-h-10" ng-
if="formCtrl.genericAttributes.length">
    <h4 translate>System Fields</h4>
   </div>
   <div class="row field-cnt" ng-
if="formCtrl.genericAttributes.length">
    <div class="col-xs-6" ng-
repeat="field in
formCtrl.genericAttributes">
     <button class="field-btn" ng-
click="formCtrl.addFormAttribute(field)
">
      <attribute-icon
type="{{field.DataType}}"></attribute-
icon>
      <span
class="name">{{field.Name}}</span>
     </button>
    </div>
   </div>
  </div>
 </div>
</div>
vm.createAttribute = createAttribute;
function createAttribute() {
 modalAttribute();
}
function modalAttribute(selectedAttribute)
{
 var oldSelectedAttribute =
selectedAttribute;
 var modalInstance = $uibModal.open({
  size: 'sm',
  templateUrl: 'app/form-
builder/views/modal-attribute.html',
  controller:
'ModalAttributeFormBuilderController',
  controllerAs: 'attribute',
  resolve: {
   selectedAttribute: function () {
    return selectedAttribute;
   }
  });
 modalInstance.result.then(
   function (selectedAttribute) {
     // Find item index using indexOf+find
     var index =
_.indexOf(vm.organizationAttributes,
_.find(vm.organizationAttributes,
oldSelectedAttribute));
     if (index) {
vm.organizationAttributes.splice(index, 1,
selectedAttribute);
     }
       else {

vm.organizationAttributes.push(selectedAt
tribute);
       }
     },
}
createAttributes:createAttributes,
function createAttributes(data) {
  return base('createAttributes', data);
}
```

CreateAttributes: __env.apiUrl + "FormBuilder/CreateAttibute" - In general, createAttributes is the service name, calls in controller and 'createAttributes' config name and link to API method.



How to lodge your Service

To File Your, Choose Customer Service of the Organization then File your service by filing the available Form

- □ Computing and Engineering
- □ Electrical Engineering
- □ Civil Engineering
- □ Material Engineering
- □ Biology

Request of material
Security is the ability of a system to prevent accidental actions outside its intended destination, which remains the big problem.

Teachers Load Request
The e-Office framework provides a seamless integration between the subcomponents and the systems currently used by the institution and other systems at the national level

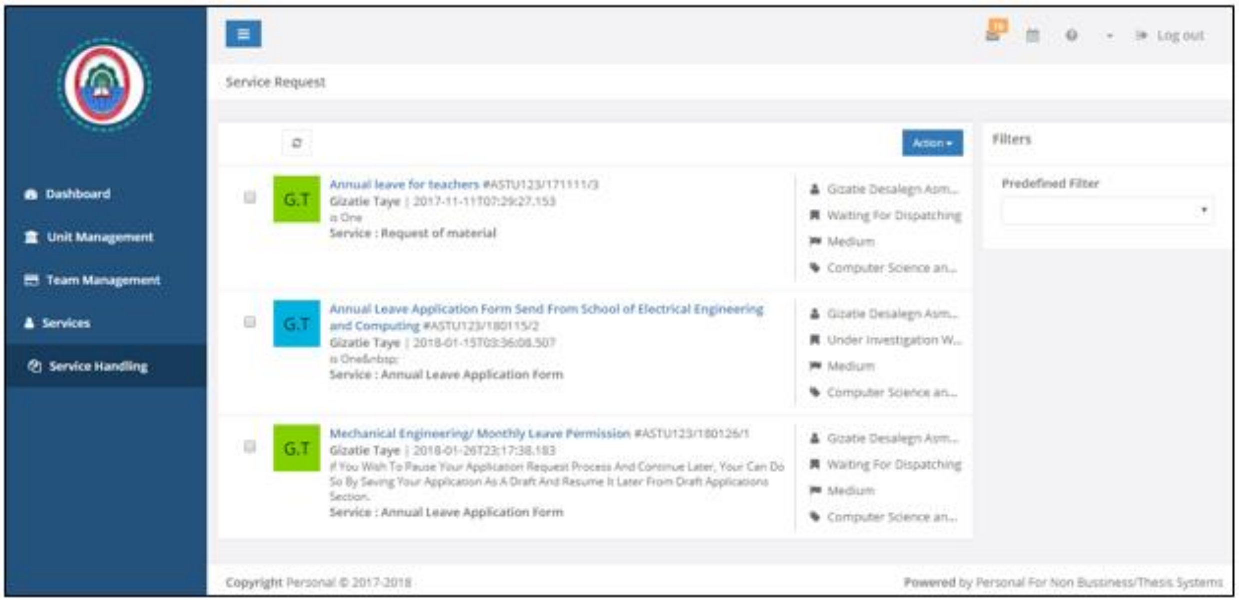
Software as a service
SaaS layer is the second layer of the architecture below the presentation layer

Platform as a service
PaaS layer is the third layer of the architecture which is deployed

Infrastructure as a service
IaaS layer is the bottom layer of the architecture in which the cloud-based electronic office system

Figure 6. 9 : Portal of Electronic Office for ASTU

Here is the figure 6.9, which shows the portal of the electronic office for ASTU which manages the various services, personal data, login, and registration management. After you have logged in, select the service on the above page and present the prepared form. Then fill out the information and forward it to selected office.



Service Request

□ G.T Annual leave for teachers #ASTU123/171111/3
Gizatie Taye | 2017-11-11T07:29:27.153
is One
Service : Request of material
Gizatie Desalegn Asm...
Waiting For Dispatching
Medium
Computer Science an...

□ G.T Annual Leave Application Form Send From School of Electrical Engineering and Computing #ASTU123/180115/2
Gizatie Taye | 2018-01-15T03:36:08.507
is One
Service : Annual Leave Application Form
Gizatie Desalegn Asm...
Under Investigation W...
Medium
Computer Science an...

□ G.T Mechanical Engineering/ Monthly Leave Permission #ASTU123/180126/1
Gizatie Taye | 2018-01-26T23:17:38.183
If You Wish To Pause Your Application Request Process And Continue Later, Your Can Do So By Saving Your Application As A Draft And Resume It Later From Draft Applications Section.
Service : Annual Leave Application Form
Gizatie Desalegn Asm...
Waiting For Dispatching
Medium
Computer Science an...

Copyright Personal © 2017-2018
Powered by Personal For Non Bussiness/Thesis Systems

Figure 6. 10 : Service Request processing at back-office at different office level

The above code indicates that the controller to display the data in view. As we have seen from figure 6.10, there is a list of services requests at back-office of the submitted request for different offices and performs by the assigned user/worker.

Chapter Seven: Evaluation, Result and Discussion of the study

In this section, we review surveys that we conducted with users, IT professionals, and software development experts to evaluate and discuss the prototype system being developed and discuss the current manual system. The current practice of processing ASTU office tasks is through a manual method of sending / receiving, approving / rejecting, and forwarding various documents from one office level to another office level, sharing the institutional knowledgebase, communication between employers and various offices, manual preparation of various institutional forms such as annual leave, teachers load, file request, and material purchase request. Later, these office tasks will not run electronically.

In addition, there are many problems with the current manual method, such as: delays in moving documents from one office level to another office, non-standard workflows and services form, i.e. inconsistency with office work procedures, iteration and duplication of work and summary preparation, lack of system approach, too many manual approvals, and reviews, time spent on non-core tasks, lack of coordination / communication between departments, schools, and senior administrative offices, delays in decision-making due to time spent in retrieving facts and information, lack of security and confidentiality of documents, lack of information officers / staff, and poor knowledge management could be avoided.

Reflect on the previous problems, we have proposed a platform framework based on the cloud (Software as a Service) and implemented a prototype system that creates a common alliance through the use of manual method to electronic office work, then bringing institutional and individual efficiency through faster decision-making, error free output, collaborative workspaces, communication speed, optimal resource utilization, scalability, and movement effectiveness through output / result orientation, quality of output, on-time delivery, and cost effectiveness (value for money) through accessibility and availability of information and through commitment. Transparency is created to provide employees with user-friendly operation, a trusted digital environment, easy file retrieval, remote capabilities, and online help / assistance.

Compared to the manual method, it follows the standard workflow and services form, the dynamic creation of forms, logging of transaction history, interactive and supported by current technologies and tools, but does not mean that it replaces the entire manual office task, but as a bridge for software developers.

ICT professionals from ASTU, colleagues, and experts in software engineering, related fields evaluated the design of the technical system give their comments and suggestions on the CBEOF system developed. They recommend that the system more users friendly and easier to use to be more effective. Therefore, the comments were positive and these comments and recommendations were left to the respective Adama Science and Technology University to the professionals who perform their functions, and the development of the full implementation will be easy for the software engineers.

Finally, this architecture and framework for sharing different institutional forms, simple communication between employers, and various office has been found through the implementation of messaging, and forum discussions, sending / receiving, approval / rejection, and routing of various documents, and building of an exchange institutional knowledge is effective and important if it is fully implemented with the comments of all interested stakeholders.

