2019-02

A Web-based application for recommendation of open source software for higher learning institutions in Tanzania

Okey, Ambokile

NM-AIST

https://doi.org/10.58694/20.500.12479/254

A WEB-BASED APPLICATION FOR RECOMMENDATION OF OPEN SOURCE SOFTWARE FOR HIGHER LEARNING INSTITUTIONS IN TANZANIA

Ambokile Okey

A Dissertation Submitted in Partial Fulfilment of the Requirements for the Degree of Master's Degree in Information and Communication Science and Engineering of the Nelson Mandela African Institution of Science and Technology

Arusha, Tanzania

February, 2019

ABSTRACT

Higher Learning Institutions (HLIs) perform a vital role in developing students to professionals, and in contributing to social and industrial development through research outputs and delivering professional support to different organizations. The growth of Information and Communication Technologies (ICTs) has led to the high use of software tools for supporting the roles of HLIs to deliver high-quality research and education. However, most of the Proprietary Software (PS) tools are expensive, hence hindering HLIs in Tanzania from using them for their core activities. Open Source Software (OSS) provide advantages for HLIs because they remove the cost of acquiring licenses and prevent software illegality issues. However, in Tanzania, most of HLIs do not use OSS due to lack of awareness.

This study aimed at improving the adoption of OSS in Tanzanian HLIs through enhancement of individual awareness on the existing OSS designed for academic purposes using a recommender application. The application uses three recommendation approaches: content-based, demographic and collaborative filtering. Questionnaires were used for identifying currently used software in different academic areas of specializations and gathering requirements for development of a web-based application. Further, it explored useful OSS designed for academic purposes from online platforms and categorized them according to their academic area of use. The list of categorized OSS were uploaded to the application's database and used as a foundation for recommendations. The developed application passed a test against users' requirements as it was able to: recommend OSS, send notification through user's email, allow view and download, and accept users' feedbacks.

DECLARATION

I, Ambokile Okey do hereby declare to the Senate of The Nelson Mandela African Institution of Science and Technology that this dissertation is my own original work and that it has neither been submitted nor being concurrently submitted for degree award in any other institution.

Ambokile Okey A. Okey

Name and Signature of Candidate

6th March 2019

Date

The above declaration is confirmed

Dr. Anael E. Sam

Name and Signature of Supervisor

6th March, 2019

Date

COPYRIGHT

DECLARATION

I, Ambokile Okey do hereby declare to the Senate of The Nelson Mandela African Institution of Science and Technology that this dissertation is my own original work and that it has neither been submitted nor being concurrently submitted for degree award in any other institution.

Ambokile Okey A. Okey

Name and Signature of Candidate

6th March 2019

Date

The above declaration is confirmed

Dr. Anael E. Sam

Name and Signature of Supervisor

6th March, 2019

Date

ACKNOWLEDGMENTS

First and foremost, I would like to thank God the almighty for giving me the gift of life and good health during my master's studies at the Nelson Mandela African Institution of Science and Technology (NM-AIST).

I would like to thank the German Academic Exchange Service (DAAD) for funding my master's studies.

I would also like to thank Dar es Salaam University College of Education (DUCE) for funding my research work.

Special Thanks to my supervisor Dr. Anael E. Sam of NM-AIST for support and assistance throughout the research work.

I would like to extend my thanks to my family members for their support during my studies.

I acknowledge the support of my classmates during our master's studies, thank you very much.

Lastly but not least I would like to thank the entire NM-AIST communities for their support during my master's studies and stay here in Arusha.

DEDICATION

My mother Queen Moses Sakamala, to you my beloved late Mother I dedicate this dissertation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

ABBREVIATIONS

ARU	Ardhi University
ARV	Average Rating Value
BIT	Bachelor of Information Technology
CSS	Cascading Style Sheet
DFD	Data Flow Diagram
DIT	Dar es Salaam Institute of Technology
DUCE	Dar es Salaam University College of Education
FOSS	Free and Open Source Software
GUI	Graphical User Interface
HLIs	Higher Learning Institutions
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transfer Protocol
ICTs	Information and Communication Technologies
IFM	Institute of Finance Management
IT	Information technology
MoEST	Ministry of Education Science and Technology
MU	Mzumbe University
NM-AIST	The Nelson Mandela African Institution of Science and Technology
OSP	Open Source Project
OSS	Open Source Software
PHP	Hypertext Preprocessor
PS	Proprietary Software

SDLC	Software Development Life Cycle
SPSS	Statistical Package for the Social Sciences
SQL	Structured Query Language
TCRA	Tanzania Communications Regulatory Authority
UML	Unified Modelling Language
UDOM	University of Dodoma
UDSM	University of Dar es Salaam

CHAPTER ONE

INTRODUCTION

1.1 Background Information

This chapter presents the general introduction of the research study, covering the following sections: background information, research problem statement, general and specific objectives, research questions, the significance of the study and the dissertation outline.

Higher Learning Institutions (HLIs) perform a vital role in developing students to professionals, and in contributing to social and industrial development through research outcomes and delivering professional support to different organizations. These roles consequent the core activities of many HLIs which includes teaching, research, and consultancy. Scientific and non- scientific methodologies support HLIs in achieving their goals. Technological growth has led to the expansion of supporting tools on the roles of HLIs. Many software tools are available to support HLIs in executing their daily activities, but there are challenges which constrain HLIs on acquiring and implementing the software in the working environment. These constraints include lack of supportive infrastructure for software implementation (Information and Communication Technologies (ICTs), hardware, personnel etc.), low knowledge on software available and implementation cost (Sife *et al.*, 2007). The use of software tools whether open source, proprietary or simulator play important role in today's educational and research activities to enhance the teaching and learning (Al-Hajri *et al.*, 2017).

In HLIs, courses in some specializations are designed with a considerable percentage of practical and training works, which are carried out using Proprietary Software (PS) or not at all. The use of PS leads to illegality issues because most of HLIs used cracked PS version which is the owner's license infringement.

Open Source Software is a concept and practice of making the program source code freely available to everyone to use it (Al-Hajri *et al.,* 2017). It removes the cost of purchasing software while providing the core functionalities that are available in PS for instance Calibre is an OSS and Alfa eBooks manager is a PS, both are for E-book Management. On the other hand, PS is not freely available, and users must, therefore, purchase in order to use the software.

The reasons for adopting OSS are total cost ownership, free to make copies and distribution, software legality, reliability, availability, performance, security and other pedagogical and administrative benefits (Al-Hajri *et al.*, 2017).

1.2 Research Problem

The growth of ICTs has led to the expansion of software tools that support HLIs in fulfilling their roles. Most of the tools are expensive and HLIs in Tanzania face challenges like budget constraints and illegal use of licenses, which hinder them from using software for research, teaching, consultancy, and learning. Open Source Software are advantageous to HLIs because they remove these licensing costs and software illegality issues due to being freely available. However, in Tanzania, most of HLIs do not use OSS in order to facilitate research, teaching, consultancy, and learning due to knowledge gap between HLIs communities and available OSS designed for academic purposes (Haule, 2015). Many different choices of OSS online but users require knowledge on the various vendors or online platforms that host those OSS. Therefore a web-based application that recommends OSS designed for academic purposes is required for enhancing HLIs' awareness on them.

1.3 Justification of the Study

In recent years, the adoption of OSS has increased rapidly. Countries like Russia make the use of OSS mandatory in its entire sectors whereas others are in the process of adopting OSS (Al-Hajri *et al.*, 2017). Usage of software tools and simulators play an important role in today's teaching and learning activities (Al-Hajri *et al.*, 2017). Recently data from the Ministry of Education Science and Technology (MoEST) in Tanzania shows that there are 106 HLIs, but only a few of them are using OSS for E - Learning e.g. Moodle (MoEST, 2015; Sife *et al.*, 2007).

Open Source Software are already available. Software development companies and individual programmers are struggling to develop and improve OSS so as to support different organizations, but many HLIs do not use OSS despite the advantage of removing the cost of purchasing a license but still facilitate teaching and learning. For instance E - Learning software like Moodle. It is argued that "universities in developing countries should adopt OSS to improve teaching and learning processes" (Sife *et al.*, 2007).

Information about appropriate OSS available to support particular learning or teaching activity needs to be easily accessed in a well-organized manner, this requires a special purpose platform that can organize well relevant information for its users.

This research study aims at developing a web-based application for the recommendation of OSS for Tanzanian HLIs so that users can be recommended with appropriate OSS for research, teaching, consultancy and learning in their academic areas of specialization.

1.4 Research Objective

1.4.1 Main Objective

The main objective of the research is to develop a web-based application for recommendation of OSS for HLIs in Tanzania.

1.4.2 Specific Objectives

(i) To identify and analyze the requirements of HLIs in Tanzania on accessing appropriate OSS research, teaching, consultancy, and learning.

(ii) To develop a web-based application for recommendation of OSS for HLIs in Tanzania.

(iii) To validate the developed web-based application.

1.5 Research Questions

(i) What are the requirements of HLIs in Tanzania on accessing appropriate OSS for research, teaching, consultancy, and learning?

(ii) What are the functional and non-functional requirements for the web-based application?

(iii) Did the developed web-based application meet the HLIs requirements?

1.6 The Significance of the Study

(i) To enhance awareness of academicians, students, lab scientists and system administrators in HLIs on existing OSS designed for academic purposes using the three recommendation algorithms: content-based, demographic and collaborative filtering.

(ii) To improve usage of OSS in HLIs by enabling academicians, students, lab scientists, and system administrators to get detailed information of OSS, and download various OSS designed for academic purposes.

(iii) To improve the quality of pedagogical activities through the use of OSS tools designed for the academic purposes recommended by the web application.

(iv) To reduce illegal use of PS by recommending OSS alternatives to academicians, students, lab scientists, and systems administrators.

(v) To pave the way for further researches on the use of recommender system in other setups other than e-commerce.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

This chapter presents the literature review. In this chapter OSS in educational context, challenges for adopting E - learning technologies, recommender systems concepts and E - Commerce recommender systems are discussed.

2.2 Open Source Software in the Context

An OSS is computer software with its source code made available with a license agreement that allows the copyright holder to provide the rights to study, change, and distribute the software (Al-Badi and Al-Badi, 2007). Many countries worldwide insist the adoption of OSS in their government sectors because of the free cost of ownership while providing a high level of security, good performance, and reliable service. For instance, Russia makes the use of OSS mandatory in all sectors meanwhile other countries like Tanzania are in the process of adoption (Al-Hajri *et al*., 2017). Open Source Software offer several benefits to HLIs as they can be customized to suit the organizational needs. Thus, lecturers and students are no longer forced to use illegal copies of software tools. Eventually, using OSS can contribute to a reduction in running costs. These core benefits are whys for the need and adaption of OSS (Al-Badi and Al-Badi, 2007). Open Source Software in HLIs lies into various categories but not limited to: learning management tools, content management tools, and collaboration tools. Various HLIs Worldwide have adopted OSS for various applications such as in course management and electronic portfolios due to several advantages OSS have over PS (Lungo and Kaasbol, 2007).

Open Source Software perform a vital role in the success of many HLIs in the world, but lack of awareness on existing OSS designed for academic purposes is a challenge to many HLIs in the adoption process. For instance, many people in Oman Sultan Qaboos University are not aware of OSS, leading to its low usage (Al-Badi and Al-Badi, 2007).

Some of OSS provide e - learning platforms, the Learning Management System (LMS) such as Moodle. A centralized set of tools is provided to HLIs by LMS to help them succeed in various aspects of curricula such as managing course catalogues, record data from learners, and providing reports to management (Fariha and Zuriyati, 2014). In today’s education various

OSS play a great role in assisting people in HLIs to carry out their tasks, to mention few: Latex for authoring research papers, python for data analysis, android studio for mobile application development, and QGIS for creating, editing, analyzing and publishing geographical information.

2.3 E - Learning Technologies Adoption Challenges in HLIs in Tanzania

E - Learning refers to the use of ICTs to enhance and support teaching, learning, and research. This has transformed a traditional way of doing pedagogical activities. E - learning technologies can be in hardware form (Televisions, Digital Versatile Disks (DVDs), Radio) or in software form (E - learning platforms, web - based technologies, software applications) (Sife *et al*., 2007). Software e - learning technologies can be proprietary or open source. Albeit the significance of E - learning in education and research its adoption in Tanzanian HLIs though at an increasing rate is apparently low (John and Sam, 2018; Lungo and Kaasbol, 2007).

Lack of enough financial resource to budget for e-learning tools is a challenge to many HLIs in Tanzania, this is due to the fact that the Government of Tanzania has reduced public funds (Lwoga, 2012).

Open Source Software tools such as Moodle require Internet infrastructure to operate but other OSS tools such as Latex do not require the Internet to operate. There are various challenges faced in utilizing the OSS because as OSS differ in characteristics, so the challenges. Few of OSS tools require the organizational support of ICT infrastructures whilst many do not. However, the ICT infrastructures support is no longer a challenge due to the high rate of growth of Internet connectivity in Tanzania where most of the telecommunication companies offer up to 4G service to their subscribers (TCRA, 2018). Several OSS tools addressed in this study do not require the Internet for them to operate, rather they solely require it during downloading and updating.

Despite the benefits provided by OSS tools, and ICT infrastructure provided by the government of Tanzania, the implementation of OSS tools in HLIs is still low due to a number challenges: lack of systemic approach to ICT implementation, awareness and attitude towards ICTs, administrative support, technical support, staff development, lack of ownership, and inadequate funds (Sife *et al*., 2007). Nevertheless, most of these challenges are not common to all OSS tools, they are most relevant to e-learning platforms as they require the involvement of the entire organization.

Generally, the most commonly observed challenge that slows down the adoption process is the knowledge gap between students, academics, system administrators, and lab scientists over the existing e-learning technologies (Haule, 2015).

2.4 Recommender Systems Concept

Computer algorithm refers to a precise method usable by a computer for the solution of a problem and, it is composed of a finite set of steps, each of which may require one or more operation (Horowitz and Sahni, 1978). It must satisfy the following criteria, it must be clear what is to be done. Second, the operations must be done within a finite amount of time. Third, outputs meaning that it must produce one or more outputs and may have zero or more inputs which are externally supplied. Fourth, it must terminate after a finite number of operations. In this study, computer algorithms were used for recommending OSS for HLIs.

A recommender system is a software tool and technique providing suggestions for items that may be of use to the user. Users during their online session can encounter information overload. The recommender systems enrich their decisions on available information. Recommender systems are designed to help individuals who lack sufficient personal experience to evaluate a large number of items that an online platform may provide (Ayushi and Prasad, 2018).

There are different classes of recommendation approaches:

(i) **Content-based:** The system learns to recommend items that are similar to ones that user liked in the past. The similarity of items is calculated based on the features associated with compared items.

(ii) **Collaborative filtering**: The system recommends to the active user the items that other users with similar taste liked in the past. The similarity in the taste of two users is calculated based on the similarity in the rating history of the users.

(iii) **Demographic:** The system recommends items to a user based on their demographic profiles such as age, education, gender etc.

(iv) **Knowledge-based:** The system recommends items based on specific domain knowledge about certain item features meet user needs and how that item can be useful for the user.

In this system, a similarity function estimates how much the user needs to match the recommendations.

(v) **Community-based:** The system recommends items based on the preferences of users friends.

(vi) **Hybrid recommender system:** This system is based on a combination of different techniques mentioned above. This system tries to use the advantages of one technique to fix the disadvantage of another technique.

2.5 E - Commerce Recommender Systems

Recommender systems have been applied in E - commerce for many years because they can help business companies in increasing their sales. They allow the E - commerce website to display advertisements or offers based on consumer behaviour (Schafer *et al.*, 2001). Recommender systems enhance E-commerce sales in three ways:

(i) Visitors to an E-commerce website can browse the website without purchasing anything. Recommender systems can help them to find items they want (Schafer *et al.*, 2001).

(ii) Recommender systems can improve cross-sell by suggesting additional items to the consumers to purchase (Schafer *et al.*, 2001).

(iii) Recommender systems can help build loyalty to consumers because if the website finds the consumers their best match, the consumers are more likely to return (Schafer *et al.*, 2001).

2.5.1 Amazon.com

Amazon.com uses the following features in its book section:

(i) **Customers who bought:** this feature uses two lists. The first recommends books that are often purchased by other customers. The second recommends authors whose books are often purchased (Schafer *et al.*, 2001).

(ii) **Your Recommendations:** this feature uses direct feedbacks from other customers. Customers rate books they have read on a scale of 5 points (Schafer *et al.*, 2001).

(iii) **Eyes:** this feature uses customers request based upon author, title, subject, ISBN, or publication date information to notify customers via email of new items that have been added to Amazon.com catalogue (Schafer *et al.*, 2001).

(iv) **Delivers:** this feature uses the checklist of specific categories (Oprah books, biographies, cooking) to send the latest recommendation by email to subscribers in the category (Schafer *et al*., 2001).

(v) **Bookstore Gift Ideas:** this feature allows customers to receive recommendations from editors. Customers pick a category of books for which they would like some suggestions (Schafer *et al*., 2001).

(vi) **Customer Comments:** this feature allows customers to receive text recommendations based on the opinions of other customers (Schafer *et al.*, 2001).

(vii) **Purchase Circles:** this feature allows customers to view the top ten list for a given geographic region, company, educational institution, government or other organization. Customers can view Purchase Circles by navigating to the Circle that interests them (Schafer *et al*., 2001).

2.5.2 CDNOW

CDNOW uses the following features on its website:

(i) **Album Advisor:** this feature works in three different modes. The first two are similar to the Customers Who Bought feature of Amazon.com. The third mode works as a gift advisor. Customer type in the names of up to three artists and the system returns a list of ten albums CDNOW considers similar to the artists (Schafer *et al*., 2001).

(ii) **Related Artists:** this feature works on the assumption that if a customer likes a certain performer, there is a group of artists with similar styles that she will also like. Thus once a customer locates an artist and selects the Related Artists link, will receive the recommendation of artists who are similar to the located (Schafer *et al*., 2001).

(iii) **Buyer's Guides:** this feature allows customers to receive recommendations based on a particular genre of music (Schafer *et al*., 2001).

(iv) **Top 100:** this feature allows customers to receive the recommendation of the top 100 that are drawn from the sales figures of the site and can theoretically be continuously upgraded to reflect actual sales (Schafer *et al*., 2001).

2.5.3 Drugstore.com

Drugstore.com uses the following feature on its website:

(i) **Advisor:** this feature allows customers to indicate their preferences when purchasing a product from a category (Schafer *et al.*, 2001).

(ii) Test Drives: this feature uses a team of volunteers. A new product is sent for them to review (Schafer *et al.*, 2001).

2.5.4 eBay

eBay uses the following feature on its website:

(i) **Feedback Profile:** this feature allows both buyers and sellers to give feedback to each other as long as they have ever done business. The feedback consists of a satisfaction rating as well as a specific comment about the other customer. Feedback is used to provide a recommender system for purchasers, who are able to view the profile of sellers (Schafer *et al.*, 2001).

(ii) **Personal Shopper:** this feature allows customers to indicate items they would like to purchase (Schafer *et al.*, 2001).

2.5.5 MovieFinder.com

MovieFinder.com uses the following features on its website.

(i) **Users Grade/Our Grade**: this feature allows the customers and editors to rate a movie using letter grade (A-F). Both the Users Grade and the Our Grade features report a letter grade recommendation to the customer (Schafer *et al.*, 2001).

(ii) **Top 10:** this feature allows the customers to get recommendations from the editors in a category of their choice (Schafer *et al.*, 2001).

2.6 Other Related Works

2.6.1 Alternative Educational Framework

Alternative education framework which is referred to as a research methodology to identify Free and Open Source Software (FOSS); colleges of applied sciences in different regions of the sultanate in Oman used this framework. It is aimed at identifying a list of FOSS to replace PS used in courses of a Bachelor degree in Information Technology (BIT) (Al-Hajri *et al.*, 2017). This framework presented a list of FOSS for use in BIT, but the list was in a paper-based which may face challenges like storage and ease of access and it was for a single educational area of specialization. In consequence, it needed improvement so that it can be accessed by many people without any restriction, and also include more academic areas of specialization.

2.6.2 Sourceforge Website

Sourceforge is a website for publishing Open Source Projects (OSPs). It allows the community to share OSPs by contributing to its improvement. Despite having many projects, Sourceforge treats educational software as a single category, thus there are no recommendations of OSS based on its usefulness to particular pedagogical activity or as an alternative to certain PS to stakeholders in HLIs (Slashdot Media, 2018).

2.6.3 Alternativeto Website

Alternativeto.net is a website that provides alternative recommendations to software. It does not allow direct download rather provide a link to an official website of the software. Also like source forge, it treats educational software as a large category as a consequence there are no recommendations of OSS based on its usefulness to particular pedagogical activity or stakeholders in HLIs (Ola and Markus, 2018).

In Tanzania, researchers presented the importance of adopting OSS tools to improve learning and teaching processes (Haule, 2015; Sife *et al.,* 2007). The existence of a knowledge gap in selecting suitable OSS tools was clearly pointed out. Thus, so far a web-based application that provides recommendations of OSS designed for the academic purpose to HLIs does not exist.

CHAPTER THREE

MATERIALS AND METHODS

3.1 Introduction

This chapter presents materials and methods used for exploring the current situation of OSS awareness and usage in HLIs, developing the solution and validation of the developed solution.

The core parts of this research were eliciting user requirements for the web application and assessing usability. Therefore qualitative research methods were used because they capture opinions and thoughts (Singh, 2006). Requirement elicitation is an important aspect for system development as it establishes what services system should provide for its users, it provides a blueprint for a developer to design how a system is going to deliver required services (Sommerville, 2011).

3.2 The Case Study Area

The study was conducted in HLIs that are located in Arusha, Dar es Salaam, Morogoro, and Dodoma regions of Tanzania. Requirements for OSS that fall in a group of business, sciences, Information Technology (IT) and Engineering academic areas were assessed at the following institutions: Institute of Finance Management (IFM), Ardhi University (ARU), University of Dodoma (UDOM), University of Dar es Salam (UDSM), Dar es Salaam Institute of Science and Technology (DIT), and the Nelson Mandela African Institution of Science and Technology (NM-AIST). While requirements for OSS that fall in a group of education management and social sciences academic areas were assessed at the following institutions: Mzumbe University (MU) and Dar es Salaam University College of Education (DUCE).

3.3 Targeted Population

The study involved a sample of students, systems administrators, and academicians. The minimum level of education for students was certificate level because most of HLIs in Tanzania start to offer higher education from this level, and the maximum level of education was doctorate level because this is the highest higher education level in most HLIs in Tanzania. Academicians from tutorial assistants to professors because they possess different experience with OSS tools available in their fields of specialization.

3.4 Sample Size and Sampling Technique

In conducting this research study, a total of one hundred seventy-four (54 males, 120 females) students, academicians, and system administrators took part in this study. Purposive sampling techniques were employed because respondents from a certain academic area of specialization in one HLI can represent other respondents in the same academic area of specialization in other HLIs.

3.5 Data Collection Methods

The methods used for data collection were questionnaires with both open-ended and close-ended questions, and structured interviews. This study used three different questionnaires. The first questionnaire was for students and academicians, it had twenty-six questions that were organized in sections as follows: general information, end users software applications usage, awareness of OSS, usage of OSS, and attitude on OSS. The second questionnaire was for system administrators and lab scientist, it had twelve questions that were organized in sections as follows: general information and software application usage. The third questionnaire was for assessing the usability of the developed web application, it had six questions. The objective of using these methods was gaining an understanding of the current situation in HLIs regarding awareness and usage of OSS, gathering the requirement for the development of the solution that this research proposes and evaluate the developed web-based application.

3.6 Data Analysis and Processing

In the process pertaining to data analysis, the Python programming language was used to analyze the collected data. Frequencies were used to represent different responses so as to compare them. Pie charts were used to provide a graphical representation of the analysed data in different cases. Inductive reasoning was used to gain the requirement from the collected data. Uses cases and prototypes were used to validate the requirements.

3.7 Web Technologies used in this Study

Web technologies are the collection of technologies that were developed as part of the birth of the world wide web (Jackson, 2007). They are the foundation of the development of web applications, in this section web technologies that were used for the development will be discussed.

(i) Hypertext Preprocessor (PHP) is an open-source server-side scripting language. It is used to generate dynamic web pages (Darie and Balanescu, 2008). It can be used together with different web technologies such as Hypertext Markup Language (HTML) and Structured Query Language (SQL). In this study, PHP is used to: accept data from the client and send them to the database server for storing, implement different functions that are executed on the server and ensure secure login using a password.

(ii) Cascading style sheet (CSS) is used for formatting the web pages, it deals with the physical and visual display of the document (Nixon, 2012). It can be used as an inline style sheet, inline style attribute or from an external file. Using files is most advantageous as a web developer will need to change only one document to control all web pages. Cascading Style Sheet is used to format the physical structure of the HTML pages.

(iii) Hypertext Markup Langauge is the main markup language for displaying web pages and other information that can be displayed in the web browser (Nixon, 2012). It is used to generate static contents of the web pages.

(iv) JavaScript is the programming language used in HTML pages. It can dynamically modify an HTML page to react with user input or validate user input (Odell, 2009). JavaScript interaction with the user does not require communication with the server thus it reduces a significant amount of response time the user would send a request to the server.

(v) JQuery is a fast, small, and feature-rich JavaScript library which makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy to use API that works across a multitude of browsers (Jquery, 2018).

(vi) My Structured Query Language (MySQL) is an open source relational database management system whose purposes is to store, manage and retrieve data (Darie and Balanescu, 2008). It can be used together with other web technologies such as Apache Server and PHP. In this study, MySQL is used as the back-end database, because in this study data are stored as the relationship between entities, each entity was represented as a table. My Structured Query Language was selected for implementation because it is open source, it is multi-threaded which enables it to be accessed by multiple users, it can be accessed through the Internet, and it provides support for query language such as SQL which is the standard language for querying the database.

(vii) Structured query language (SQL) is the standard language for storing, manipulating and retrieving data in databases (Darie and Balanescu, 2008). Structured Query Language is used to communicate with relational databases. In this study, SQL was used to communicate with MySQL for the purpose of storing, managing and retrieving data.

These web technologies used in this study were selected because they are open source, platform independent and widely used for web development. Further, these web technologies can be easily integrated with each other. For instance PHP with SQL, and JavaScript with HTML.

3.8 Server Configurations

The following are different software tools used for setting up the development environment as well as procedures specific configurations made in the server.

(i) WampServer is a window web development environment. It allows you to create web applications with Apache2, PHP, and a MySQL database. Apache, MySQL, and PHP can be installed individually and most of the times for the purpose of web development are installed together. WampServer provides a user-friendly way of installing and configure Apache, MySQL and PHP components on Windows. In this study, WampServer 3.0.6 was used as a development environment with Apache version 2.4.23, MySQL version 5.7.14 and PHP version 5.6.25 installed. The following were the procedures for setting up the WampServer.

(ii) phpMyAdmin is a tool written in PHP intended to handle the administration of MySQL over the Web (Ibennetch, 2018). In this study, PhpMyAdmin was installed together with the WampServer and it was used to manage database, tables, relations, indexes, users, and columns.

(iii) Default configurations of the Wamp server do not allow large files to be uploaded. Thus PHP was configured so as to allow large files to be uploaded in the server. Those files were the setup files of the OSS. The following were the changes made in the php.ini file found in (C:\wamp64\bin\php\php5.6.25) to allow files up to 3000MB: post_max_size = 3000M and upload_max_filesize = 3000M.

(iv) Apache HyperText Transfer Protocol (HTTP) server is an open source web server developed by the Apache software foundation open source community (Apache Software Foundation, 1995). In this study, the Apache HTTP server was used to store web contents and accepting user request and send a response to those request by executing PHP scripts. Default Apache configurations do not allow the application to be available over the Internet. Thus Apache was configured so as to make the developed web-based application available over the Internet. The following changes were made in the httpd.conf file and httpd-vhosts.conf which is found in (C:\wamp64\bin\apache\apache2.4.23\conf) and (C:\wamp64\bin\apache\apache2.4.23\conf\extra) respectively.

(v) The server was configured to listen to any IP address by changing httpd.conf because the server did not have static IP address during development and testing, also the server was configured to listen to port 80 and 8080.

a) Listen 0.0.0.0:80

b) Listen [: 0]:80

c) Listen [: 0]:8080

(vi) The server was configured to be accessible by any IP address using the following configurations in the httpd-vhosts.conf file. This was done by replacing the line “Require local” with “Require all granted”.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Introduction

This chapter presents the results obtained from the analysis of the primary data, requirements, design, development, verification and validation of the web-based application. Further, the chapter presents a discussion of the developed application.

4.2 Data Presentation

This study involved data collection in HLIs in Tanzania according to the first objective. The primary data collected were analyzed using the Python programming language and the results were presented as detailed in chapter subsections below.

4.2.1 Respondents' Demographic Data

In this study, out of 174 respondents, 120 respondents (55%) were males and 54 respondents (45%) were females. The minimum age group of the respondents was 16 to 22 years, this is the lowest age group for students in Tanzania to start higher education. Majority of the respondent 102 (58.6%) fall between the age group of 23 to 29 years in which majority were students studying first degree and the next group has 25 respondents (14.4%) which also is comprised by students studying first degree and the third age group has 23 respondents (13.2%) in which majority were academician at the level of assistant lecturer. The fourth age group has 18 respondents (10.3%) in which majority were academician at the level of lecturer and the last age group was 44 years or above which has 6 respondents (3.4%). Table 1 and Table 2 summarizes respondents' demographic data.

Table 1: Gender and Age of Respondents

Demographic variable		**Respondents**	**Percentage (%)**
Gender	Male	54	31.0
	Female	120	69.0
Age groups(Years)	16 to 22	25	14.4
	23 to 29	102	58.6
	30 to 36	23	13.2
	37 to 43	18	10.3
	44 or above	6	3.4

Table 2: Age and type of Respondents

Respondents	**Age (years)**					Percentage (%)
	16 to 22	23 to 29	30 to 36	37 to 43	44 or above	
Academicians		**7**	**12**	**11**	**4**	**19.5**
Lecturer			1	7	1	
Assistant Lecturer		1	8	4	2	
Tutorial Assistant		6	3		1	
Students	**25**	**94**	**9**	**6**	**1**	**77.6**
Doctorate			1	2		
Advanced/Higher Diploma		1				
Bachelor	19	83	3	2		
Masters		8	5	2	1	
Ordinary Diploma	6	2				
System administrator		**1**	**2**	**1**	**1**	**2.9**
System Administrator		1	2	1	1	
Total	25	102	23	18	6	**174**

4.2.2 Software Categories Usage by Respondents for Academic Purposes

From this section, the results will be referring to 169 respondents. These respondents were involved in the data analyzed from this section. Majority of the respondents 56.8% were using PS for academic purposes and only few 4.1% respondents who claimed that they do not use any software. Although the majority of the respondents are using PS, the majority of those who use software 67.7% obtain illegal copies of PS and only 6.1% obtain a copy of PS legally. More information about software categories usage by respondents and how they obtain copies of proprietary are shown in Fig. 1 and Fig. 2.

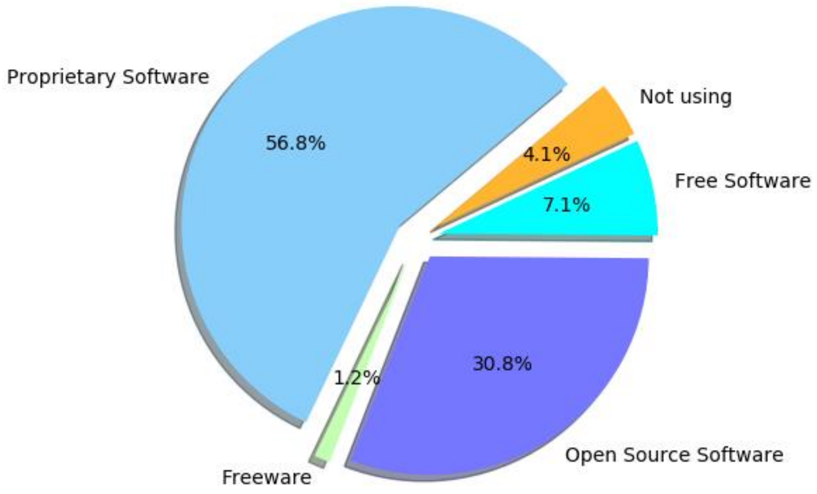


Figure 1: Software Categories Usage.

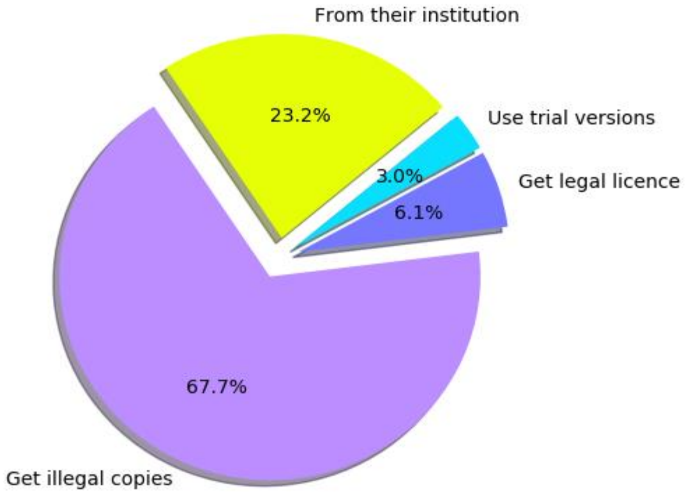


Figure 2: Methods used to obtain PS.