> To validate this evaluation was done on how users use the CBEOF UI and attractive interface. They practiced/used the designed user interface and other features of the developed prototype. Finally, they suggest several additional problems to be included in the system. Therefore, the evaluation questions were made with four ASTU employers from different offices, three ICT professionals working in the ASTU, three experts and eight colleagues in software engineering and related fields.
>
> To evaluate the CBEOF system, each user group uses its own pages created for them. Moreover, rate the result in terms of usability of the prototype and framework of the CBEOF system in the cloud. The framework and developed prototype evaluate and re-work the framework and modify the prototype.

Here is the evaluation result prepared from the prototype-developed system and framework of the CBEOF system. It represents Poor (1), Fair (2), Good (3), V. Good (4), Excellent (5).

Table 7. 1 : Evaluation criteria for developed prototype system and framework of the CBEOF

<table>
<tr><th><i>No</i></th><th><i>Evaluation Criteria</i></th><th><i>1</i></th><th><i>2</i></th><th><i>3</i></th><th><i>4</i></th><th><i>5</i></th></tr>
<tr><td>1</td><td>How do you rank your satisfaction with the developed prototype system?</td><td>-</td><td>2</td><td>10</td><td>5</td><td>1</td></tr>
<tr><td>2</td><td>How do you rank the dynamic form builder to design for different services?</td><td>-</td><td>-</td><td>9</td><td>7</td><td>2</td></tr>
<tr><td>3</td><td>How is the browser's response time when you start requesting services on the server?</td><td>-</td><td>1</td><td>9</td><td>8</td><td>-</td></tr>
<tr><td colspan="7"><b><i>Evaluation Criteria’s (Question 4-11, Answered by Expert in Software Engineering and Related Field)</i></b></td></tr>
<tr><td><b><i>No</i></b></td><td><b><i>Evaluation Criteria</i></b></td><td><b><i>1</i></b></td><td><b><i>2</i></b></td><td><b><i>3</i></b></td><td><b><i>4</i></b></td><td><b><i>5</i></b></td></tr>
<tr><td>4</td><td>How do you rate the problem of non-standard process solutions in the design framework and in the developed prototype system?</td><td>-</td><td>-</td><td>8</td><td>6</td><td>4</td></tr>
<tr><td>5</td><td>How do you rate the movement speed of the application and the record of the transaction history in the design framework and the developed prototype system?</td><td>-</td><td>2</td><td>7</td><td>6</td><td>3</td></tr>
<tr><td>6</td><td>How do you see the design framework and prototype system developed to solve the manual creation of forms, sending and approving / rejecting requests?</td><td>-</td><td>3</td><td>6</td><td>7</td><td>2</td></tr>
<tr><td>7</td><td>How do you see the design framework for electronic communication in the office, such as e-messaging, SMS and e-mail exchange, which resolves current communication?</td><td>-</td><td>2</td><td>9</td><td>6</td><td>1</td></tr>
<tr><td>8</td><td>How do you rank the prototype-developed system to decrease the time lost in non-core process?</td><td>-</td><td>1</td><td>4</td><td>10</td><td>3</td></tr>
<tr><td>9</td><td>How do you rate the automatic task distribution and escalation engine used to solve the problem of delaying tasks by escalating to higher-level management?</td><td>-</td><td>2</td><td>5</td><td>9</td><td>2</td></tr>
</table>

No	***Evaluation Criteria***	***Yes***	***No***	***Yes, in %***
10	Do you say the cloud-based design framework and develop the prototype system that has been implemented to create the smart environment and digital workplace?	15	3	83.3
11	Do you think that this cloud-based electronic office system has improved the institution effectiveness and transparency?	16	2	88.8

As a user of the CBEOF system in Table 7.1: shows a summary result of the evaluation for developed prototype system.

First criteria, is the satisfaction of the evaluator of the prototype-developed system rated as fair, good, very good, and excellent by 11%, 27.7%, 38.8%, and 22.2% of the respondents respectively. As we have seen from the result of the evaluation the developed prototype, most of the evaluator give their satisfaction good and very good. Moreover, most of the evaluator have a good perspective for the prototype-developed system.

Second criteria, the rating of the dynamic form builder to design for different forms is good, very good, and excellent by 27.7%, 38.8%, and 33.3% of the respondents respectively. As we have seen from the response most of the respondent responded V. Good and excellent about the dynamic form builder.

Third criteria, the browser response time starting to request a service on the server be rated as fair, good, very good and excellent by 5.5%, 50%, 33.3% and 5.5% of the respondents respectively.

Fourth criteria, the problem of non-standard process solutions in the design framework and developed prototype system rated as good, very good and excellent by 38.8%, 44.4%, and 16.6% of the respondents respectively.

Fifth criteria, the speed of movement of the request and the record of the transaction history in the design framework and developed prototype system is rated as fair, good, very good and excellent by 11%, 27.7%, 27.7%, and 33.3% of the respondents respectively. Moreover, to increase the accountability of the proposed system based on the transaction history for every request, audit data, and raising awareness of customers and users of the system and higher living standards overtime.

Sixth criteria, the design framework and the developed prototype system for solving the manual creation of forms, sending / receiving, and approving / rejecting requests is rated as fair, good, very good and excellent by 16.6%, 33.3%, 38.8%, and 11% of the respondents respectively.

Seventh criteria, the design framework for electronic communication in the office, such as E-messaging, SMS and E-mail exchange, which resolves current communication, is fair, good, very good, and excellent by 5.5%, 38.8%, 44.4%, and 16.6% of the respondents respectively. Moreover, in order to greater the collaboration by using different communication tools like SMS, E-mail, E-forum discussions among customers, departments, and other administrative officers. There is filing system, central information place, establish consistent, and important decisions. Therefore, there is common understanding how to put new thing, where to find the data, to access anywhere, and regardless of the location.

Eighth criteria, the prototype developed system to decrease the time lost in the non-core process is rated as fair, good, very good and excellent by 5.5%, 27.7%, 55%, and 16.6% of the respondents respectively.

Ninth criteria, the automatic task distribution and escalation engine used to solve the problem of delaying tasks by escalating to higher level management is rated as fair, good, very good and excellent by 11%, 27.7%, 55.5%, and 5.5% of the respondents respectively.

Tenth criteria, the cloud-based design framework and develop prototype system has been implemented to create the smart environment and digital workplace is rated as Yes, and No by 83.3%, and 16.7% of the respondents respectively.

Eleventh criteria, the cloud-based electronic office system has improved the institution effectiveness and transparency is rated as Yes, and No by 88.8%, and 10.2% of the respondents respectively. In addition, the proposed system is transparent because there is standard use of different services form, and process/workflow, the steps where to start and end for given request easily use by customers based on the institutional knowledgebase, and the customers use the services based on their roles.

In general, as we have shown in the previous discussion on the most evaluative response, when the cloud-based electronic office implemented increases effectiveness, transparency of office tasks, decreases the time lost in non-core processes and facilitates the work of the institution, the designer dynamic form for services also responded that it is a good and very good idea to solve the problems of the enormous manual preparation of forms and to have standard forms and workflows in the institution. In addition, the design framework has an automatic distribution of tasks and an escalation engine to complain to a higher level of management and to create a faster work.

Chapter Eight: Conclusion and Future Work

8.1. Conclusion

Each Ethiopian Higher Education University organizes with academic, research and administrative offices to achieve good learning, to provide researchers and community service. In Ethiopian universities there is problems related to office tasks. Therefore, we have taken ASTU to solve related to office tasks. To solve the problems, we propose the cloud-based electronic office framework and prototype implementation for the Adama Science and Technology University. In this study, we introduced the private cloud platform architecture and prototype implementation of electronic office called CBEOF for sending / processing documents from one office level to another, managing various institutional services, collaboration and messaging, dynamic form creation / generator design, dynamic dispatching of tasks and escalation engines to escalate to higher managerial office if there is any delaying of tasks, to have alerts and notifications, SMS and E-mail exchange, to have forum-discussions and managing institutional knowledgebase. Therefore, this study introduces the concept of implementing cloud computing into office tasks/works to manage and process various institution services to empower the institution employees in their profession and their technological adaptation. CBEOF is based on IBM, NIST Cloud Reference Architecture, and Microsoft Web Application Architecture. We have developed a cloud-based framework of electronic office for ASTU. For the designed architecture, a prototype system was developed on the Microsoft Azure platform and deploy on IIS local server.

This NIST architecture focused on electronic office services to increase availability at a low cost and to create a good communication and collaboration network. Various users, ICT professionals, other experts, and colleagues in software engineering and related fields evaluate the developed prototype web application. The proposed framework is just the roadmap for implementing an entire electronic office and service delivery ecosystem. It facilitates and creates a digital workplace solution and an intelligent environment. Designed and synchronized with the needs of modern Ethiopian Universities, it was conceived as an instrument for the next generation Ethiopian University.

8.2. Future Work

The proposed framework is not the complete system developed, but we prepare the prototype. Moreover, to send and receive the various documents from one department to another department, department to school, school to senior managerial office. Top-down or down-top communication. To prepare the dynamic form builder, to manage the institutional knowledgebase to replace the manual method of communication channels such as the use of E-messaging, and E-forum discussions.

The proposed system was developed for ASTU. The service delivery infrastructures are prepared as management of the university, departments/units, and schools of the institution implemented in the prototype. In addition, employers / users of the institution and different services are created. After that, when identifying the number of attributes of the organization (ASTU), design the form builder/generator dynamically for different forms and then distribute them in different departments, teams, and other offices depending on their request. However, the presented architecture and the prototype implemented in this thesis must be improved and evaluated for a more complete and worthy electronic office. Furthermore, even if the integration of this CBEOF system with the signature tool was shown in the proposed framework and architecture, this feature is not implemented in the prototype. When the electronic office implements in higher education institutions we recommend using community cloud. Because the Ethiopian higher education institutions have shared concerns, and requirements. However, the proposed framework for ASTU, we recommend to use private cloud.

In addition, there are different components or functionalities of e-office not implemented such as: E-forum-discussions and institutional knowledgebase, alerts and notifications to notify clients for different status, implement the preparation of different letter templates and integrate with the signature tool, and design workflow for different business processes, develop electronic messaging and collaboration, escalation and dispatcher engines, SMS and Email exchange.

Here is a list of some of the future work such as:

- The proposed framework is developed for ASTU and should be implemented in each higher education institution in the future and evaluated with an iterative reflection of employers / users, developers, experts in software engineering and related fields. So that the proposed framework is modified based on feedback and deploy using community cloud deployment.

- The study on measuring of performance and effectiveness of the users by having different users.

References

[1] R.Kumar, M.K. Sharma, et al, "Impact of ICT in Growth and Development of rural System in Uttarakhand," *World Rural Observations*, 2011.

[2] Mulatu, Abdi, "E-complaint Management System and GIS Mapping for Addis Ababa City Administration," 2011.

[3] P.Markellou, A.Panayiotaki, et al, "E-Government: Applications Levels," [Online]. Available: elab.csd.auth.gr/bci1/Panhellenic/58markellou.pdf. [Accessed 20 March 2017].

[4] [Online]. Available: www.trackcustomercomplaints.com. [Accessed 21 March 2017].

[5] Lallana, Emmanuel C., "mGovernment: Mobile/Wireless Applications in Government," [Online]. Available: http://www.egov4dev.org/mgovernment/. [Accessed 09 March 2017].

[6] *Case Study on implementation of eOffice in Mantralaya, Mumbai, Government of Maharashtra*, 2013 – 2014.

[7] Michael Roper; Laura Millar, "MANAGING ELECTRONIC RECORDS," INTERNATIONAL COUNCIL ON ARCHIVES, United Kingdom., 1999.

[8] Logicsoft International Private Limited, "eOffice- An administrative information management system".

[9] WeGO Secretariat Office, "Final Report for ADDIS ABABA e-Office System," 2013.

[10] Dharmarajan, N., Lokman, A.M., Zainol, Z., "E-OFFICE For UiTM: A Survey Analysis," vol. Vol8(1), pp. 71-79, 2006.

[11] Eugene, G., "Cloud Computing Models," *Massachusetts Intitute Technol*, p. 1–81, January 2013.

[12] Lead, S. Analyst and C. P., "Public Cloud the Key to a Successful Digital Transformation".

[13] Mell, P., and Grance, T, "The NIST definition of cloud computing. National Institute of Standards and Technology," 2009.

[14] Kepes, Ben, "UNDERSTANDING the Cloud Computing Stack SaaS, Paas, IaaS," 2011.

[15] Tim Walters PMP, "SAS® as SaaS - The Benefits and Challenges of Implementing an

Enterprise Scale SAS® Data Warehouse and Business Intelligence Shared Service," *SAS Global Forum 2009*, 261-2009.

[16] F.Liu, J.Tong, J.Mao, R.Bohn, J.Messina, L.Badger and D.Leaf, "NIST Cloud Computing Reference Architecture," *U.S. Department of Commerce*, pp. 1-35, September 2011.

[17] DEMEKE, SEWALE BELACHEW, "A CLOUD COMPUTING FRAMEWORK FOR ETHIOPIAN HIGHER EDUCATION INSTITUTIONS," 2012.

[18] Pardeshi, V. H., "Cloud Computing for Higher Education Institutes: Architecture, Strategy and Recommendations for Effective Adaptation," vol. 11, no. 14, p. 589–599, 2014.

[19] Tahvildari, M. Hamdaqa and L., "Cloud Computing Uncovered: A Research Landscape," vol. 86, 2012.

[20] Erkoç, S. B. K. Mehmet Fatih, "Cloud Computing for Distributed University Campus: A Prototype Suggestion," 2011.

[21] Rastogi, Dr.Ashish, "A Model based Approach to Implement Cloud Computing in E-Governance," vol. Volume 9– No.7, pp. Page 1-4, November 2010.

[22] G.Sahoo, K.Mukherjee, "Cloud Computing: Future Framework for e-Governance," vol. Volume 7– No.7, pp. Page 1-4, October 2010.

[23] K.Veljanovska, V.Zdravevska, "E-Government Based on Cloud Computing," vol. Vol. 4, pp. Page 1-5, No. 4 April 2013.

[24] S.Suchaiya, S.Keretho, "Analyzing National e-Government Interoperability Frameworks: A Case of Thailand," pp. Page 1-6., 2014.

[25] M.Kumar, R.Bhatt and K.Singh Vaisla,, "Central Architecture Framework for e-Governance System in India using ICT infrastructure," 2015.

[26] Mohammed, Endris, "Participatory Indigenous Knowledge Management Architecture for Traditional Medicine," Adama Science and Technology University, 2016.

[27] Kothari, C. R., "Research methodology: methods and techniques," *New Age International*, 2008.

[28] NJ, Hoboken, "Introduction to qualitative research methods 3rd ed," *US: John Wiley & Sons Inc*, vol. 34, no. 6, pp. 687-70, 1998.

[29] Uma D and Pansiri, J.ogulu, "Mixed methods: A research design for management doctoral

dissertations," *Management research review,* vol. 34, no. 6, pp. 687-70, 2011.

[30] Fallman, Daniel, "Why Research-oriented Design Isn’t Design oriented research," *Knowledge, Technology & Policy,* vol. 20, no. 3, pp. 193-200, January 2007.

[31] etal, Hevner A., "Design Science in Information Systems Research," *MIS Quarterly,* vol. 28, no. 1, pp. 75-105, 2004.

[32] Mattsson, M. et al (n.d.), "Software Architecture Evaluation Methods for Performance, Maintainability, Testability, and Portability.".

[33] Bell, D, "Software Engineering for students A Programming Approach fourth edition," 2005.

[34] Ralhan, Puneet, "Prototyping," 18 12 2000.

[35] Patton, M. Q, Qualitative evaluation and research methods (3rd Ed.), Newbury Park, CA: Sage Publications, 2001.

[36] "Stratified sampling calculator".Available: http://stratifiedsampling.orgfree.com/calc. php.

[37] Keith G. Diem, Ph.D., "A Step-by-Step Guide to Developing Effective Questionnaires and Survey Procedures for Program Evaluation & Research," *Rutgers Cooperative Extension Program.*

[38] Dr Thomas F Burgess, "A general introduction to the design of questionnaires for survey research," *INFORMATION SYSTEMS SERVICES,* vol. 1.1, May 2001.

[39] Mäenpää, Jouni, "Cloud computing with the Azure platform," 27 04 2009.

[40] Sullivan, D., "PaaS Providers List: Comparison and Guide," toms it pro, 31 January 2014. Available: http://www.tomsitpro.com/articles/paas-providers,1-1517.html. [Accessed 19 September 2017].

[41] Zahariev, Alexander, "Google App Engine," 27 04 2009.

[42] tutorialPoint.Pvt.Ltd,"tutorialPoint,"Available:https://www.tutorialspoint.com/openshift/ index.htm. [Accessed 9 December 2017].

[43] Wikipedia. (n.d.), "Microsoft Visual Studio," wikipedia.org:. Available: https://en. wikipedia.org/wiki/Microsoft_Visual_Studio. [Accessed 15 November 2017].

[44] "All About Visual Studio .Net," Microsoft Available: http://www.c-sharpcorner.com/

blogs/all-about-visual-studio-net1. [Accessed 20 01 2018].

[45] IBM Cloud Computing Reference Architecture (CCRA) 4.0 Overview, IBM, 2014.

[46] "NIST cloud reference architecture".

[47] Microsoft Corporation, "Web Application Architecture Guide," Microsoft Corporation, 2008.

[48] Keeling, M, "Reflections on Software Engineering: Dealing with Constraints in Software Architecture Design," neverletdown, 22 October 2014. [Online]. Available: http://www.neverletdown.net/2014/10/dealing-with-constraints-in-software-architecture.html. [Accessed 27 May 2017].

[49] Ali, M. S. (n.d.), "Cloud Management Mechanisms," SlideShare, [Online]. Available: http://www.slideshare.net/SajjadAliMOHAMMED/cloud-management-mechanisms. [Accessed 12 October 2017].