4.2.3 Respondents Awareness on OSS

Out of 169 respondents 74.6% are not aware of OSS that are designed for the academic purpose; also there are those who did not even know there are software that are available with a free license, it's only 25.4% that are aware of OSS designed for academic purposes. More details on awareness of respondents are shown in Fig. 3.

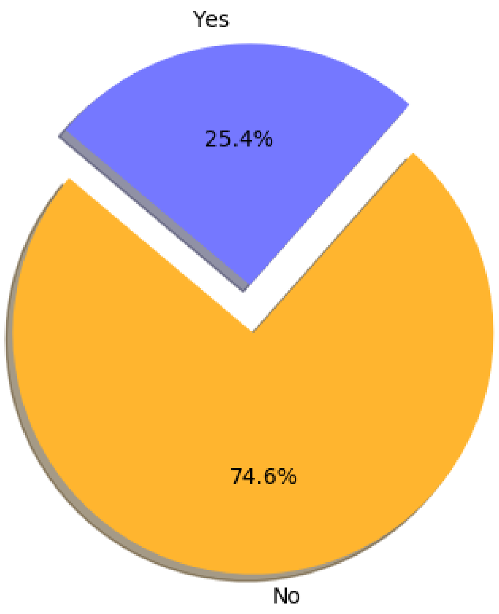


Figure 3: Respondents' Awareness of OSS

4.2.4 Factors that Contribute to Respondents' Unawareness on OSS

Out of 126 respondents who claimed that they are not aware of OSS designed for academic purposes, 38.1% of the respondents' claim that they do not know if there OSS designed for academic purposes while only 0.8% claim that mostly contain the virus and thus they are not interested at all. Table 3 summarizes factors that contribute to respondents' unawareness on OSS.

Table 3: Factors that Contribute to Respondents' Unawareness on OSS

Reason for unawareness	Respondents	Percentage (%)
I do not know if there is OSS designed for academic purposes.	48	38.1
I do not know what OSS is.	15	11.9
I do not know which online sites for OSS to visit.	45	35.7
They are not well promoted.	17	13.5
They mostly contain the virus and thus am not interested in them at all	1	0.8

4.2.5 Methods used to obtain OSS and their Challenges

Out of 43 respondents who use OSS 62.8% obtain their copies from a website while 37.2% obtain from colleagues or friends. Both of the respondents experience different challenges with the method used to obtain copies of OSS. Table 4 summarizes the challenges stated by respondents that use each of the mentioned methods.

Table 4: Methods used to obtain Software and their Challenges

Method and its challenges	Respondents	Percentage (%)
From a colleague or friend	16	37.2
Denial of some OSS from them, time consuming looking for them, sometimes they need money in return.	2	
No OSS updates, time-consuming to find a friend with the OSS.	7	
Not reliable as it depends on willingness and availability of a colleague random search is not always successfully.	1	
Some copies of OSS are corrupted	2	
Some OSS are not found from a friend, time consuming	1	
Failure to access OSS on time.	1	
An improper version of OSS from a friend	1	
It is not easy to find OSS to fit the purpose	1	
From a website	27	62.8
Redirect to many other pages	1	
Missing some features	1	
Most of OSS websites I know are for general purposes and are not specifically for educational purposes	4	
No options available to get updates	1	
Not easy to get OSS in time, a lot of limitations to download a software	16	
Sometimes they require money to be paid to download the software	1	
It consumes time on finding the OSS, no categorization of OSS according to their applicability	1	
Unreliable sites most of the times they do get shut down.	1	
You need to know the name, else a flood of results, difficult to choose one	1	

4.2.6 Respondents Perspective on the need for a Web-Based Application for Recommendation of OSS

Out of 169 respondents 53.8% strongly agreed that it will be helpful if there will be a web-based application for the recommendation of OSS for HLIs in Tanzania, only 0.6% strongly disagreed. Figure 4 below summaries the perspective of respondents on the need for a web-based application for the recommendation of OSS for HLIs. Table 5 presents the requirements provided by the users.

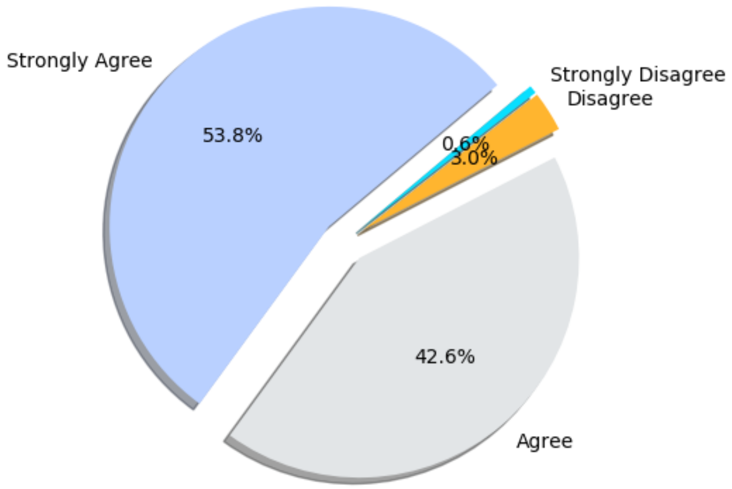


Figure 4: Respondents' Perspective on the need for a Web-based Application for Recommendation of OSS

4.3 Requirements for the Web-Based Application

In this study, requirements were gathered from stakeholders in HLIs. Table 5 below shows a sample of user requirements obtained from the HLIs. The requirements were analysed and it appeared that majority of the respondents wanted to get recommendations of useful OSS designed for academic purposes, to be able to view and download OSS and to receive email notification about OSS recommendations. However, some requirements were not very clear which led to the application of evolutionary prototyping as it allows getting users' feedback during development. Table 6 below shows the functional and functional requirements.

Table 5: User Requirements

Requirements	**Respondents**
A table showing usefulness/capacity of various software	6
Availability	2
Categorization of software according to their applicability	2
Data analysis and symbolic math's computations as categories in the application	1
Direct view and download	7
Drop down icon with a list of different topics/subjects	1
Easy access, use and reliability of the web-based application	8
Few clicks to contents, System Uptime.	1
Flexible, Trustworthy and secure	1
Friendly user interface	1
Friendly user interface, navigation bar tool, GUI	1
Interactive, Informative and straight forward explanations	1
A memory of downloaded documents	1
Name of software, features of the software, software manuals	1
Rating and commenting	5
Recommendations, categories of software, easy to search and email notification	7
There should be no page redirection	3
Use simple language	1
User privacy and security	1
We need user friend software	1
A well-Organized menu bar	1
Total	**53**

Table 6: Functional and Non-Functional Requirements

Type of requirement	Application requirement
Functional Requirements	1. The application shall store information about OSS and users 1.1 Application shall store the usefulness/capacity of the various software 1.2 Application shall store OSS in categories of their applicability such as for arts and social sciences 1.3Application shall store a detailed description of OSS 1.4 Application shall allow users to give feedback 1.5 Application shall keep a log of user activities 1.6 Application shall allow the user to register, login and logout 1.7 Application shall allow rating, commenting 1.8 Application shall include data analysis OSS and symbolic math's computations OSS as categories 2. The application shall provide stored information to users 2.1 Application shall allow the user to search for information 2.2 Application shall allow download 2.3 Application shall give a recommendation of OSS based on user preferences 2.4 Application shall send a notification to the user through email based on user subscriptions 2.5 Application shall provide alternative OSS for a particular OSS 2.6 Application shall give manual and other OSS information 2.7 Application shall give updates on OSS
Non-Functional Requirement	3. The application shall be available at any time, reliable, timely updates. 4. The application shall not redirect to other websites 5. The application shall be easy to use 6. The application shall give free access 7. The application shall provide a friendly user interface

4.4 Design of the Web-based Application

4.4.1 System Modelling

System modelling is the process of developing abstract models of a system, in which each model present a different view of that system (Valacich *et al*, 2012). System modelling represents the system using graphical notations, in which now are based on a Unified Modeling Language (UML). It provides various UML diagrams that help the analyst to understand the functionality of the system and models and used to communicate with the customers. In this study, the following diagrams were used to represent the system. First, the data flow diagram was used to show the movement of data between external entities, processes and data stores within the application. Second, the use case diagram was used to represent a task that involved external interaction with the application. Third, the class diagram was used to show an association between different database tables. A UML diagram was also used to show modular decomposition of the application, modules represent independent components of the application that perform a specific task and together form a complex application.

4.4.2 Design of the Web-based Application

Design of the application has two parts which are the database design and the application design. In database design unified modelling language (UML) was used in creating class diagrams to shows different tables for storing data and their relationship which forms a relational database.

Unified Modelling Language created Data Flow Diagrams (DFDs), modular decomposition and use case diagrams to show relationship and operability of internal applications components, different modules that perform various tasks in the application to make a recommender application for OSS and interaction between different actors.

The proposed web-based application for recommendation of OSS for HLIs is shown in Fig. 5 below, the architecture of the application is three-tier architecture in which application server layer that plays a role of storing rules that are used to access data from the database server is added to the client and database server architecture (Elmasri and Navathe, 2004), this architecture is selected because it has an advantage of improving database security by checking client’s credentials before forwarding request to the database server.

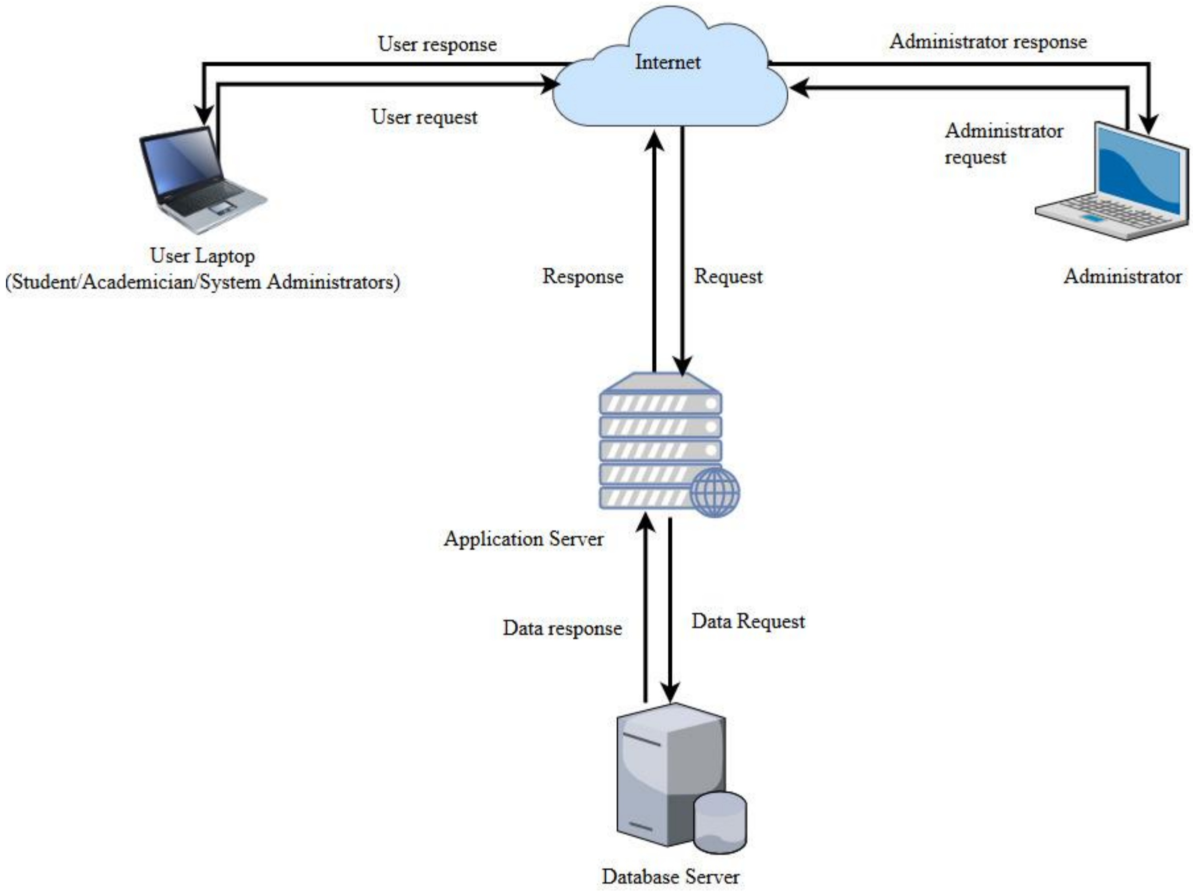


Figure 5: The Proposed System Architecture

(i) Database Server

This server comprises the hardware and software components. They provide the data storage for the proposed solution. The hardware component provides disk space and other physical resources while the software component host the database for the application, and provides query and transaction controls (Elmasri and Navathe, 2004). The database design for the application is presented by the class diagram as shown in Fig. 6. Data Flow Diagrams both context and level 0 as shown in Fig. 7 and Fig. 8 respectively. They show the movement of data between the external entities, processes and data stores within the application (Valacich *et al*., 2012).