[50] Ali, M. S. (n.d.), "Cloud Security Mechanisms," slide Share, [Online]. Available: http://www.slideshare.net/SajjadAliMOHAMMED/cloud-security-mechanisms. [Accessed 12 October 2017].

[51] Tutorials Point (I) Pvt. Ltd, "Software Architecture and Design," [Online]. Available: https://www.tutorialspoint.com/software_architecture_design/interaction_oriented_architecture.htm. [Accessed 9 December 2017].

Appendixes

Appendix A: Interview Questions

Interview: Academic/Faculty Deans, ICT professionals, Administrative Staffs, and Teachers should respond questions.

1. How many incoming/outgoing documents and a letter from/to your office in a day?
2. Does your university use any type of software for the purpose of accepting or analyzing reports?
3. In an average how many days/ how much time in (hours/minutes) it takes to deliver one document like letter from one office to another internal office in the school/department, and higher managerial office?
4. How the documents are indexed and searched and how do you know the latest document?
5. How do you know the status of communication in the internal department, school, and with senior managerial office e.g. arrival of the letter or document?
6. What are the different services to be delivered at your office level related to office work that you provide them to the university community?
7. How can you route the different documents during internal communication among departments and the senior management?
8. What are the most repetitive and paperwork intensive processes/ workflows?
9. Does your institution focus on and plan to use the new emerged technology like cloud computing, which aims to increase the efficiency and quality of services provided for the university community?
10. What kind of cloud computing deployment model and cloud platform service provider is the more suitable for the proposed system? (for expert in cloud computing and related field)

Appendix B: Questionnaire Questions

Questionnaire to be completed by University Higher Managerial office, Academic School / Faculty Deans, ICT professionals, Research Dean, Administrative / Academic Department Heads, Teachers and Post-graduate students.

Job Title: ------------------------------ Educational Level: --

Position: ------------------------- Work Experience: ------------------- Age: ---------- Sex: ----------

The purpose of this self-administered questionnaire is to investigate the communication among one office level with another office level to gather information on the process of office task specifically on how to send/receive, manage and routing of different documents, to know the most repetitive form-based requests and approval process, collaboration and messaging. Your responses will not be identified with you personally. All responses that you provide will be treated confidentially. Therefore, I would like to requests you to fill this questionnaire carefully. The information that you provide me through the questionnaire would be of paramount importance to the research we are undertaking.

1. How do you send/receive different documents from department to/from university management and rank the most dominant used (*Giving multiple ranks is possible)?*

□ A) Colleague/friend

□ B) Email

□ C) Personal Messenger

□ D) Fax

□ E) by him/her self

□ F) Any Application software

If any other use specifies here …...

2. Do you believe that the current communication system like paper document, phone, fax, email, messaging is efficient? *(Please tick one)*

□ A) Yes

□ B) No

If your answer is (b), specify the main problem?

____

3. Did the documents and letters have been losing opportunity from your office? *(Please tick one)*

□ A) Yes □ B) No

4. Do you believe that on using of paperless office working place used to create the smart environment? *(Please tick one)*

□ A) Yes □ B) No

If your answer is (A) how ____

5. Is there any log file that can hold the transaction history on your side to know the status of delivery for your request? *(Please tick one).*

□ A) Yes □ B) No

6. Does the current system support an easy way to search or find documents and letters?

□ A) Yes □ B) No

7. Is there any automated office management system in your university to route, send/receive, and approving/rejecting documents from one office level to another office level and to process a form-based request? *(Please tick one).*

□ A) Yes □ B) No

If your answer is (A) please write the system here______________________________

8. Do you believe to improve your effectiveness and efficiency, if electronic office management system is implemented in your university?

□ A) Yes □ B) No

9. Is there any central repository of the institutional knowledgebase and best practices?

□ A) Yes □ B) No

If your choice A) please write the repository____________________________________

10. In average, how much time (hour/minute) or how many days has been taken to process in sending/receiving, and approving the documents and letters. ________________________

11. In average, is there any delay of the current manual system in movement of file from one office level to another office level and to give decision-making?

□ A) Yes □ B) No

12. Can you say the current manual system is following a standard workflow to process the given task such as annual leave, request material, etc...?

□ A) Yes □ B) No

13. Is there any way to report/complain if a task is delayed or does not perform in time?

□ A) Yes □ B) No

If you select A) write the method here ____

14. What do you think, if there is an automatic dispatching/distribution system of letters, documents, and different form based request?

____

15. What do you think on using an electronic office management system to communicate easily among employees with in the university and to request, review, and approve/reject different documents and most repetitive form based request?

____

____

____

Appendix C: Observation on Services

No	Observed Services	Customer Satisfaction	Offices	Currently use
1	The process of documents and letters from one office to another office.		Computer Science Engineering Department, Registrar, Academic V/President, and ICT office.	
2	The current communication system			
3	The most repetitive request forms			
4	Technology adoption and knowledgebase.			

Appendix D: Evaluation Criteria Questions

Here are the evaluation criteria prepared from the prototype-developed system and design framework of CBEOF. It represents Poor (1), Fair (2), Good (3), V. Good (4), Excellent (5).

Evaluation Criteria (Question 1-3, Answered by ASTU users and ICT professionals)

1. How do you rank your satisfaction with the developed prototype system?

   A. Fair B. Good C. V. Good D. Excellent

2. How do you rank the dynamic form builder to design for different services?

   A. Fair B. Good C. V. Good D. Excellent

3. How is the browser's response time when you start requesting services on the server?

   A. Fair B. Good C. V. Good D. Excellent

Evaluation Criteria (Question 4-11, Answered by Expert in Software Engineering and Related Field)

4. How do you rate the problem of non-standard process solutions in the design framework and in the developed prototype system?

   A. Fair B. Good C. V. Good D. Excellent

5. How do you rate the movement speed of the application and the record of the transaction history in the design framework and the developed prototype system?

   A. Fair B. Good C. V. Good D. Excellent

6. How do you see the design framework and prototype system developed to solve the manual creation of forms, sending / receiving and approving / rejecting requests?

   A. Fair B. Good C. V. Good D. Excellent

7. How do you see the design framework for electronic communication in the office, such as e-messaging, SMS and e-mail exchange, which resolves current communication?

   A. Fair B. Good C. V. Good D. Excellent

8. How do you rank the prototype-developed system to decrease the time lost in non-core process?

   A. Fair B. Good C. V. Good D. Excellent

9. How do you rate the automatic task distribution and escalation engine used to solve the problem of delaying tasks by escalating to higher-level management?

   A. Fair B. Good C. V. Good D. Excellent

10. Do you think that the cloud-based design framework and develop the prototype system that has been implemented to create the smart environment and digital workplace?

    A. Yes B. No

11. Do you think that this cloud-based electronic office system has improved the institution effectiveness and transparency?

    A. Yes B. No

Appendix E: Working with Microsoft Window Azure Platform

In this section, we discussed how to work with Microsoft window azure to develop and host application. So, to create the proposed application on window azure platform services: we have used App service; that is the platform services is provided to the user by app service. The APP services enable developers to focus on their code and reach a stable, highly scalable production state quickly. With APP services, we can develop the web app, mobile app, API APP, logic API etc. Among this, we have selected the web app services.

To create the web app services on the Azure app service platform, we have followed the following three steps.

1. Selecting app service type

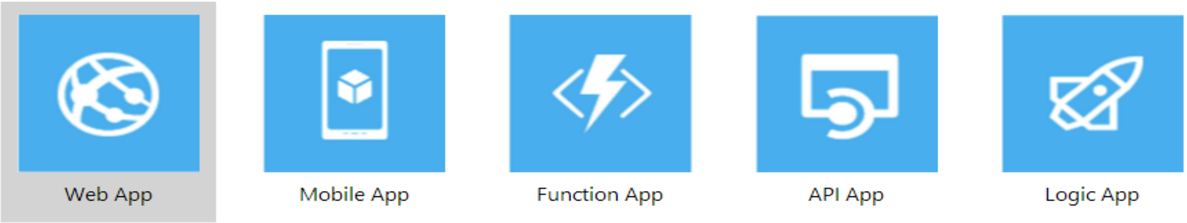
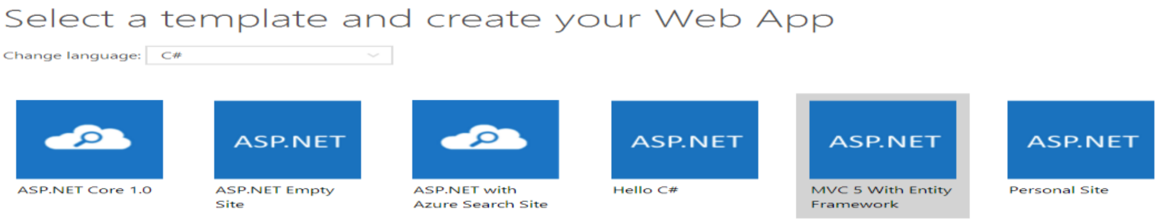


Figure 1 : Web App Service Type

As we shown in above figure, we have selected web app service. Azure Web Apps enable developers to easily deploy and scale enterprise grade web applications written in a variety of languages and integrated with a multitude of services without ever worrying about infrastructure management.