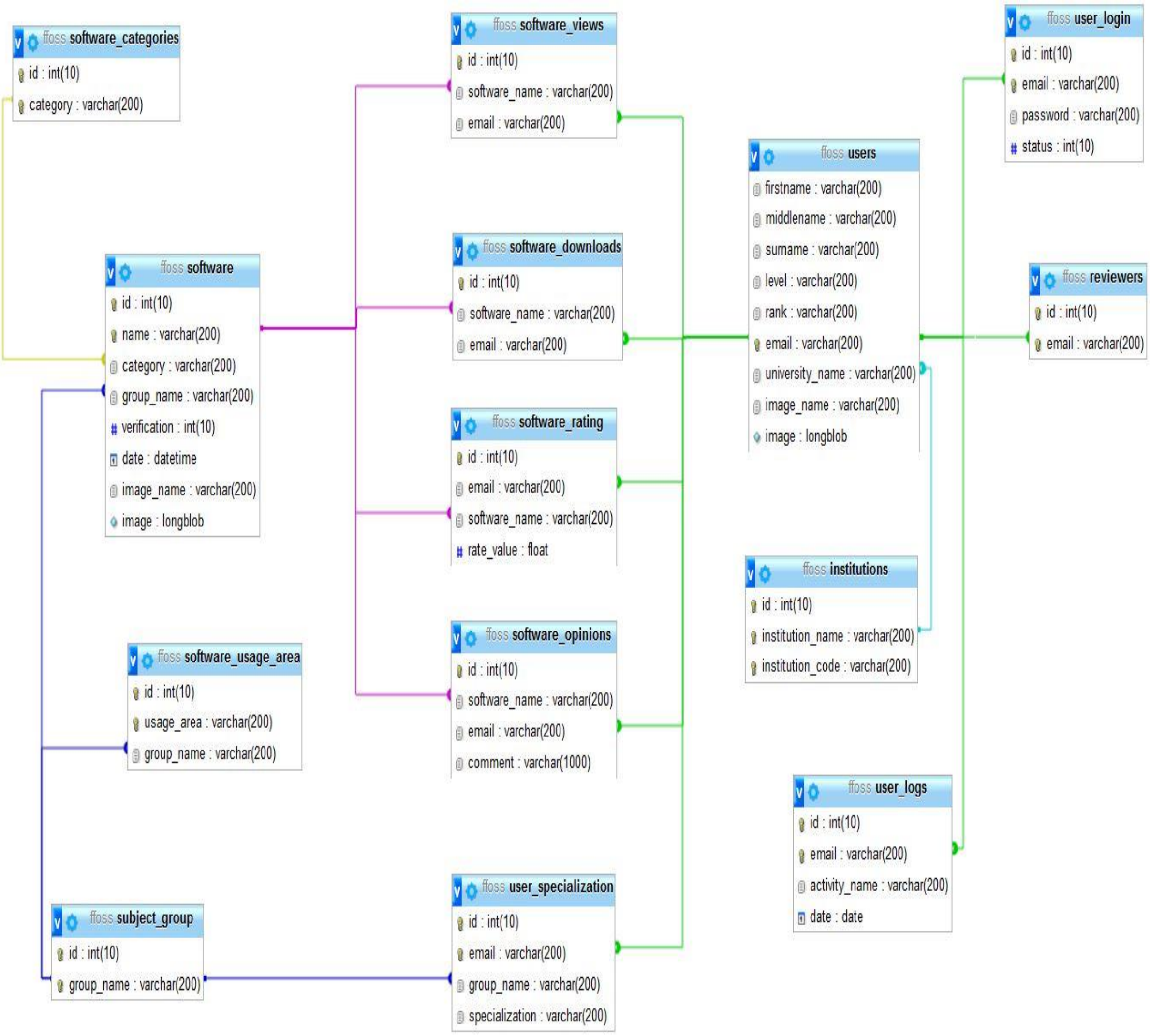


Figure 6: Database Class Diagram

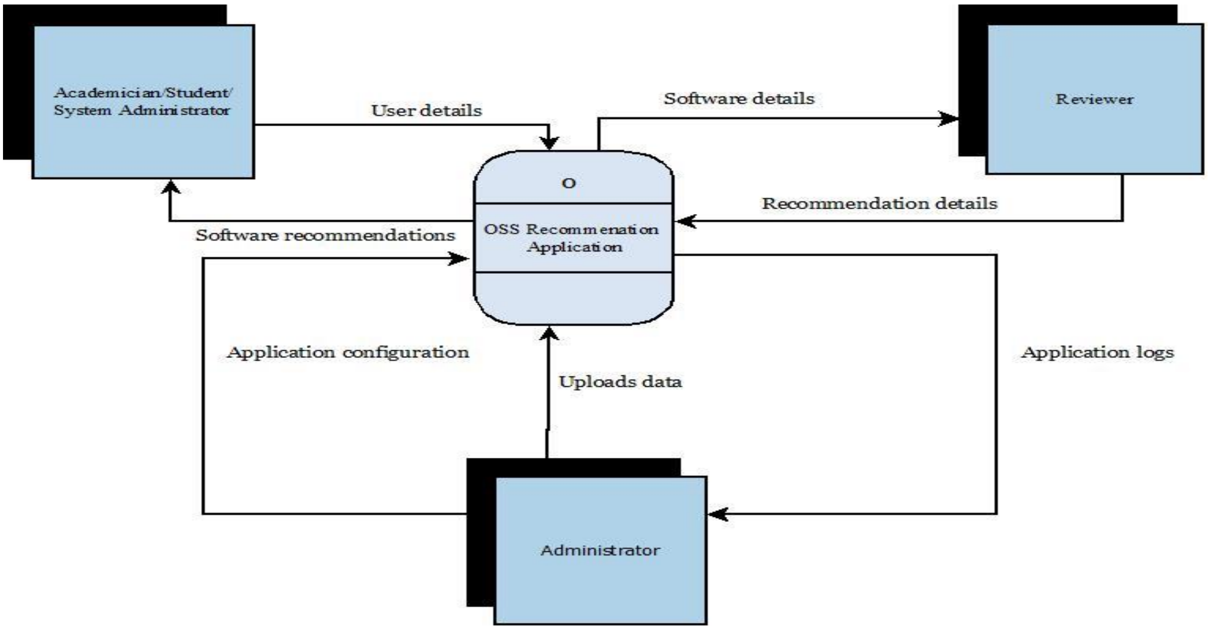


Figure 7: Data Flow Diagram Context Diagram

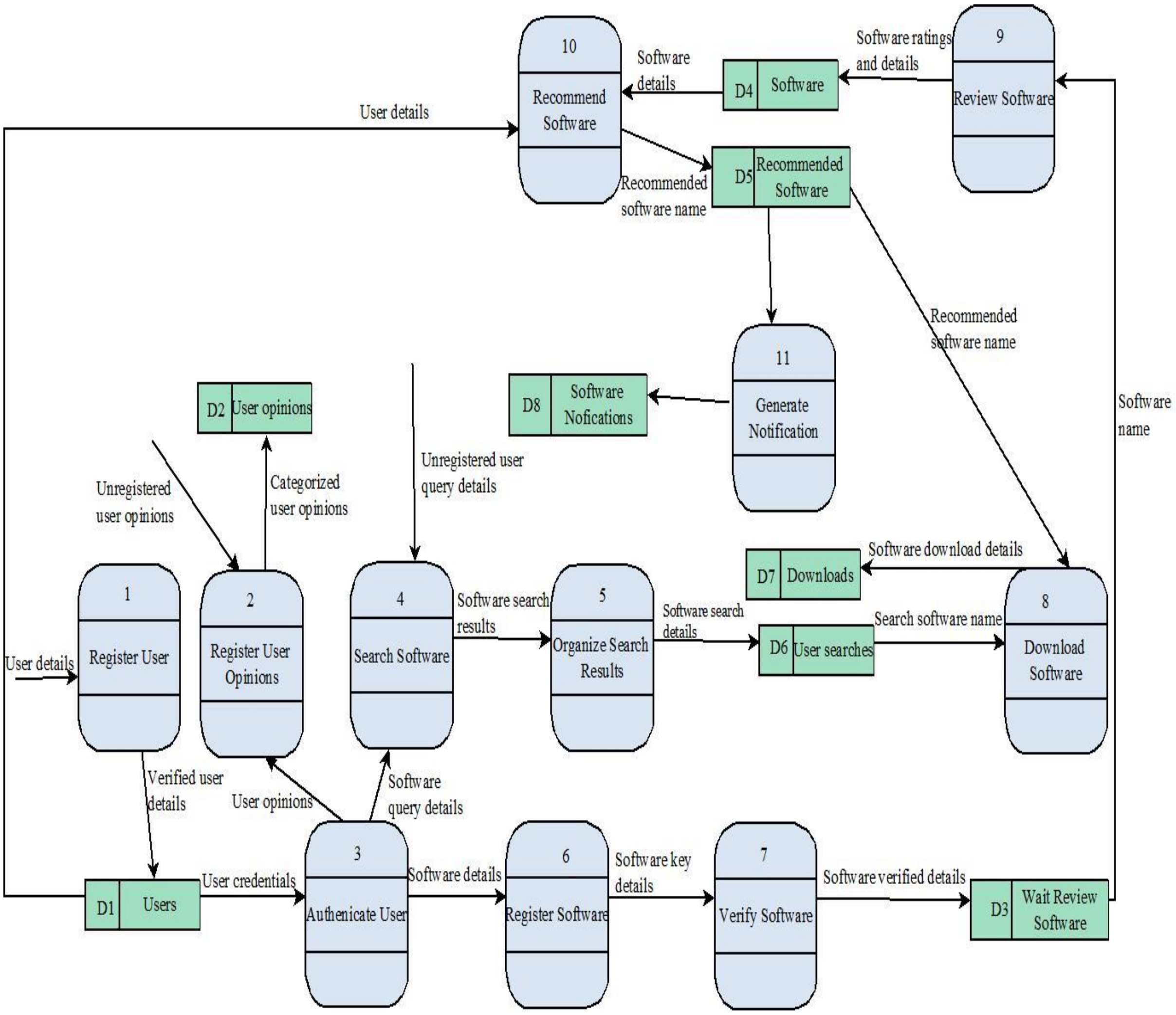


Figure 8: Data Flow Diagram Level 0

(ii) Application Server

The application server, like the database server, has hardware and software. The hardware component provides disk space to store application files and another system resource like the CPU time, RAM etc. On the other hand, the software component provides rules and procedures to handle data access to the database server. It is the main component of the proposed solution as it handles all application operations. The use case diagram in Fig. 9 and the application modules shown in Fig. 10 summarize the structure and operations of the application server.

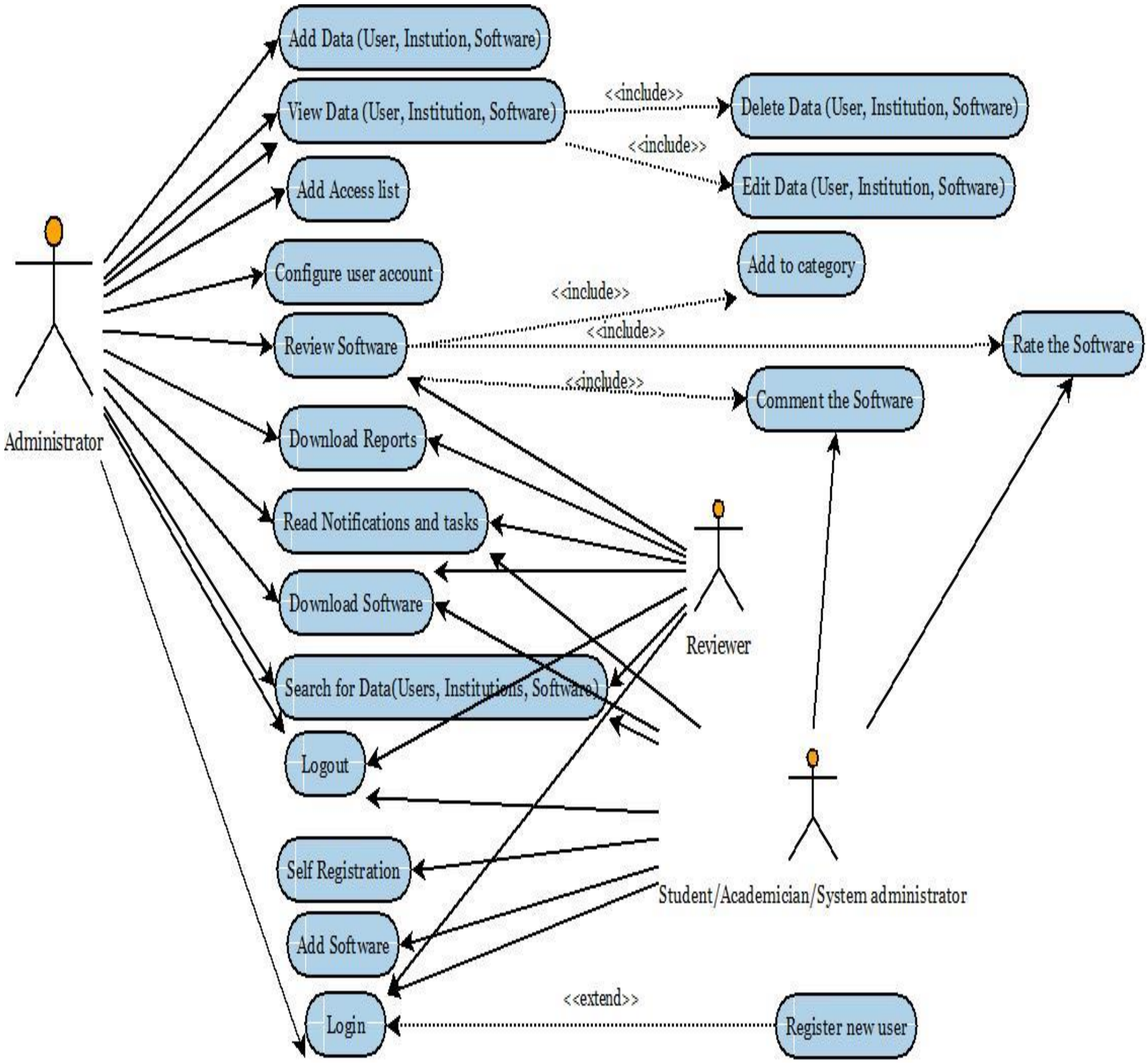


Figure 9: Use Case Diagram

Figure 9 above is composed of actors and use cases, actors are human that interact with the application and they are categories according to their roles in the application. A use case describes how actors use the application to accomplish a particular goal. Table 7 below provides a summary of the above use case diagram

Table 7: Use Cases Descriptions

Use Case	Description	Actors	Pre-condition	Post-condition
View Software	This use case describes how an actor can view details of the software	Student/ Academician /System administrator, Reviewer, and Administrator	Actors must supply application URL to the web browser or must be logged in to the application	The actor will be able to view the software details if the use case is successful.
Download Software	This use case describes how actors can download the software from the application to their local machines.	Student/ Academician /System administrator, Reviewer, and Administrator	Actors must supply application URL to the web browser or must be logged in to the application	The actor will be able to download the software and store to their local machines if the use case is successful.
Add Software	This use case describes how actors can add software to the application.	Student/ Academician /System administrator, Reviewer, and Administrator	Actors must be logged in to the application.	The actor will be able to add the software that does not exist if the use case is successful.
Review Software	This use case describes how an actor can rate and comment on the software.	Reviewer	The actor must be logged in to the application.	Actors will be able to review the software if the use case is successful.
Read Notification	This use case describes how an actor can receive notifications about software from the application	Student/ Academician /System administrator, Reviewer, and Administrator	The actor must be logged in to the application and subscribed to at least one educational area of specialization	The actor will receive notifications to their email address if the use case is successful.
Self-Registration	This use case describes how an actor can register for using the application	Student/ Academician /System administrator	The actor must possess a valid email address	The actor will be registered and initial password will be sent to his/her email address
Login	This use case describes how an actor can log in to gain access to the application	Student/ Academician /System administrator, Reviewer, and Administrator	The actor must possess a valid email address and password	The actor will be logged in to the application if the use case is successful.
Logout	This use case describes how an actor can successfully close his/her login session	Student/ Academician /System administrator, Reviewer, and Administrator	The actor must have a valid login session.	The actor will be logged out and his/her session will be terminated if the use case is successful.

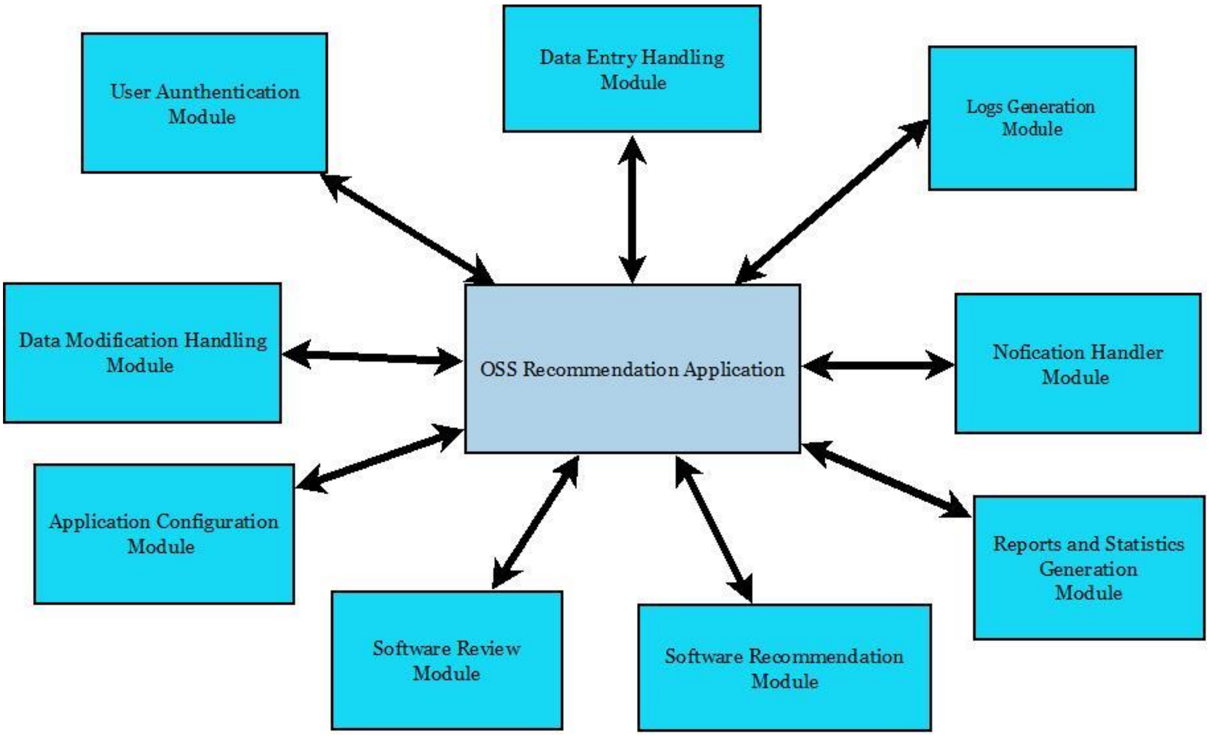

User Aunthentication Module
Data Entry Handling Module
Logs Generation Module
Data Modification Handling Module
OSS Recommendation Application
Nofication Handler Module
Application Configuration Module
Reports and Statistics Generation Module
Software Review Module
Software Recommendation Module

Figure 10: Application Modules

4.4.3 Structure of the Recommendation Algorithms

The structure of how the recommendation process works is presented using the decision tree. The application checks if the uploaded OSS academic area of use is similar to active user's academic area of specialization, or if it is in the same area of use as those the active user viewed in the past, or if it viewed by other users that share academic area of specialization with the active users. The summary of the decisions is shown in Fig. 11.

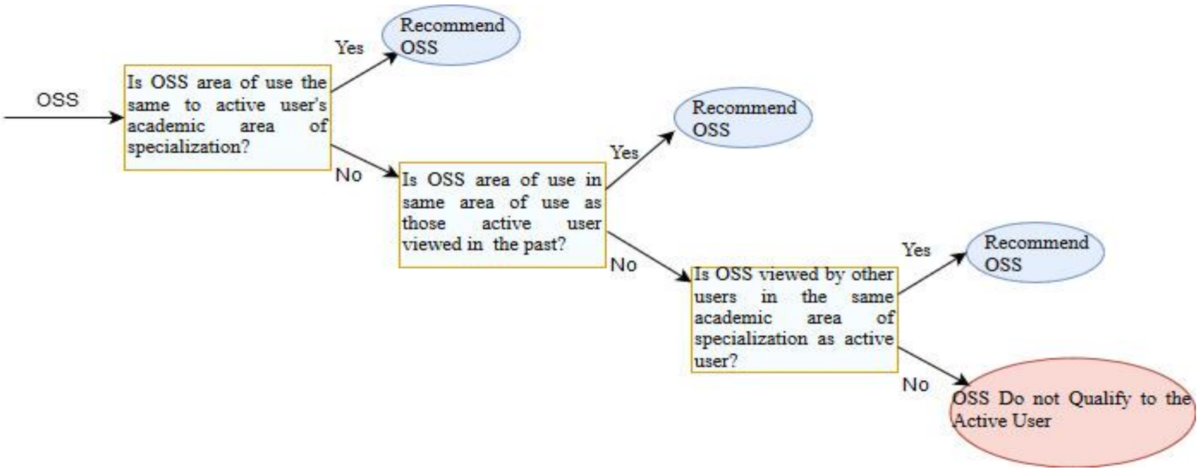

OSS
Is OSS area of use the same to active user's academic area of specialization?
Yes
Recommend OSS
No
Is OSS area of use in same area of use as those active user viewed in the past?
Yes
Recommend OSS
No
Is OSS viewed by other users in the same academic area of specialization as active user?
Yes
Recommend OSS
No
OSS Do not Qualify to the Active User

Figure 11: Decision Tree of the Recommendation Algorithms

4.4.4 Web-based Application Functionalities

The application functionalities were implemented using the web technologies discussed in the previous section. Users refer to students, academicians, system administrators and lab scientists or any member of HLIs.

(i) **User Registration:** Users are allowed to use the application with or without registration, the difference is that registered users benefit with more application services. During registration, users are required to provide various information including email and academic area of specialization. The email is verified and used for receiving notification and academic area of specialization was used for recommendation purposes.

(ii) **Open Source Software Uploading:** The application gives privilege to the administrator and other users to upload the OSS. They are required to provide important information about the OSS that included a link to the official website, academic area of use, supported operating system, key features and description. The academic area of use is used for recommendation purposes.

(iii) **User commenting and Rating:** Users are given the ability to comment and rate the OSS after use to help other users of the OSS. The application computes the Average Rating Value (ARV) of each OSS, and it is used to rank the OSS during computations of recommendations.

(iv) **Open Source Software Recommendations:** Open Source Software recommendations is the main functionality of this application. The academic area of specialization of the user, the academic area of use of the OSS and the ARV is used in the computation in recommendation functions. Three recommendation approaches were used in the application as discussed below.

a) **Content-based approach:** The application recommends OSS to the active user that are in the same academic area of use as those OSS that user liked in the past. The OSS are then sorted based on the ARV. Figure 12 presents the algorithm using the flowchart.

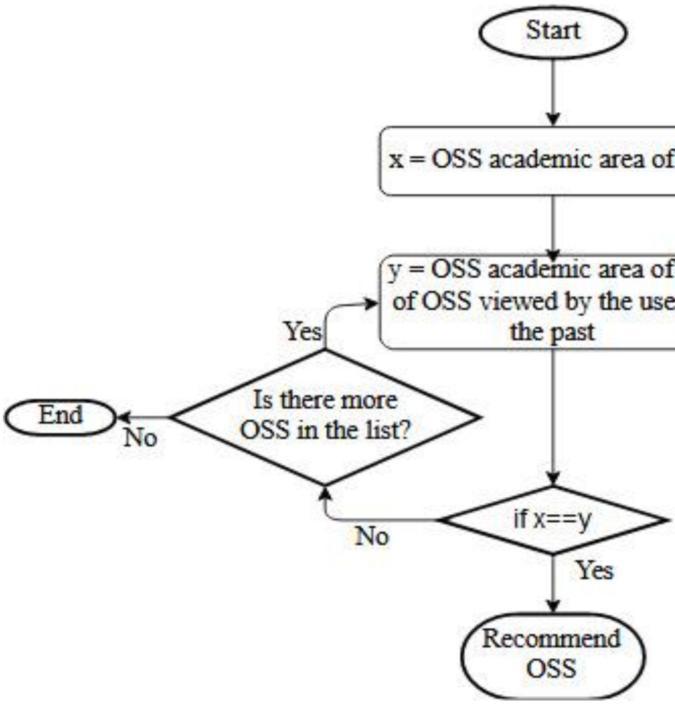

Start
x = OSS academic area of use
y = OSS academic area of use of OSS viewed by the user in the past
Yes
End
No
Is there more OSS in the list?
if x==y
No
Yes
Recommend OSS

Figure 12: Content-based Approach Algorithm

b) Demographic approach: The application recommends OSS to the active user when user's academic area of specialization is similar to the OSS academic area of use. The OSS are then sorted based on the ARV. Figure 13 presents the algorithm using the flow chart.

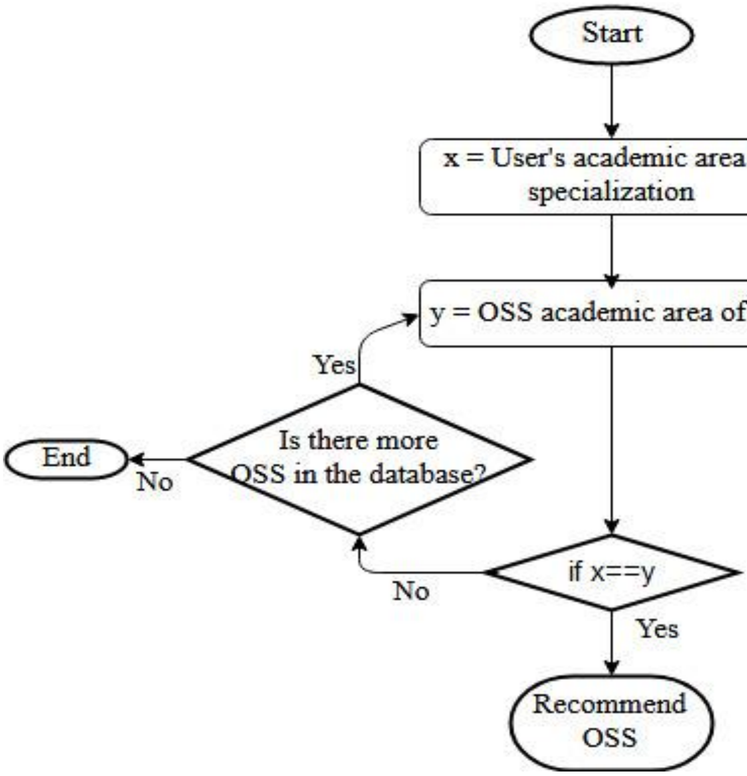

Start
x = User's academic area of specialization
y = OSS academic area of use
Yes
Is there more OSS in the database?
End
No
if x==y
No
Yes
Recommend OSS

Figure 13: Demographic Approach Algorithm

c) **Collaborative filtering approach:** The application recommends OSS to the active user based on which OSS other users from a similar academic area from specialization have liked in the past. The OSS are then sorted based on the ARV. Figure 14 presents the algorithm using the flowchart.

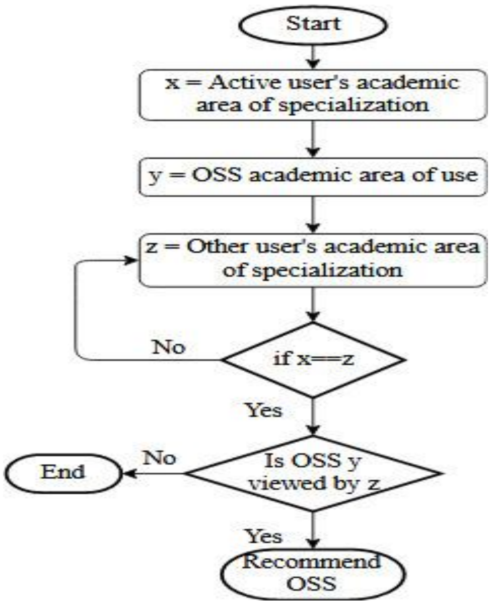

Start
x = Active user's academic area of specialization
y = OSS academic area of use
z = Other user's academic area of specialization
No
if x==z
Yes
End
No
Is OSS y viewed by z
Yes
Recommend OSS

Figure 14: Collaborative Filtering Approach Algorithm

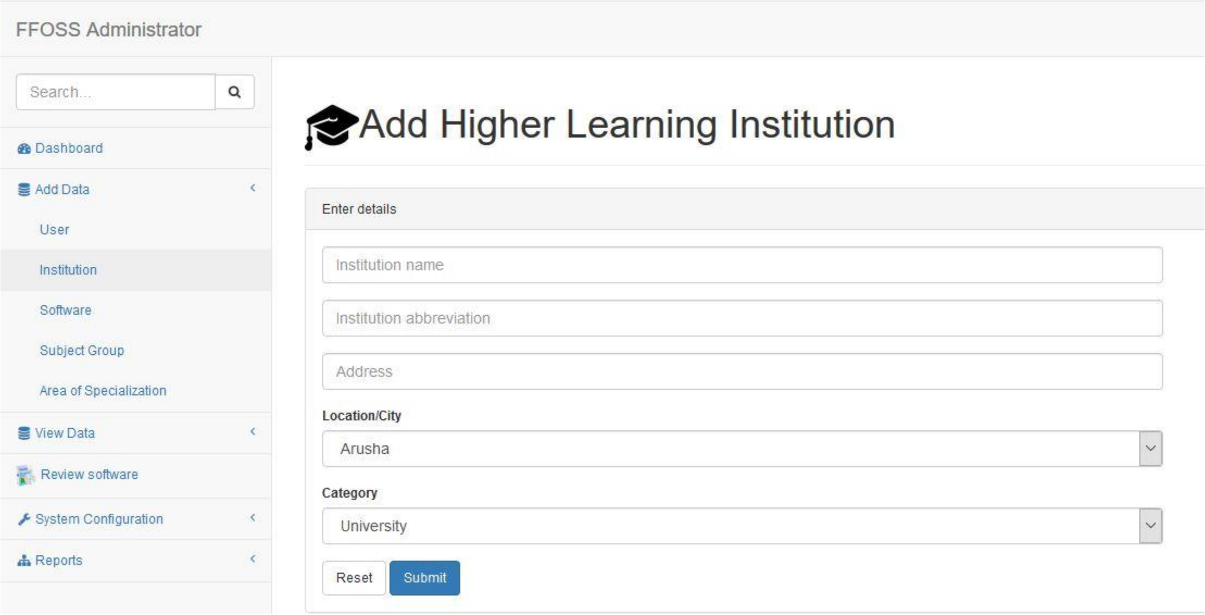
The three approaches used for recommendations make the application a hybrid recommender application because it uses more than one recommendation approach to recommend OSS to active users. Further, the application sends a notification to the user when they have a new OSS recommendation.

4.5 The Developed Web-based Application for Recommendation of OSS for HLIs

This section discusses the results of the developed web-based application for the recommendation of OSS for HLIs by demonstrating various functions of the application.

4.5.1 Application Configuration

The application allowed the administrator to add OSS and HLIs. These were the basic data for the core functionality of the application. Further, this functionality allowed the administrator to add users' subject groups and user area of specializations to the application such that each user do not have to enter these data but just to choose them in a drop menu. The application administrator defined these configuration data through an HTML form, and PHP script was used to validate such that there are no duplicate entries to the database and finally SQL was used to write those data permanently to the database. Figure 15 below shows one of the HTML forms that enabled the administrator to configure the application.



Add Higher Learning Institution

Enter details

Institution name

Institution abbreviation

Address

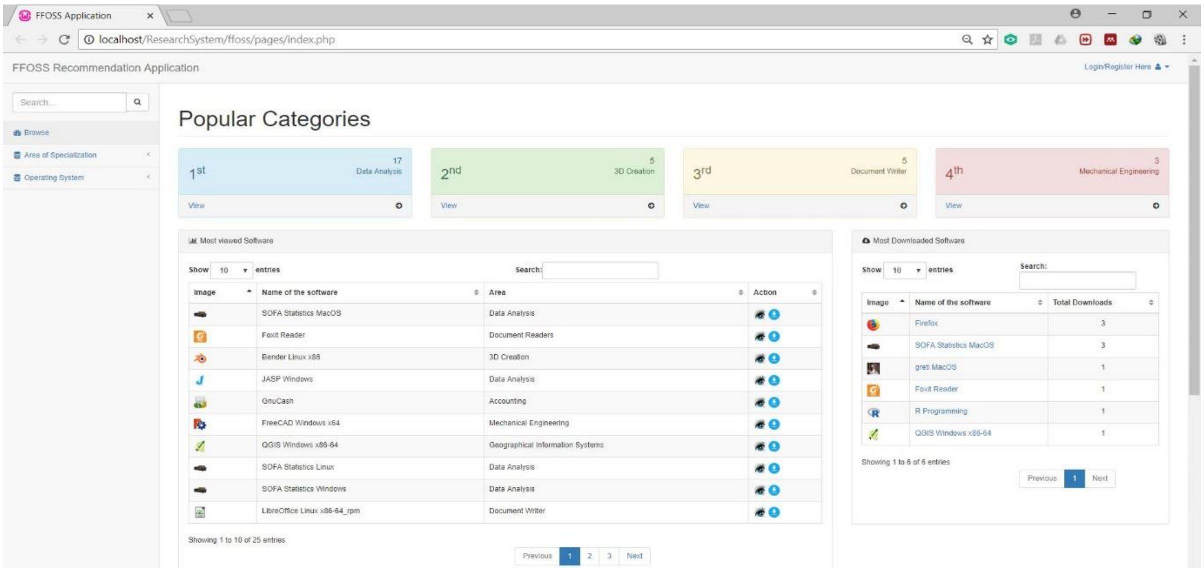
Location/City: Arusha

Category: University

Figure 15: HTML form for adding HLI

4.5.2 Graphical User Interfaces

The application provided a Graphical User Interface (GUI) to allow easy interaction with the users. The information to be displayed on the web page were defined using HTML while physical layout was implemented using CSS and JavaScript. The interfaces contained forms where the user could fill in data, navigation menus where the user can navigate through different other interfaces, links such as view and download, tables where the user could view different information from the database. Figure 16 below shows the homepage of the interface which was implement using HTML, CSS, and JavaScript.



Popular Categories

1st	17 Data Analysis	View	
2nd	5 3D Creation	View	
3rd	5 Document Writer	View	
4th	3 Mechanical Engineering	View	

Most viewed Software

Show 10 entries Search:

Image	Name of the software	Area	Action
	SOFA Statistics MacOS	Data Analysis	
	Foxit Reader	Document Readers	
	Bender Linux x86	3D Creation	
	JASP Windows	Data Analysis	
	GnuCash	Accounting	
	FreeCAD Windows x64	Mechanical Engineering	
	QGIS Windows x86-64	Geographical Information Systems	
	SOFA Statistics Linux	Data Analysis	
	SOFA Statistics Windows	Data Analysis	
	LibreOffice Linux x86-64_rpm	Document Writer	

Showing 1 to 10 of 25 entries

Most Downloaded Software

Show 10 entries Search:

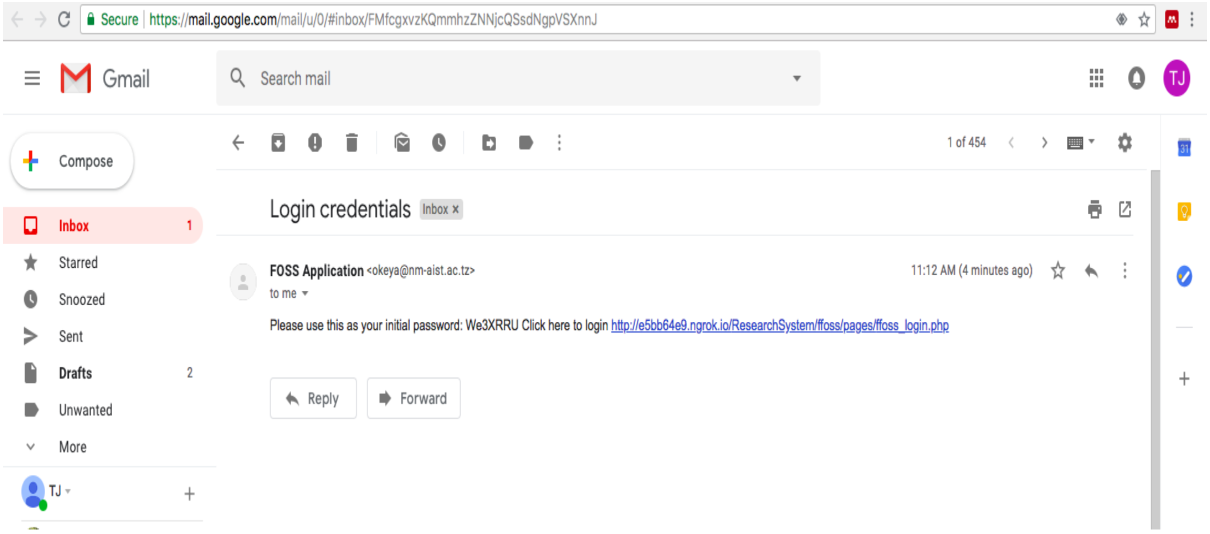
Image	Name of the software	Total Downloads
	Firefox	3
	SOFA Statistics MacOS	3
	gretl MacOS	1
	Foxit Reader	1
	R Programming	1
	QGIS Windows x86-64	1

Showing 1 to 6 of 6 entries

Figure 16: The Application Homepage

4.5.3 User Registration

The users were allowed to register themselves for using more functions of the application, they provided their details through an HTML form. The information provided was validated using JavaScript such that no invalid email can be accepted and empty form cannot be submitted. Further, JavaScript was used to hide and show input fields depending on what input provided by the user in another input field. The PHP and SQL were used to communicate with the database; to check if the email provided exist in the database or not, and to store user details permanently to the database on if the user does not exist. Moreover, PHP was used to encrypt the user password using md5 encryption before storing it in the database. The user received login credentials through their emails to confirm their accounts. Figure 17 below shows the email from the application which shows user login credentials.