2. Selecting a language in window azure platform, there is Visual Studio integration tools that allows for the developer for creating, deploying, and debugging applications in online and at developer local machine. The following diagram shows the language selected and its template.

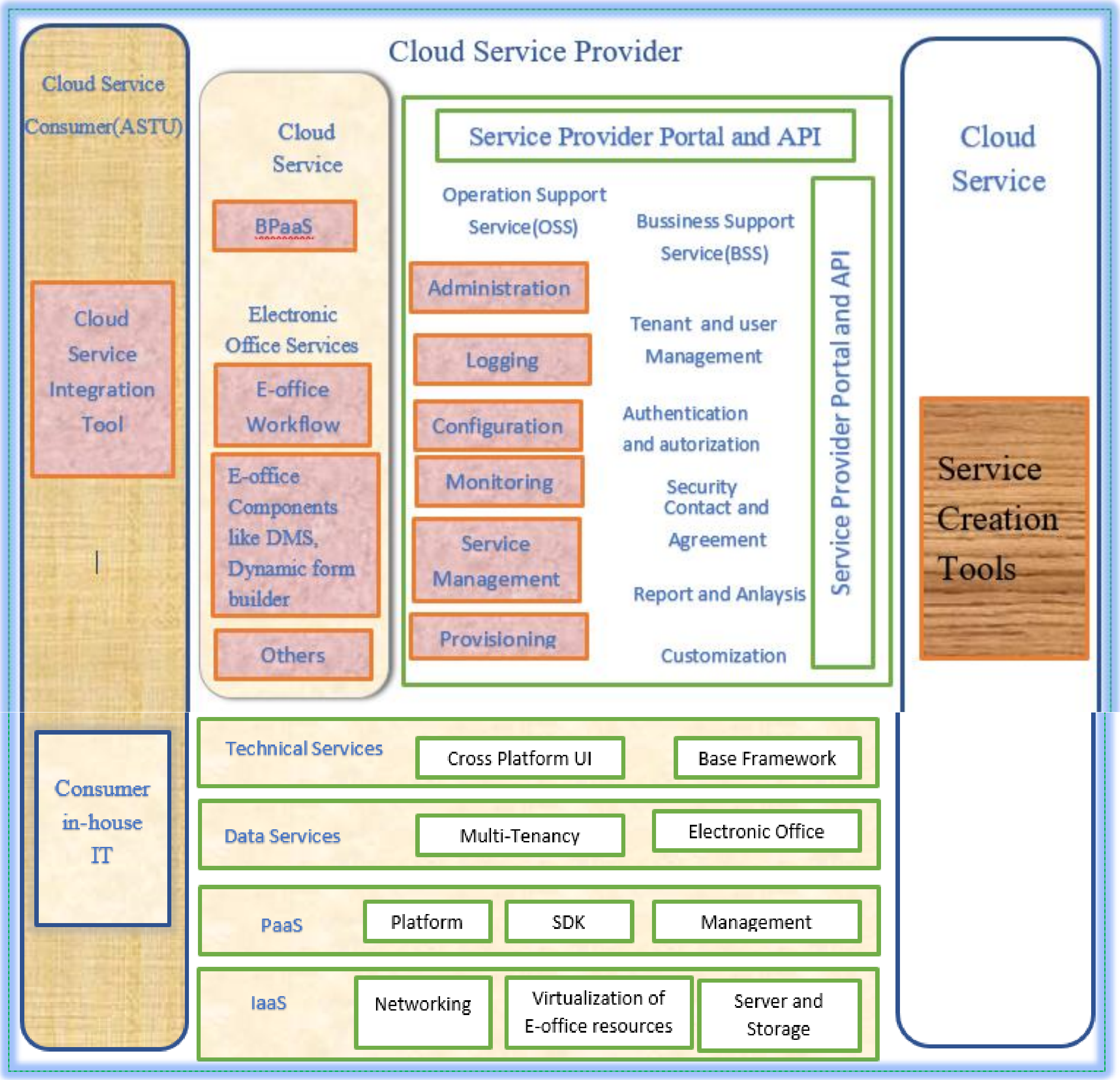

Select a template and create your Web App

Change language: C#

Figure 2 : Selection of Language in Window Azure Platform

3. Creating the web app services and other related information to deploy the application.

Appendix F: Functional Overview of CBEOF on Cloud Architecture

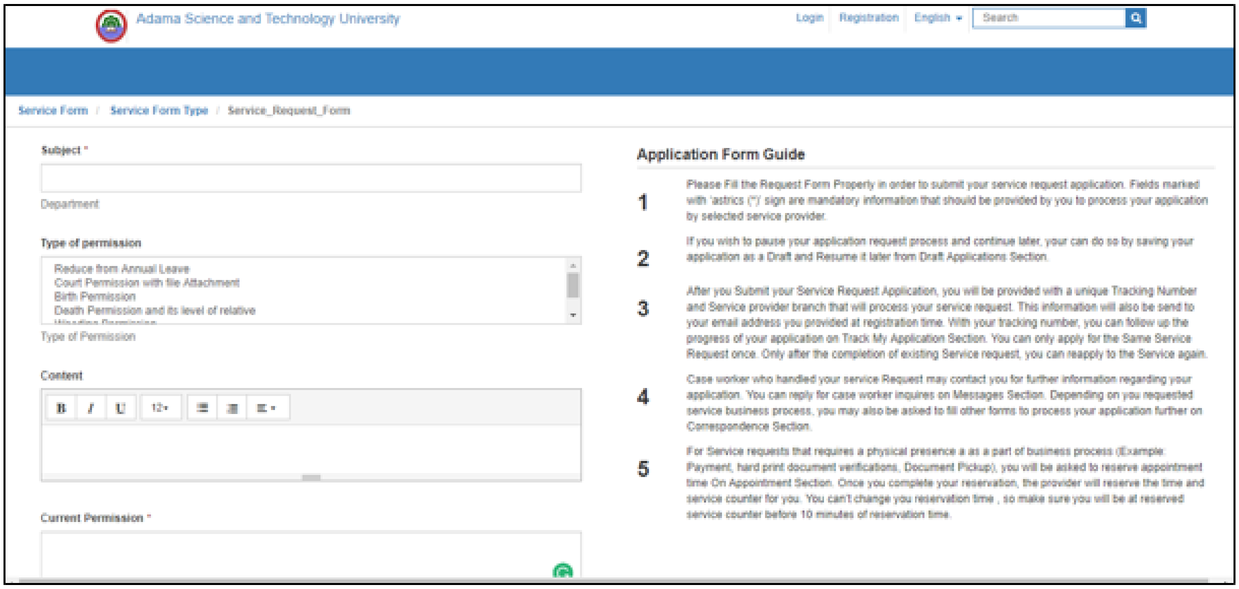


Cloud Service Provider
Cloud Service Consumer(ASTU)
Cloud Service Integration Tool
Consumer in-house IT
Cloud Service
BPaaS
Electronic Office Services
E-office Workflow
E-office Components like DMS, Dynamic form builder
Others
Service Provider Portal and API
Operation Support Service(OSS)
Bussiness Support Service(BSS)
Administration
Logging
Configuration
Monitoring
Service Management
Provisioning
Tenant and user Management
Authentication and autorization
Security Contact and Agreement
Report and Anlaysis
Customization
Service Provider Portal and API
Cloud Service
Service Creation Tools
Technical Services
Cross Platform UI
Base Framework
Data Services
Multi-Tenancy
Electronic Office
PaaS
Platform
SDK
Management
IaaS
Networking
Virtualization of E-office resources
Server and Storage

Figure 3 : Overview of functional view of CBEOF (Source: adapted from [45]*)*

As shown in the above diagram (Figure 3), the cloud based EOF system will work by logging into the service provider's portal and by ordering electronic office services through the university. This service has been created by the cloud service provider or simply by Microsoft Azure and it is a web based application accessible using web browser. Then the Microsoft Azure which is a cloud service provider in this case, will validate client request through the BSS like authorization and authentication, and if the validation is okay, it will provision the request through the OSS (i.e., provisioning, monitoring, and automation management services). In this case, the OSS is used to provide the service; the BSS is used to validate the request; and the two components found in the left and right side of the diagram which are cloud service consumer and cloud service creator, will provide interfaces and tools for the end users and service creators. The infrastructure where the cloud based CBEOF will be implemented will support the different types of cloud (IaaS, PaaS, SaaS, BPaaS).

Appendix G: Sample User Interface


Adama Science and Technology University
Login
Registration
English ▾
Search
Service Form / Service Form Type / Service_Request_Form
B
I
U
12▾

Subject *

Department

Type of permission

Reduce from Annual Leave
Court Permission with file Attachment
Birth Permission
Death Permission and its level of relative

Type of Permission

Content

Current Permission *

Application Form Guide

1 Please Fill the Request Form Properly in order to submit your service request application. Fields marked with 'astrics (*)' sign are mandatory information that should be provided by you to process your application by selected service provider.

2 If you wish to pause your application request process and continue later, your can do so by saving your application as a Draft and Resume it later from Draft Applications Section.

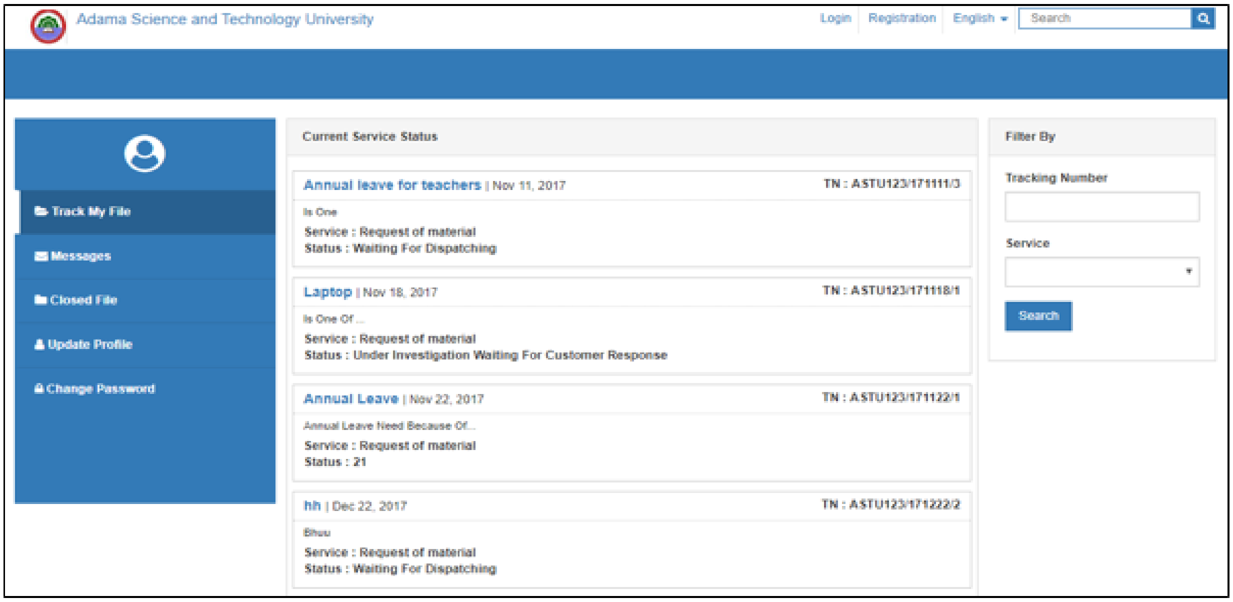
3 After you Submit your Service Request Application, you will be provided with a unique Tracking Number and Service provider branch that will process your service request. This information will also be send to your email address you provided at registration time. With your tracking number, you can follow up the progress of your application on Track My Application Section. You can only apply for the Same Service Request once. Only after the completion of existing Service request, you can reapply to the Service again.

4 Case worker who handled your service Request may contact you for further information regarding your application. You can reply for case worker inquires on Messages Section. Depending on you requested service business process, you may also be asked to fill other forms to process your application further on Correspondence Section.

5 For Service requests that requires a physical presence a as a part of business process (Example: Payment, hard print document verifications, Document Pickup), you will be asked to reserve appointment time On Appointment Section. Once you complete your reservation, the provider will reserve the time and service counter for you. You can't change you reservation time , so make sure you will be at reserved service counter before 10 minutes of reservation time.

Figure 4 : Submitting Service Request Form

As shown from figure 4, the form that used to submit the information for selected service.



Current Service Status

Annual leave for teachers | Nov 11, 2017 TN : ASTU123/171111/3
Is One
Service : Request of material
Status : Waiting For Dispatching

Laptop | Nov 18, 2017 TN : ASTU123/171118/1
Is One Of ...
Service : Request of material
Status : Under Investigation Waiting For Customer Response

Annual Leave | Nov 22, 2017 TN : ASTU123/171122/1
Annual Leave Need Because Of...
Service : Request of material
Status : 21

hh | Dec 22, 2017 TN : ASTU123/171222/2
Bhuu
Service : Request of material
Status : Waiting For Dispatching

Filter By

Tracking Number

Service

Figure 5 : Manage personal file page

As shown from the figure5, managing the personal file to identify active, closed, total request and to replay messages for selected service.

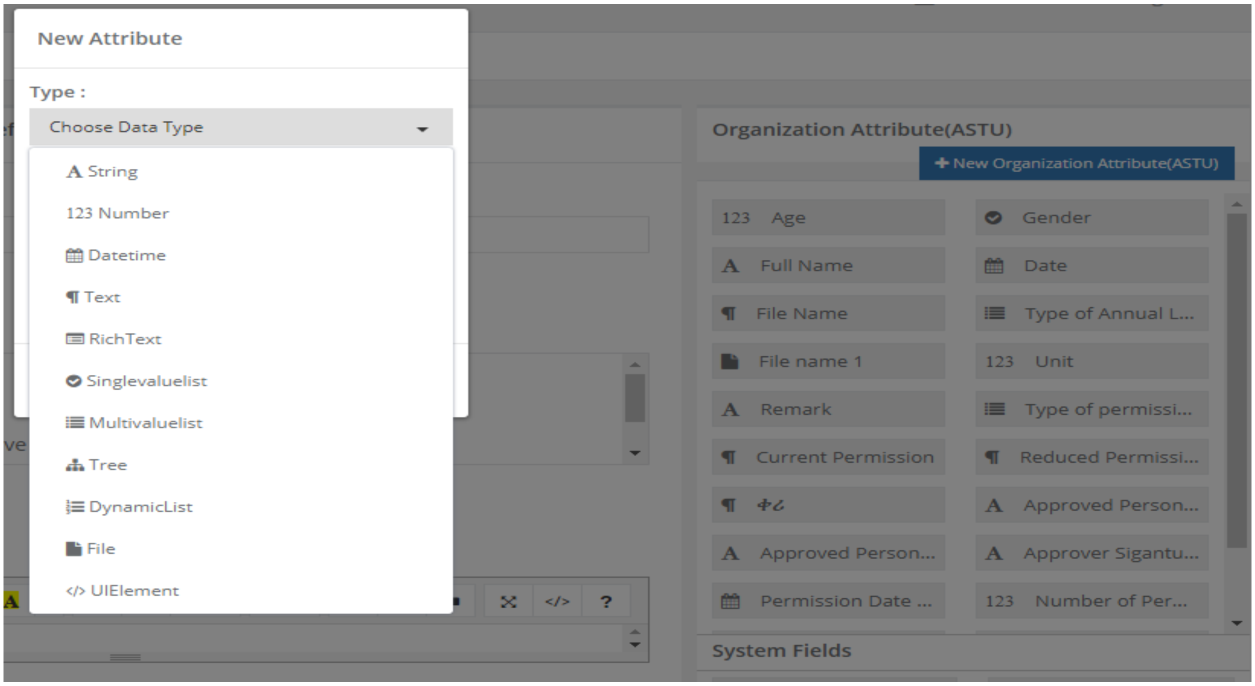
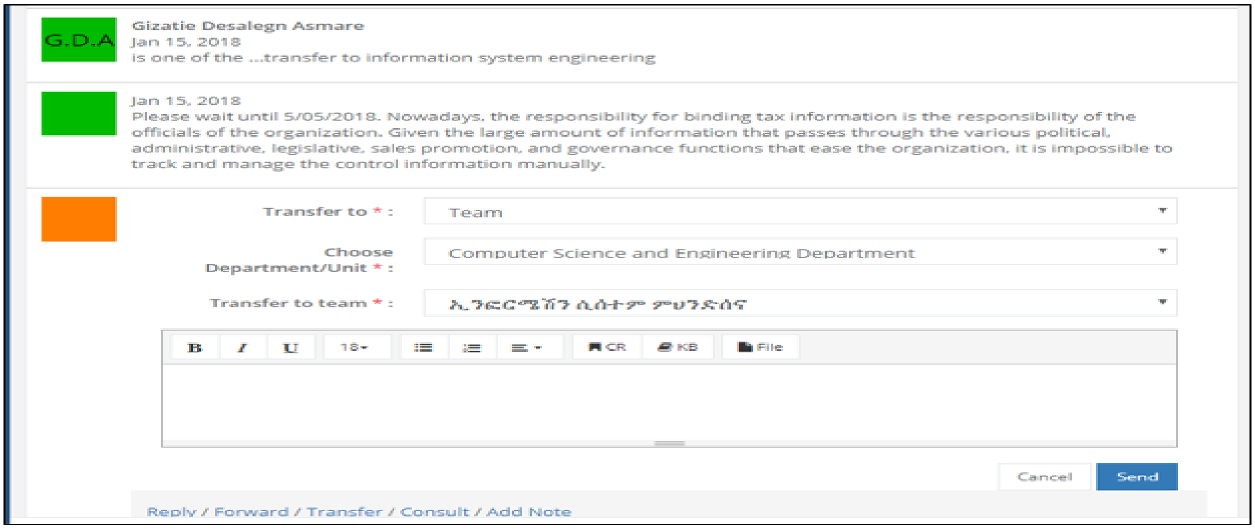


Figure 6 : Create organizational attribute (ASTU)

As shown figure 6, to create the organizational attribute (ASTU) by selecting the appropriate data type for given attribute.