Login credentials

FOSS Application <okeya@nm-aist.ac.tz>
to me
11:12 AM (4 minutes ago)

Please use this as your initial password: We3XRRU Click here to login http://e5bb64e9.ngrok.io/ResearchSystem/ffoss/pages/ffoss_login.php

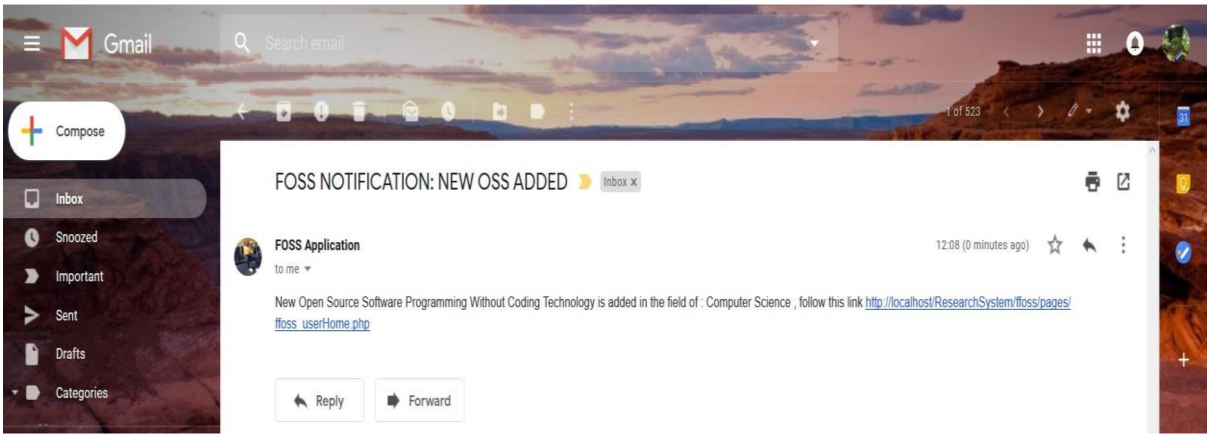
Figure 17: Login Credential through User's Email Address

4.5.4 User Login

The users were allowed to log in to the application by providing their username and password through the HTML form, JavaScript was used to validate user input such that the username is a valid email address and empty form cannot be submitted. The PHP and SQL were used to fetch user details from the database and compare them with those entered by the user since the password was encrypted, the user-supplied password was first encrypted using md5 technique before comparison. Further, PHP was used to check if the email and password supplied by the user matched with those in the database. If they matched, the user was granted access to the system, otherwise, access was denied.

4.5.5 Open Source Software Uploading

The users were allowed to upload OSS to the application by providing details specified in the HTML form including the setup file. JavaScript was used to add or remove input field upon user selection such as adding features of the OSS. The PHP and SQL were used to fetch OSS details from the database and compare them with those entered by the user, if the OSS exists in the database the upload action will be denied otherwise OSS will be permanently written to the database. Further, to prevent users from blind waiting, JavaScript was used to monitor the progress of file upload. Moreover, after successfully upload, notification was sent to users who subscribed to receive notification from the category that OSS is added. Figure 18 below shows the email notification sent to the user when new OSS is uploaded.



FOSS NOTIFICATION: NEW OSS ADDED

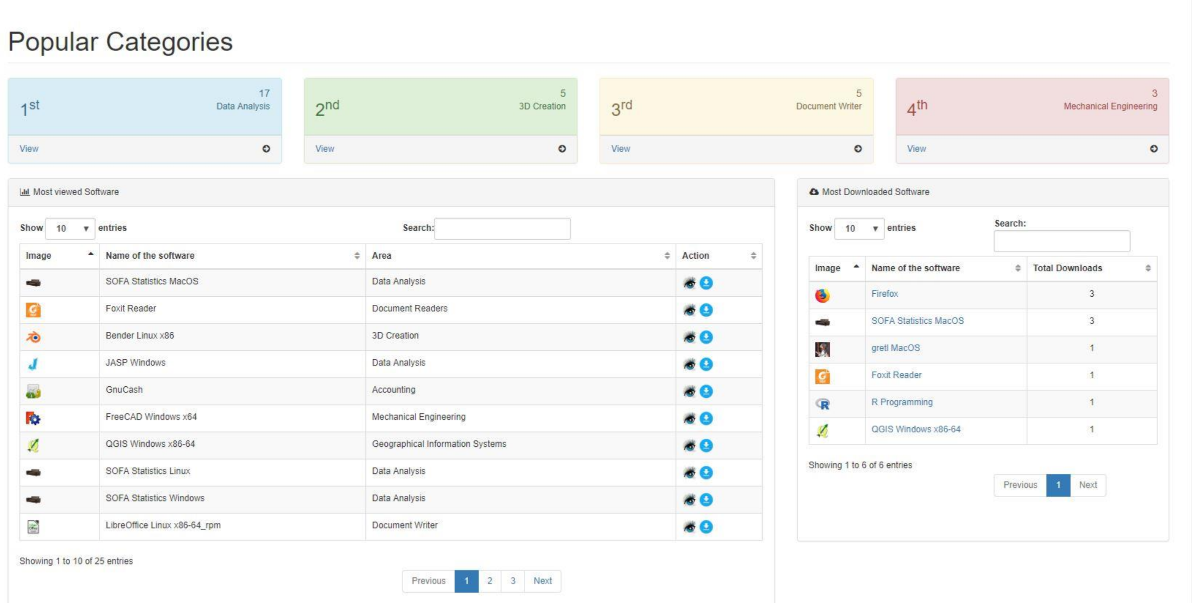
FOSS Application
to me
12:08 (0 minutes ago)

New Open Source Software Programming Without Coding Technology is added in the field of : Computer Science , follow this link http://localhost/ResearchSystem/ffoss/pages/ffoss_userHome.php

Figure 18: Notification when new OSS is uploaded

4.5.6 Displaying Stored Data

The users were allowed to view stored data, this data include OSS. Hypertext Preprocessor and SQL were used to fetch data from the database and display them. The HTML table was used to show the data, JavaScript was used to manipulate the data displayed in the table by searching and sorting the data. Further, the table was able to do pagination. Figure 19 below shows a table displaying stored data.



1st	2nd	3rd	4th
17 Data Analysis	5 3D Creation	5 Document Writer	3 Mechanical Engineering

Most viewed Software

Image	Name of the software	Area	Action
	SOFA Statistics MacOS	Data Analysis	
	Foxit Reader	Document Readers	
	Bender Linux x86	3D Creation	
	JASP Windows	Data Analysis	
	GnuCash	Accounting	
	FreeCAD Windows x64	Mechanical Engineering	
	QGIS Windows x86-64	Geographical Information Systems	
	SOFA Statistics Linux	Data Analysis	
	SOFA Statistics Windows	Data Analysis	
	LibreOffice Linux x86-64_rpm	Document Writer	

Showing 1 to 10 of 25 entries

Most Downloaded Software

Image	Name of the software	Total Downloads
	Firefox	3
	SOFA Statistics MacOS	3
	gretl MacOS	1
	Foxit Reader	1
	R Programming	1
	QGIS Windows x86-64	1

Showing 1 to 6 of 6 entries

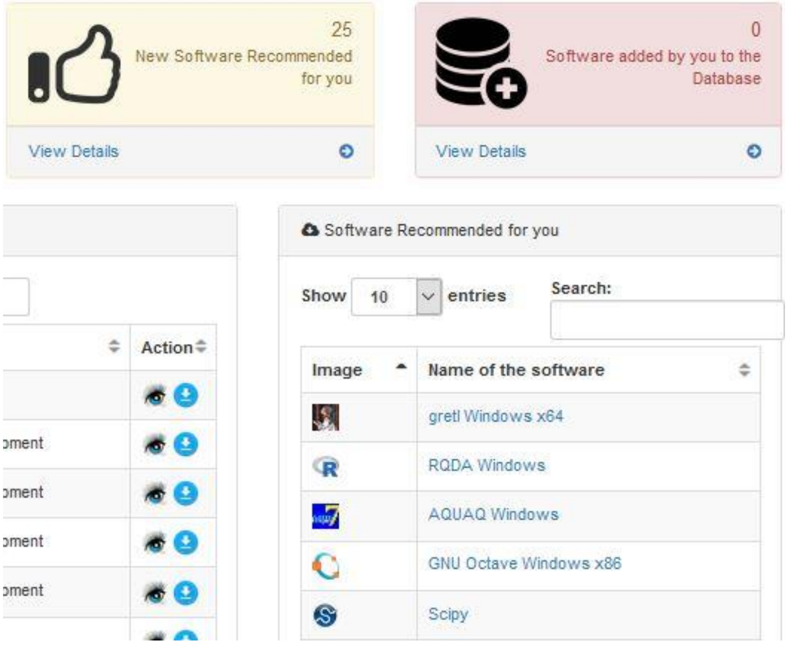
Figure 19: Table for Displaying Data

4.5.7 Update Stored Data

The users needed to update stored data, stored data were displayed on the HTML form using PHP and SQL. The update information was written to the database using PHP and SQL.

4.5.8 Open Source Software Recommendation

The application provides OSS recommendation to the users as a result of the implementation of recommendation algorithms. The recommendation approaches discussed in section 4.4.4 of this dissertation were implemented and the OSS are recommended to the users. The dashboard of the user's page shows a summary of the number of OSS recommended to the user, also the right section of the same page shows a list of the OSS recommended to the user. Figure 20 shows an example of how the application recommended OSS to one of the users.



25
New Software Recommended for you
View Details

0
Software added by you to the Database
View Details

	Action
oment	
oment	
oment	
oment	

Software Recommended for you

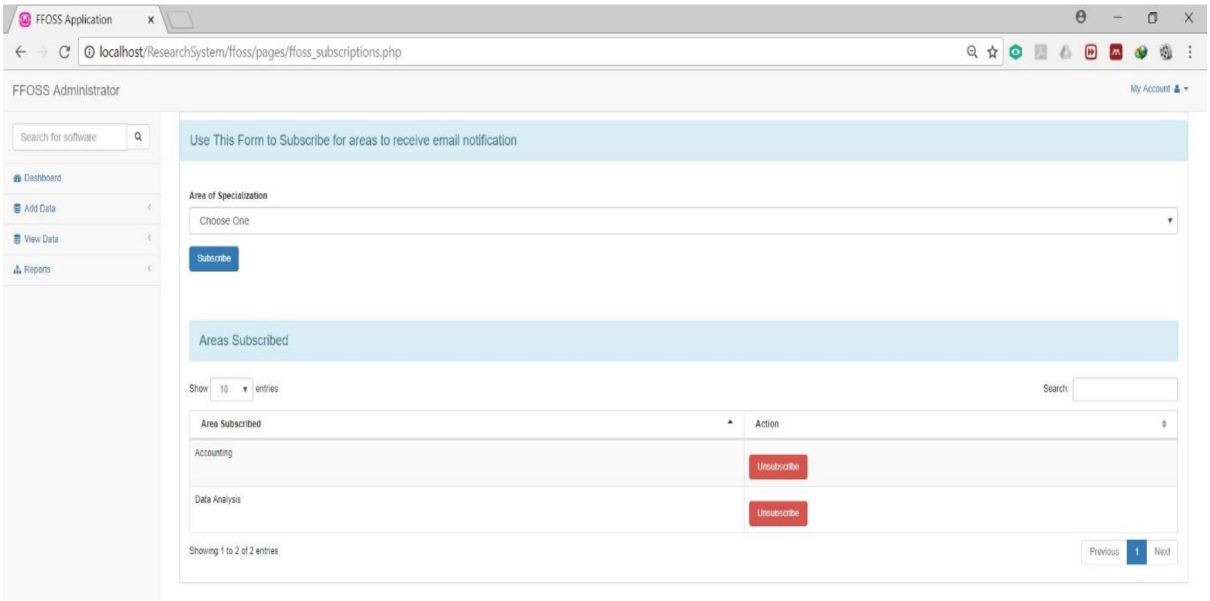
Show 10 entries

Search:

Image	Name of the software
	gretl Windows x64
	RQDA Windows
	AQUAQ Windows
	GNU Octave Windows x86
	Scipy

Figure 20: Open Source Software Recommendations

On top of the recommendation based on the user's academic area of specialization, users were allowed to choose other areas to subscribe for receiving recommendations. Figure 21 below shows the user's subscriptions.



Use This Form to Subscribe for areas to receive email notification

Area of Specialization

Choose One

Areas Subscribed

Show 10 entries Search:

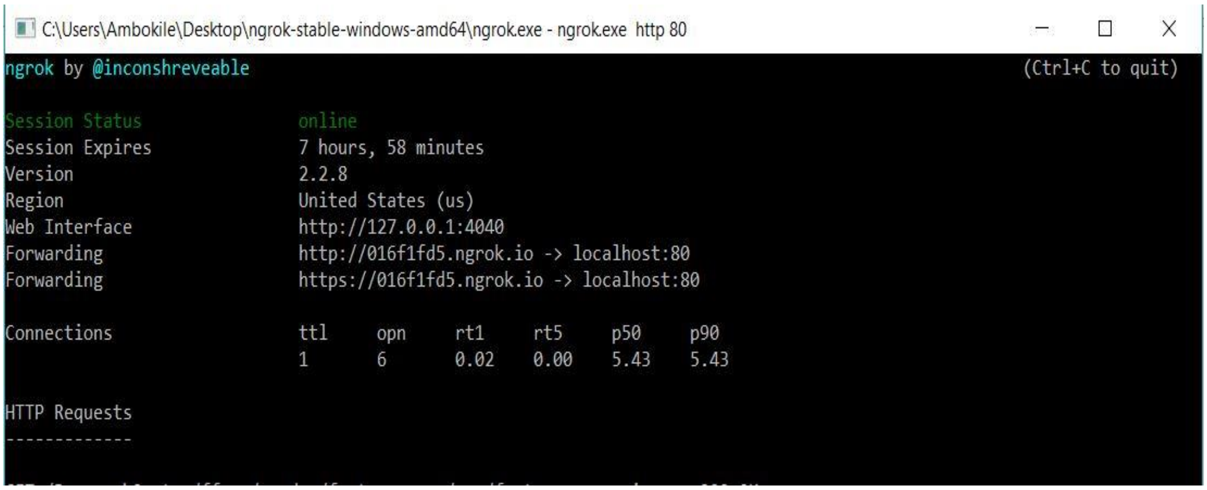
Area Subscribed	Action
Accounting	Unsubscribe
Data Analysis	Unsubscribe

Showing 1 to 2 of 2 entries

Figure 21: User's Subscriptions Control Page

4.5.9 Application Accessibility

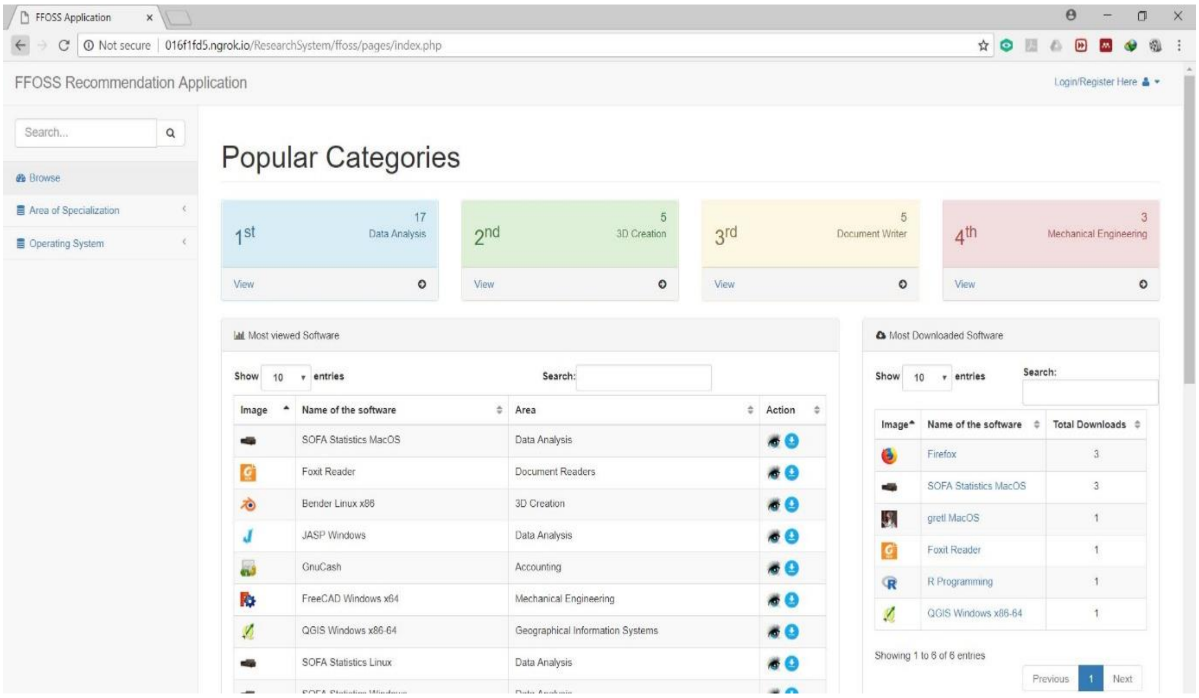
The application was configured to be accessible through the Internet. Ngrok is a software application that exposes local server behind NATs and firewalls to the public Internet over secure tunnels ("Ngrok - secure introspectable tunnels to localhost," 2018). Recall the apache configurations in section 3.8 of this dissertation, ngrok was used to verify these configurations and the result showed that the application is accessible through the Internet from any IP address. Figure 22 below shows ngrok running and displaying some information including subdomain name given.



```
C:\Users\Ambokile\Desktop\ngrok-stable-windows-amd64\ngrok.exe - ngrok.exe  http 80
ngrok by @inconshreveable                                                    (Ctrl+C to quit)

Session Status                online
Session Expires               7 hours, 58 minutes
Version                       2.2.8
Region                        United States (us)
Web Interface                 http://127.0.0.1:4040
Forwarding                    http://016f1fd5.ngrok.io -> localhost:80
Forwarding                    https://016f1fd5.ngrok.io -> localhost:80

Connections                   ttl     opn     rt1     rt5     p50     p90
                              1       6       0.02    0.00    5.43    5.43

HTTP Requests
-------------
```

Figure 22: Ngrok Running

Figure 23 below shows a computer that accessed the application through ngrok using the given subdomain.



Popular Categories

1st	17 Data Analysis	View	
2nd	5 3D Creation	View	
3rd	5 Document Writer	View	
4th	3 Mechanical Engineering	View	

Most viewed Software

Show 10 entries Search:

Image	Name of the software	Area	Action
	SOFA Statistics MacOS	Data Analysis	
	Foxit Reader	Document Readers	
	Bender Linux x86	3D Creation	
	JASP Windows	Data Analysis	
	GnuCash	Accounting	
	FreeCAD Windows x64	Mechanical Engineering	
	QGIS Windows x86-64	Geographical Information Systems	
	SOFA Statistics Linux	Data Analysis	
	SOFA Statistics Windows	Data Analysis	

Most Downloaded Software

Show 10 entries Search:

Image	Name of the software	Total Downloads
	Firefox	3
	SOFA Statistics MacOS	3
	gretl MacOS	1
	Foxit Reader	1
	R Programming	1
	QGIS Windows x86-64	1

Showing 1 to 6 of 6 entries

Figure 23: User-Computer accessing the Application through the Internet using the Ngrok Secure Tunnel

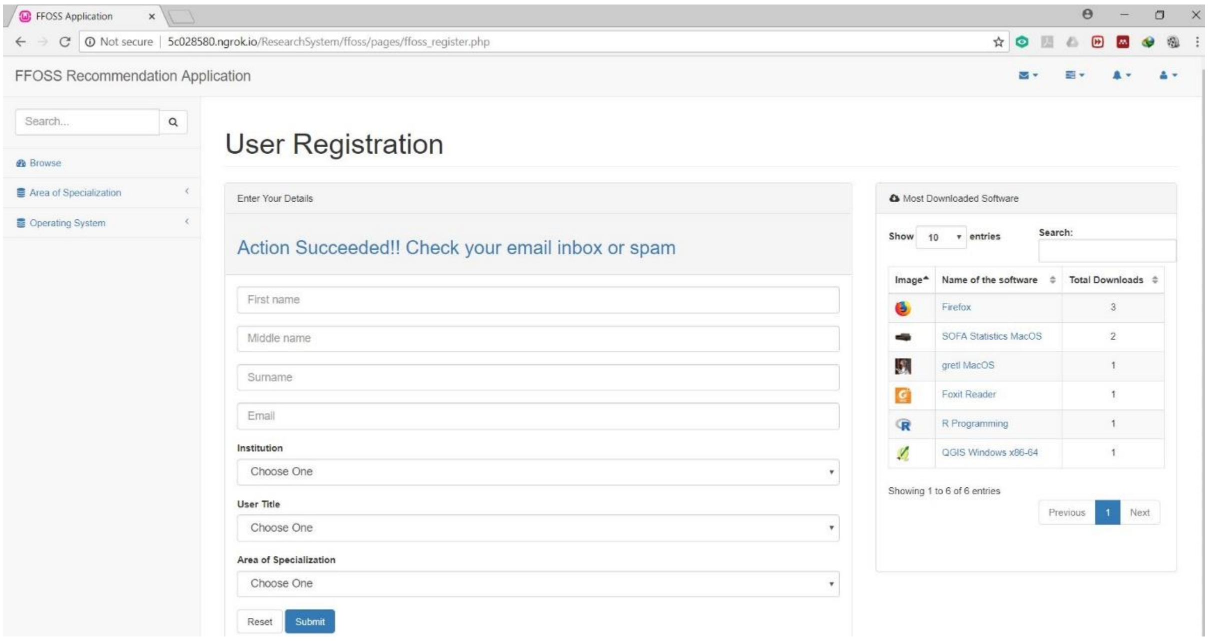
4.6 Experimental results

Verification and validation are conducted on the developed application to make sure there are no coding errors, it meets all of the requirements defined in the software requirements specification and it delivers services required by its intended users. Unit testing and validation testing was conducted using dummy data in the laboratory. Acceptance testing was conducted by involving the intended user on using the application and give their feedback.

4.6.1 Results of Verification

Verification testing is normally conducted to address that software is built right. For this sake unit testing was conducted in the laboratory for different modules of the application.

Data entry handling module was a fundamental module as it provides the foundation for other modules. It was dealing with user registration and software uploading. Testing result for this module passed and all activities that involved data entry could successfully write to the database. A user input data through an HTML form and PHP was used to write data to the database. Figure 24 and 25 below shows the successful user registration notification and OSS uploading success notification respectively.

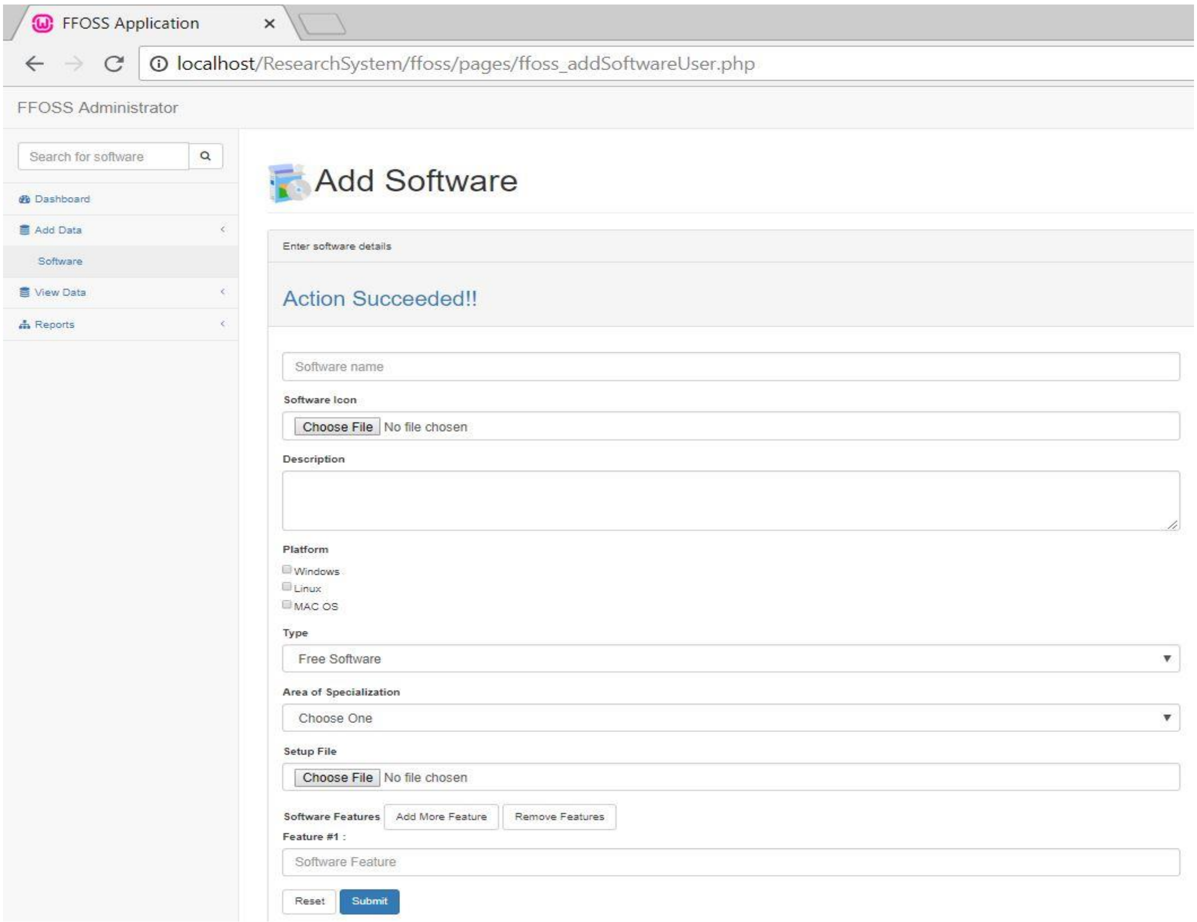


Action Succeeded!! Check your email inbox or spam

Image	Name of the software	Total Downloads
	Firefox	3
	SOFA Statistics MacOS	2
	gretl MacOS	1
	Foxit Reader	1
	R Programming	1
	QGIS Windows x86-64	1

Showing 1 to 6 of 6 entries

Figure 24: User Successful Registration Notification



FFOSS Application

localhost/ResearchSystem/ffoss/pages/ffoss_addSoftwareUser.php

FFOSS Administrator

Add Software

Enter software details

Action Succeeded!!

Software name

Software Icon: Choose File No file chosen

Description

Platform

□ Windows
□ Linux
□ MAC OS

Type: Free Software

Area of Specialization: Choose One

Setup File: Choose File No file chosen

Software Features

Feature #1 : Software Feature

Figure 25: Uploading OSS Success Notification

The following are the results of this data entry handling module:

(i) Users could register and log in and receive their recommendation and perform other function.

(ii) The software was added and could be viewed and downloaded by users. Figure 26 below shows software ready for view and download.

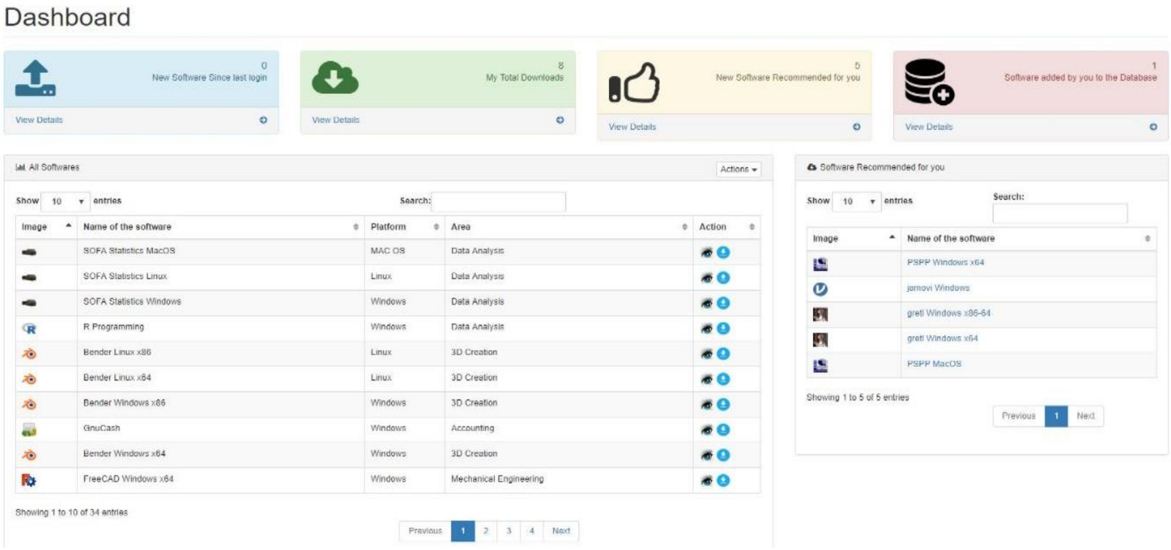


Image	Name of the software	Platform	Area	Action
	SOFA Statistics MacOS	MAC OS	Data Analysis	
	SOFA Statistics Linux	Linux	Data Analysis	
	SOFA Statistics Windows	Windows	Data Analysis	
	R Programming	Windows	Data Analysis	
	Bender Linux x86	Linux	3D Creation	
	Bender Linux x64	Linux	3D Creation	
	Bender Windows x86	Windows	3D Creation	
	GnuCash	Windows	Accounting	
	Bender Windows x64	Windows	3D Creation	
	FreeCAD Windows x64	Windows	Mechanical Engineering	

Showing 1 to 10 of 34 entries

Image	Name of the software
	PSPP Windows x64
	jamovi Windows
	gretl Windows x86-64
	gretl Windows x64
	PSPP MacOS

Showing 1 to 5 of 5 entries

Figure 26: Software View, Download, and Recommendation

Table 8 below shows the results of the unit testing of various modules in the application

Table 8: Verification Results

Test Number	Test Performed	Test Result
	Unregistered User	
1	Allow registration if email provided does not exist in the database	PASS
2	Allow view and download of software for unregistered user	PASS
3	Recommend software for unregistered users	PASS
4	Allow the unregistered user to browse for various categories defined in the application	PASS
	Registered Users	
5	Allow the user to login if correct username and password is supplied	PASS
6	Deny user to login if incorrect username or password is supplied	PASS
7	Allow only users with privileges to perform a certain action	PASS
8	Allow users with privileges to add and modify data to the database	PASS
9	Allows users to view and download software	PASS
10	Provide recommendations specifically to the intended user not a group of users	PASS
11	Send notification only to the intended user	PASS

4.6.2 Results of Validation

Validation is normally conducted to address that the right software is built. For this sake application was tested against the software requirements specification for this study. Table 9 below shows a summary of the tests against the requirements.