G.D.A Gizatie Desalegn Asmare
Jan 15, 2018
is one of the ....transfer to information system engineering

Jan 15, 2018
Please wait until 5/05/2018. Nowadays, the responsibility for binding tax information is the responsibility of the officials of the organization. Given the large amount of information that passes through the various political, administrative, legislative, sales promotion, and governance functions that ease the organization, it is impossible to track and manage the control information manually.

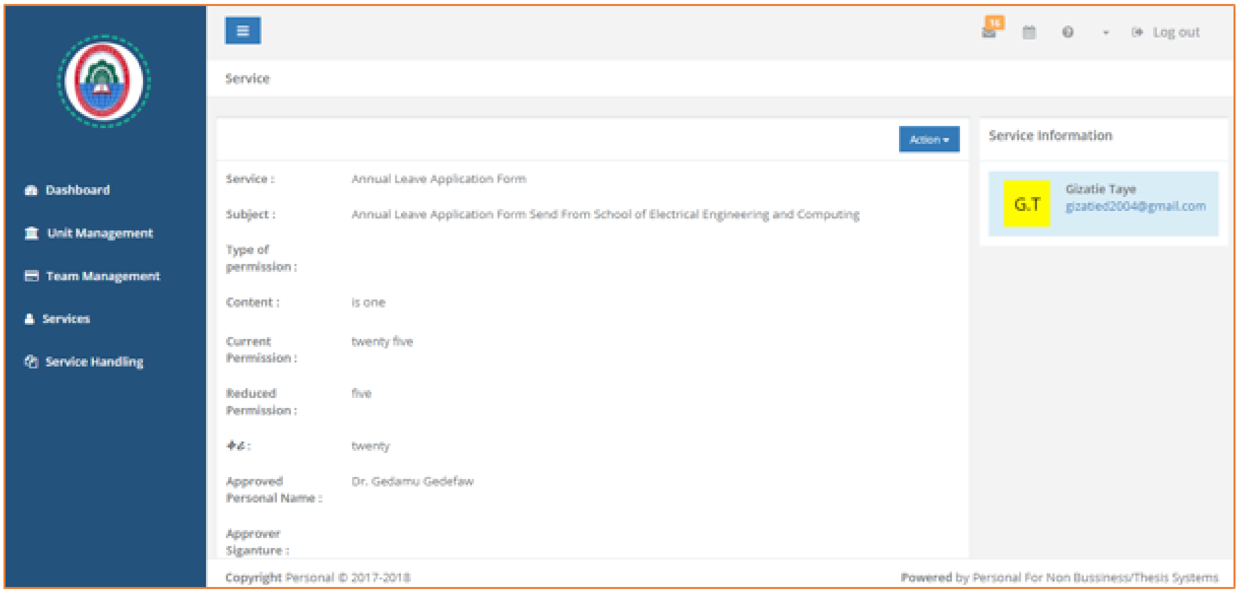
Transfer to * : Team

Choose Department/Unit * : Computer Science and Engineering Department

Transfer to team * : ኢንፎርሜሽን ሲስተም ምህንድስና

Figure 7 : Transfer the service request to department and team/division

The above figure 7, shows that used to transfer messages of the request to other department/team.



Service

Service : Annual Leave Application Form

Subject : Annual Leave Application Form Send From School of Electrical Engineering and Computing

Type of permission :

Content : is one

Current Permission : twenty five

Reduced Permission : five

ቀሪ : twenty

Approved Personal Name : Dr. Gedamu Gedefaw

Approver Siganture :

Service Information

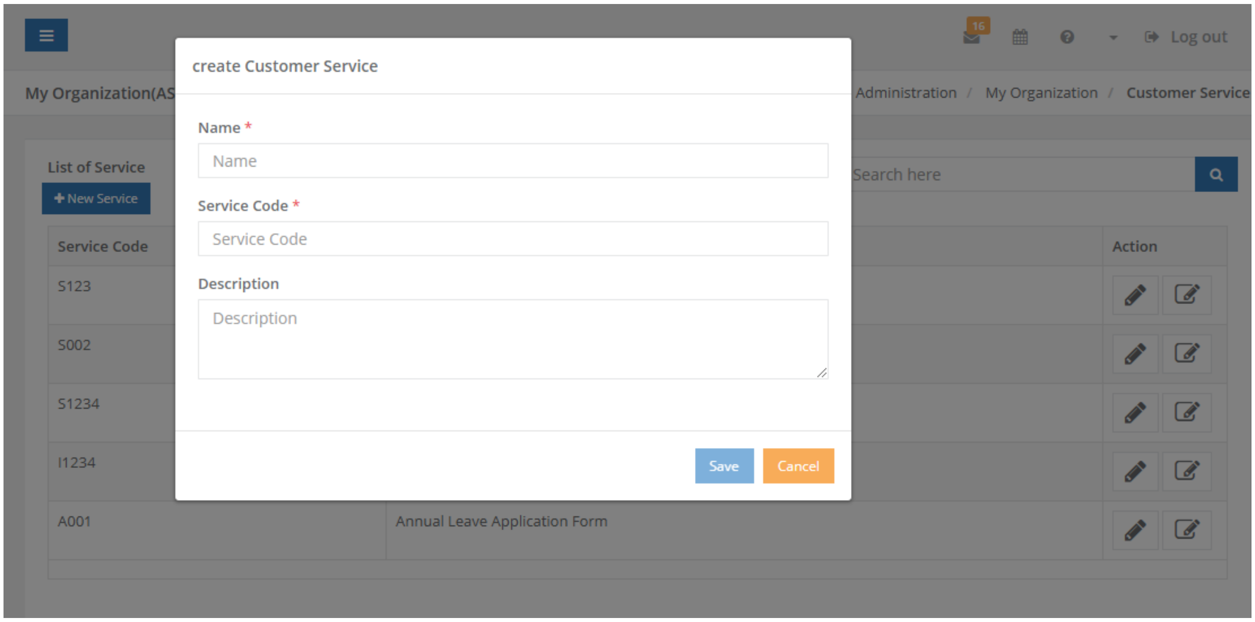
G.T Gizatie Taye gizatied2004@gmail.com

Copyright Personal © 2017-2018

Powered by Personal For Non Bussiness/Thesis Systems

Figure 8 : Detail of the service request

The figure 8 indicates that the detail of the information of the service request.



create Customer Service

Name *

Name

Service Code *

Service Code

Description

Description

Service Code		Action
S123		
S002		
S1234		
I1234		
A001	Annual Leave Application Form	

Figure 9 : Create Organizational Service (ASTU)

The above figure 9, shows that to create the organizational service of ASTU by filling the name, code and its description of the information.

Appendix H: Sample Code

The sample code is about the creating the dynamic form builder/generator: back-end implementation.

```
using eOffice.Config;
using eOffice.Design;
using eOffice.Entity.FormBuilderV2;
using eOffice.Entity.FormBuilderV2.Enumeration;
using eOffice.Facade.FormBuilder.Contract;
using System;
using System.Collections.Generic;
using System.Data.Entity;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace eOffice.Core.FormBuilderV2
{
    public class FormBuilder : IFormBuilder
    {
        DbContext context;
        Repositories.FormBuilder.Contract.IFormBuilderRepository repository;
        public FormBuilder()
        {
            this.repository = new Repositories.FormBuilder.FormBuilderRepository(new Context(new eOfficeEntityConnection()).Contexts());
        }
        #region Attibutes
        public Entity.FormBuilderV2.Attribute CreateAttribute(Entity.FormBuilderV2.Attribute attribute, int userID)
        {
            if (attribute == null || String.IsNullOrEmpty(attribute.Name) || attribute.Organization == null || attribute.Organization.ID <= 0)
```

```
                throw new InvalidOperationException("Invalid Data");
            if (this.repository.DoesAttibuteExist(attribute.Name, attribute.Organization.ID))
            {
                attribute.IsFailed = true;
                return attribute;
            }
            FormBuilderAttribute _attibute = attribute.MapToModel<FormBuilderAttribute>();
            _attibute.CreatedBy = userID;
            _attibute.CreatedOn = DateTime.Now;
            try
            {
                this.repository.Add(_attibute);
                this.repository.UnitOfWork.SaveChanges();
                attribute = new Entity.FormBuilderV2.Attribute(_attibute);
                attribute.IsFailed = false;
                return attribute;
            }
            catch (Exception ex)
            {
                attribute.IsFailed = true;
                attribute.SystemMessage.Exception = ex;
                attribute.SystemMessage.ExceptionMessage = ex.Message;
                return attribute;
            }
        }
        #region Form
        public Form CreateForm(Form form, int userID)
        {
            if (form == null || String.IsNullOrEmpty(form.Name) || form.Organization == null ||
form.Organization.ID <= 0)
                throw new InvalidOperationException("Invalid Data");
```

```
if (this.repository.DoesFormExist(form.Name, form.Organization.ID))
{
    form.IsFailed = true;
    return form;
}
FormBuilderForm _from = form.MapToModel<FormBuilderForm>();
_from.CreatedBy = userID;
_from.CreatedOn = DateTime.Now;
try
{
    this.repository.Add(_from);
    this.repository.UnitOfWork.SaveChanges();
    form = this.GetForm(_from.FormBuilderFormID);
    form.IsFailed = false;
    return form;
}
catch (Exception ex)
{
    form.IsFailed = true;
    form.SystemMessage.Exception = ex;
    form.SystemMessage.ExceptionMessage = ex.Message;
    return form;
}}}
```

The sample code is about the creating the dynamic form builder/generator: front-end (UI) implementation.

```
(function () {
  'use strict';
  angular
    .module('formBuilderModule')
    .controller('FormBuilderController', FormBuilderController);
  FormBuilderController.$inject = ['$log', '$state', '$stateParams', '$uibModal', '$timeout',
'formBuilderService', 'messaging', 'formBuilderMessage'];
  function FormBuilderController($log, $state, $stateParams, $uibModal, $timeout,
formBuilderService, messaging, formBuilderMessage) {
    var vm = this;
    var dataTypes = [];
    vm.title = 'FormBuilderController';
    vm.formAttributes = [];
    vm.formType = 'new';
    vm.genericAttributes = [];
    vm.organizationAttributes = [];
    vm.uiAttributes = [];
    vm.selectedAttribute = {};
    vm.selectedFormAttribute = {};
    vm.createAttribute = createAttribute;
    vm.saveForm = saveForm;
    vm.preview = preview;
    activate();
    function activate() {
     dataTypes = [
      "String",
      "Number",
      "Datetime",
      "Text",
      "RichText",
      "Singlevaluelist",
      "Multivaluelist",
      "Tree",
      "DynamicList",
      "File",
      "UIElement"
     ]
function createAttribute() {
  modalAttribute();
}
function modalAttribute(selectedAttribute) {
  var oldSelectedAttribute = selectedAttribute;
  var modalInstance = $uibModal.open({
    size: 'sm',
    templateUrl: 'app/form-builder/views/modal-attribute.html',
    controller: 'ModalAttributeFormBuilderController',
    controllerAs: 'attribute',
    resolve: {
      selectedAttribute: function () {
        return selectedAttribute;
      }}
  });
  modalInstance.result.then(
    function (selectedAttribute) {
      var index = _.indexOf(vm.organizationAttributes, _.find(vm.organizationAttributes, oldSelectedAttribute));
      if (index) {
        // Replace item at index using native splice
        vm.organizationAttributes.splice(index, 1, selectedAttribute);
      }
      else {
        vm.organizationAttributes.push(selectedAttribute);
      }}}
})();
```

Appendix G: Database Script

```
USE [eOfficeDB2]

GO
/****** Object:  Table [dbo].[FormBuilderAttributes]    Script Date: 12/19/2017 11:20:19 PM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[FormBuilderAttributes](
	[FormBuilderAttributeID] [int] IDENTITY(1,1) NOT NULL,
	[Name] [nvarchar](max) NULL,
	[OrganizationID] [int] NULL,
	[FormBuilderDataTypeID] [tinyint] NOT NULL,
	[Code] [nvarchar](max) NULL,
	[IsDeleted] [bit] NOT NULL,
	[CreatedBy] [bigint] NOT NULL,
	[CreatedOn] [datetime] NULL,
	[UpdatedBy] [bigint] NULL,
	[UpdatedOn] [datetime] NULL,
 CONSTRAINT [PK_dbo.FormBuilderAttributes] PRIMARY KEY CLUSTERED
(
	[FormBuilderAttributeID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

GO
/****** Object:  Table [dbo].[FormBuilderDataTypes]    Script Date: 12/19/2017 11:20:19 PM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[FormBuilderDataTypes](
	[FormBuilderDataTypeID] [tinyint] NOT NULL,
	[DataType] [nvarchar](max) NULL,
	[IsDeleted] [bit] NOT NULL,
	[CreatedBy] [bigint] NOT NULL,
	[CreatedOn] [datetime] NULL,
	[UpdatedBy] [bigint] NULL,
	[UpdatedOn] [datetime] NULL,
 CONSTRAINT [PK_dbo.FormBuilderDataTypes] PRIMARY KEY CLUSTERED
(
	[FormBuilderDataTypeID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

GO
```

```
/****** Object:  Table [dbo].[FormBuilderFormAttributes]    Script Date: 12/19/2017 11:20:19 PM
******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[FormBuilderFormAttributes](
	[FormBuilderFormAttributeID] [int] IDENTITY(1,1) NOT NULL,
	[FormBuilderFormID] [int] NOT NULL,
	[FormBuilderAttributeID] [int] NOT NULL,
	[Name] [nvarchar](max) NULL,
	[Code] [nvarchar](max) NULL,
	[Description] [nvarchar](max) NULL,
	[IsDefault] [bit] NOT NULL,
	[IsRequired] [bit] NOT NULL,
	[IsVariable] [bit] NOT NULL,
	[FieldOrder] [tinyint] NOT NULL,
	[IsDefaultEditableExceptOrder] [bit] NOT NULL,
	[IsCaseClasification] [bit] NOT NULL,
	[CanCutomerEdit] [bit] NOT NULL,
	[CanCaseWorkerEdit] [bit] NOT NULL,
	[DisplayTo] [nvarchar](max) NULL,
	[DependsOn] [nvarchar](max) NULL,
	[DependsWhen] [nvarchar](max) NULL,
	[Validation] [nvarchar](max) NULL,
	[RegEx] [nvarchar](max) NULL,
	[DataTypeDescription] [nvarchar](max) NULL,
	[AgentOnlyFormAttibute] [bit] NOT NULL,
	[Group] [nvarchar](max) NULL,
	[IsDeleted] [bit] NOT NULL,
	[CreatedBy] [bigint] NOT NULL,
	[CreatedOn] [datetime] NULL,
	[UpdatedBy] [bigint] NULL,
	[UpdatedOn] [datetime] NULL,
 CONSTRAINT [PK_dbo.FormBuilderFormAttributes] PRIMARY KEY CLUSTERED
(
	[FormBuilderFormAttributeID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

GO
/****** Object:  Table [dbo].[FormBuilderForms]    Script Date: 12/19/2017 11:20:19 PM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[FormBuilderForms](
	[FormBuilderFormID] [int] IDENTITY(1,1) NOT NULL,
	[Name] [nvarchar](max) NULL,
	[Version] [nvarchar](max) NULL,
	[IsActive] [bit] NOT NULL,
	[Code] [nvarchar](max) NULL,
	[Status] [tinyint] NULL,
	[OrganizationID] [int] NOT NULL,
	[IsDeleted] [bit] NOT NULL,
	[CreatedBy] [bigint] NOT NULL,
	[CreatedOn] [datetime] NULL,
	[UpdatedBy] [bigint] NULL,
	[UpdatedOn] [datetime] NULL,
 CONSTRAINT [PK_dbo.FormBuilderForms] PRIMARY KEY CLUSTERED
(
	[FormBuilderFormID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

GO
ALTER TABLE [dbo].[FormBuilderAttributes]  WITH CHECK ADD  CONSTRAINT
[FK_dbo.FormBuilderAttributes_dbo.FormBuilderDataTypes_FormBuilderDataTypeID] FOREIGN
KEY([FormBuilderDataTypeID])
REFERENCES [dbo].[FormBuilderDataTypes] ([FormBuilderDataTypeID])
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[FormBuilderAttributes] CHECK CONSTRAINT
[FK_dbo.FormBuilderAttributes_dbo.FormBuilderDataTypes_FormBuilderDataTypeID]
GO
ALTER TABLE [dbo].[FormBuilderFormAttributes]  WITH CHECK ADD  CONSTRAINT
[FK_dbo.FormBuilderFormAttributes_dbo.FormBuilderAttributes_FormBuilderAttributeID] FOREIGN
KEY([FormBuilderAttributeID])
REFERENCES [dbo].[FormBuilderAttributes] ([FormBuilderAttributeID])
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[FormBuilderFormAttributes] CHECK CONSTRAINT
[FK_dbo.FormBuilderFormAttributes_dbo.FormBuilderAttributes_FormBuilderAttributeID]
GO
ALTER TABLE [dbo].[FormBuilderFormAttributes]  WITH CHECK ADD  CONSTRAINT
[FK_dbo.FormBuilderFormAttributes_dbo.FormBuilderForms_FormBuilderFormID] FOREIGN
KEY([FormBuilderFormID])
REFERENCES [dbo].[FormBuilderForms] ([FormBuilderFormID])
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[FormBuilderFormAttributes] CHECK CONSTRAINT
[FK_dbo.FormBuilderFormAttributes_dbo.FormBuilderForms_FormBuilderFormID]
GO
```