Table 9: Validation Results

Type of requirement	Application requirement	Test Result
Functional Requirements	1. The application shall store information about OSS and users	PASS
	1.1 Application shall store the usefulness/capacity of the various software	PASS
	1.2 Application shall store OSS in categories of their applicability such as for arts and social sciences	PASS
	1.3 Application shall store a detailed description of OSS	PASS
	1.4 Application shall allow users to give feedback	PASS
	1.5 Application shall keep a log of user activities	PASS
	1.6 Application shall allow the user to register, login and logout	PASS
	1.7 Application shall allow chat forum, sharing, rating, commenting	PASS
	1.8 Application shall include data analysis OSS and symbolic math's computations OSS as categories	PASS
	2. The application shall provide stored information to users	PASS
	2.1 Application shall allow the user to search for information	PASS
	2.2 Application shall allow download	PASS
	2.3 Application shall give a recommendation of OSS based on user preferences	PASS
	2.4 Application shall send a notification to the user through email based on user subscriptions	PASS
	2.5 Application shall provide alternative OSS for a particular OSS	PASS
	2.6 Application shall give manual and other OSS information	PASS
	2.7 Application shall give updates on OSS	PASS
Non-Functional Requirement	3. The application shall be available at any time, reliable, timely updates.	PASS
	4. The application shall not redirect to other websites	PASS
	5. The application shall be easy to use	PASS
	6. The application shall give free access	PASS
	7. The application shall provide a friendly user interface	PASS

4.6.3 Results of Acceptance Testing

Acceptance testing was conducted by involving students, academicians, systems administrators, laboratory engineers and researchers in using the application, evaluation criteria were set and users were required to give their feedback using questionnaires. The purpose was to assess the application usability. The following is the discussion of results obtained from the users’ experience

(i) Application of Ease of Use

Users were asked to use the application and assess the ease of use for the developed application. Out of 30 users that were involved in testing the application, 43.3% strongly agreed, 46.7% agreed and 10.0% disagreed. These results indicated that the developed application was easy to learn and use. Figure 27 below shows the summary of the stated results

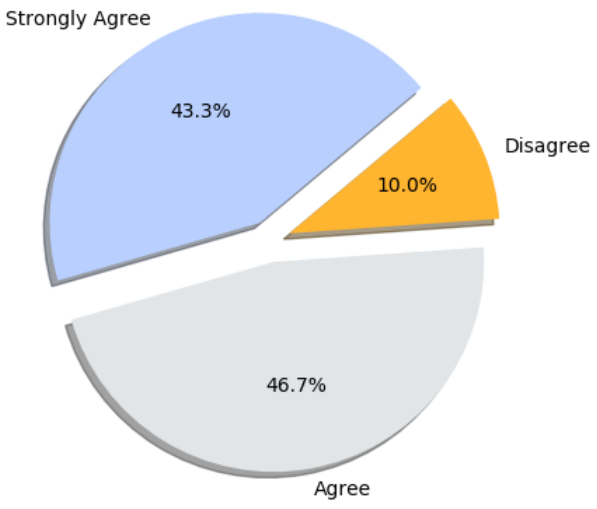


Figure 27: Application ease of use

(ii) Usefulness of OSS Recommendations from the Application

Users were asked to use the application and find the OSS that may be of use to them from the application. Out of 30 users that were involved in testing the application, 80.0% strongly agreed that the OSS recommendations are useful and 20.0% agreed that the OSS recommendations are useful. These results indicated that the OSS recommendations from the application are useful for HLIs in Tanzania. Figure 28 below shows the summary of the stated results

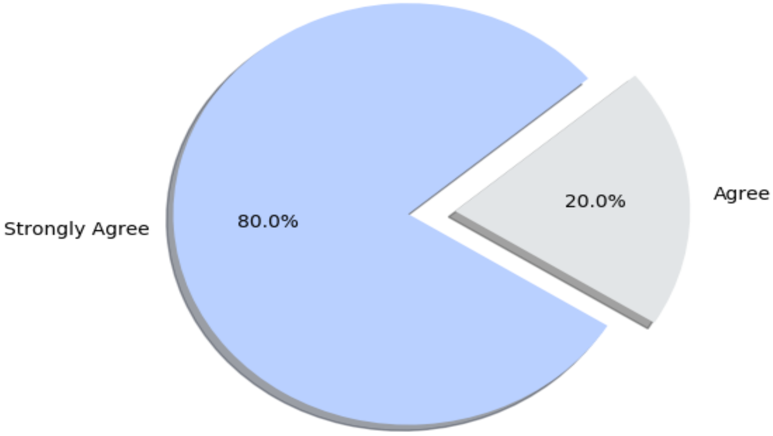


Figure 28: Usefulness of OSS recommendations from the application

(iii) Ease of Viewing and Downloading OSS

Users were asked to use the application. The users were tasked to view and download OSS of their choice. Out of 30 users that were involved in testing the application, 66.7% strongly agreed that it was easy to view and download OSS and 33.3% agreed that it was easy to view and download OSS. These results indicated that it is easy for users to view OSS in the application and they could easily download OSS of their choice. Figure 29 below shows the summary of the stated results.

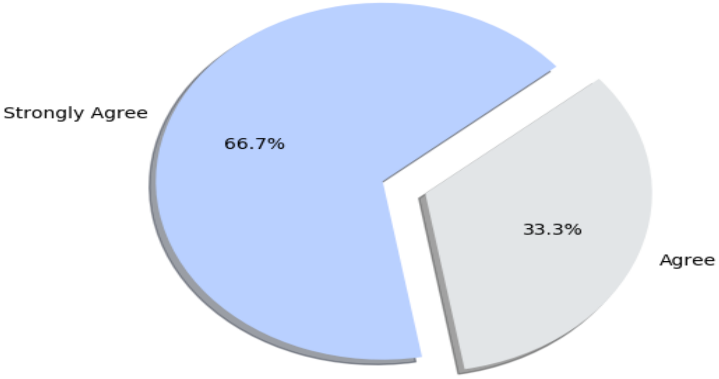


Figure 29: Easiness of viewing and downloading OSS from the application

(iv) Application Overall Performance

Users were asked to give their opinions on the level of satisfaction from the overall performance of the application. Out of 30 users that were involved in testing the application, 60.0% strongly agreed that they were satisfied with the overall application performance and 40.0% agreed that they were satisfied with the overall application performance. These results indicated that the overall application performance in accomplishing the recommendation of OSS to HLIs is good. Figure 30 below shows the summary of the stated results.

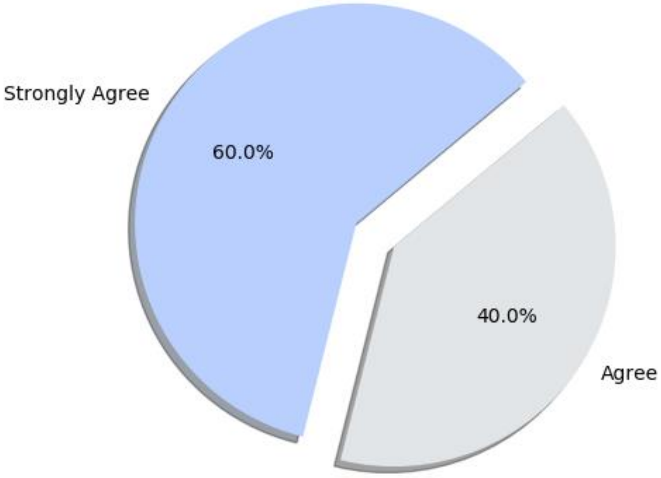


Figure 30: Users satisfaction on the application overall performance

The test was successful activity as the application was able to accomplish its intended purpose, most of the users understood the application concept. The application will provide more recommendation as more OSS are added to the database, students, academicians, system administrator, lab scientist and application administrator can easily add OSS to the database as long as they have a reliable Internet connection. Out of 30 users that were involved in testing the application, 50.0% strongly agreed that they will recommend the application to their colleagues and 50.0% agreed that they will recommend the application to their colleagues. These results indicate that most of the users valued the contribution of the application during the testing stage. They believe this application will help them to receive good OSS recommendation as most of the users appreciated the OSS recommendation they received during OSS, as they were good alternatives of PS that they obtained illegally. Figure 31 below shows a summary of the stated results.

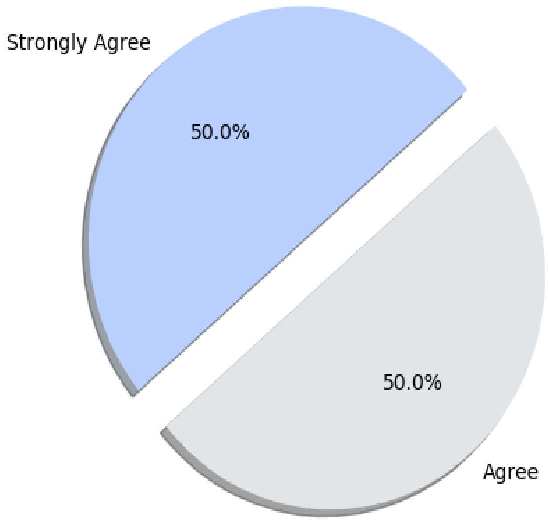


Figure 31: Users results if they will recommend this application to their colleagues.

4.7 Discussion

A number of OSS tools to be used for academic purposes were identified, majority of them fall in these areas of use; data analysis, computer science, electronics, telecommunications, software development, network administration, mathematics, document reader and writer, mobile application development, mechanical engineering, image editors, computer drawings, web browser, geographical information systems, accounting, language management and reference management systems. These areas of use are the basis of recommendations provided by the developed application. The number of OSS recommended to the user depends on the recommendation algorithms. Each recommendation algorithm provides advantages but also has disadvantages that are solved by another. For example, the demographic approach cannot provide recommendations of OSS that are viewed by other users or OSS that are in the same area of use as those active user has used in the past, but these disadvantages are achieved by collaborative filtering approach and content-based approaches respectively. Basically, OSS that could not fit in one algorithm criteria may fit in the other. Eventually, this provides a large number of OSS recommended to the active user. This study suggests that hybrid recommender systems are more efficient and reliable in providing recommendations to users than normal recommender systems.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

This study aimed at improving the adoption of OSS in Tanzanian HLIs through enhancement of individual awareness on the existing OSS designed for academic purposes using the recommendation approaches: content-based, demographic and collaborative filtering. This study, it nevertheless assessed the level of awareness of various people from HLIs on existing OSS that are designed for academic purposes and software that are currently used in HLIs. The study, besides these, it explored OSS that can be alternative to currently used software, and more useful OSS that can fit for educational purposes from OSS vendors and other online platforms. Many OSS designed for the academic purpose were found and categorized according to their academic area of use. Eventually, a web-based application for the recommendation of OSS was developed by considering the existing situation found during the survey in HLIs. The OSS designed for academic purposes that were found and categorized were uploaded to the application's database and served as a foundation of OSS recommendations. Additionally, the study reviewed and selected proper web technologies that were used for the implementation of application functionalities. The three recommendation algorithms were developed using web technologies to serve the purpose of recommending OSS designed for academic purposes to the HLIs.

The developed web-based application using the recommendation algorithms is able to; provide recommendations to both registered and unregistered users, send notification about new OSS recommendation through user's email, allow users to share OSS designed for academic purposes with another user through uploading function, allows the user to view and download the OSS and leave their feedbacks.

The ability to recommend OSS designed for academic purposes to its users and send notification through the user's email whenever new OSS is uploaded is a unique feature when compared to similar online platforms.

The developed web-based application can minimize the search time for OSS online, to minimize HLIs' cost since OSS are freely available and reduce cases of copyright infringements because most of the people in Tanzanian HLIs use cracked PS. We encourage the use of the web-based application for the cost-effective selection of software tools for HLIs.

5.2 Recommendations

(i) The HLIs should enforce the use of OSS in their curriculum and other setups this will reduce the HLIs cost as well as illegal issues. This will also smoothen adaptation of OSS since the students will familiarize themselves to those OSS thus this will be passed through generations and their future workplaces.

(ii) The lab scientist, academicians, and researchers should adopt using OSS in their lectures, seminar, workshop, and conferences as these are areas where people learn and it’s easy for them to adopt new technologies.

(iii) The HLIs should provide a conducive learning environment that can accelerate adoption of new technologies in which most of them are OSS, for example, software used for deep learning, machine learning etc. that includes python, R programming are open source.

5.3 Future Research Areas

The study offers some areas of extension to make the application more efficient and update on existing OSS designed for the academic purpose. The recommended area of extension is suggested to be integration with other websites that publish OSS such as SourceForge or Github, through this the application can be more update since once software is uploaded in that website they can be visible in the application and if the software fits for academic purpose it can be added to the application database for later recommendations. This was not done in this study due to the scope of the study and time limitations.

REFERENCES

Al-Badi, A. S. and Al-Badi, A. H. (2007). Open Source Software in Higher Education: Oman SQU Case Study. *Information Technologies in Education in the 21st Century*, *8*(8), 116–124. https://doi.org/10.1109/MITP.2010.120.

Al-Hajri, R., Al-Mukhaini, G. and Ramalingam, R. (2017). Adoption of Free and Open Source Software Using Alternative Educational Framework in the College of Applied Sciences. *Free and Open Source Software Conference*, 15–18.

Apache Software Foundation. (1995). Welcome! - The Apache HTTP Server Project. Retrieved September 14, 2018, from https://httpd.apache.org/.

Ayushi, S. and Prasad, V. R. B. (2018). Cross-Domain Recommendation Model based on Hybrid Approach. *International Journal of Modern Education and Computer Science, 10*(November), 36–42. https://doi.org/10.5815/ijmecs.2018.11.05.

Darie, C. and Balanescu, E. (2008). *E-Commerce Beginning PHP and MySQL* (Second Edi). United States of America: Apress.

Elmasri, R. and Navathe, S. B. (2004). *Fundamental of Database systems* (Fourth Edi). New York: Pearson Education, Inc.

Fariha, Z. and Zuriyati, A. (2014). Comparing Moodle and eFront Software for Learning Management System. *Australian Journal of Basic and Applied Sciences*, *8*(4), 158–162.

Haule J, M. (2015). Open Educational Resources Utilization in Tanzania Higher Learning Institutions. *Business Education Journal*, *1*(1), 1–28.

Ibennetch, N. (2018). phpMyAdmin download | SourceForge.net. Retrieved August 15, 2018, from https://sourceforge.net/projects/phpmyadmin/.

Jackson, J. C. (2007). *Web Technologies: a computer science perspective*. New Jersey: Pearson Education, Inc. Retrieved from http://dl.acm.org/citation.cfm?id=1197160.

John, L. and Sam, A. E. (2018). An Online Collaborative Discussion Platform for Bridging a Technological Reliance Gap in Higher Learning Institutions in Tanzania. *International Journal of Modern Education and Computer Science*, 29–35. https://doi.org/10.5815/ijmecs.2018.11.04.

Jquery. (2018). Retrieved August 4, 2018, from https://jquery.com/.

Lungo, J. H. and Kaasbol, J. J. (2007). Experiences of Open Source Software in Institutions : Cases From Tanzania and Norway. In *Proceedings of the 9th International Conference on Social Implications of Computers in Developing Countries*. São Paulo, Brazil.

Lwoga, E. (2012). Making learning and Web 2.0 technologies work for higher learning institutions in Africa. *Campus-Wide Information Systems*, *29*(2), 90–107. https://doi.org/10.1108/10650741211212359.

MoEST. (2015). Udahili wa Wanafunzi wa Vyuo Vikuu Ngazi ya Shahada na Isiyokuwa Shahada - Seti za data - Open Data Tanzania. Retrieved November 2, 2017, from http://opendata.go.tz/sw/dataset/udahili-wa-wanafunzi-wa-vyuo-vikuu-ngazi-ya-shahada-na-isiyokuwa-shahada-2015-2016.

Ngrok - secure introspectable tunnels to localhost. (2018). Retrieved August 24, 2018, from https://ngrok.com/product.

Nixon, R. (2012). *Learning PHP, MySQL, JavaScript, and CSS* (2nd Editio). Sebastopol: O’Reilly Media Inc.

Odell, D. (2009). *Pro JavaScript RIA Techniques*. New York: Apress.

Ola and Markus. (2018). AlternativeTo - Crowdsourced software recommendations. Retrieved May 12, 2018, from https://alternativeto.net/.

Schafer, B. J., Konstan, J. A., & Riedl, J. (2001). E-Commerce Recommendation Applications. *Kluwer Academic Publishers*, 5, 115–153.

Sife, A. S., Lwoga, E. T. and Sanga, C. (2007). New technologies for teaching and learning: Challenges for higher learning institutions in developing countries. *International Journal of Education and Development Using Information and Communication Technology*, 57–67. Retrieved from http://ijedict.dec.uwi.edu/viewarticle.php?id=246/andlayout=html.

Singh, Y. K. (2006). *Fundamental of Research Methodology and Statistics*. *Methodology Research* (Vol. 53). New Delhi: New Age International (P) Limited, Publishers. https://doi.org/10.1017/CBO9781107415324.004.

Slashdot Media. (2018). SourceForge - Download, Develop and Publish Free Open Source

Software. Retrieved May 12, 2018, from https://sourceforge.net/.

Sommerville, I. (2011). *Software Engineering*. https://doi.org/10.1111/j.1365-2362.2005. 01463 .x.

TCRA. (2018). *Quarterly Communications Statistics*. Dar es Salaam. Retrieved from https://www.tcra.go.tz/images/documents/telecommunication/TelCom_Statistics_December_2018.pdf.

Valacich, J. S., George, J. F. and Hoffer, J. A. (2012). *Essentials of Systems Analysis and Design*. (S. Yagan, E. Svendsen, B. Horan, and A. Bradbury, Eds.) (Fifth). New Jersey: Pearson Education, Inc.

APPENDICES

Appendix 1: Survey Questionnaire for Academicians

Questionnaire on Awareness, Usage, and Attitudes of Higher Learning Institutions Community on Open Source Software.

Section 1: General Information

QN. 1	Name of the Higher Learning Institution	
QN. 2	Gender of the respondent	[ ] Male [ ] Female
QN. 3	What is your age group?	[ ] 16 to 22 [ ] 23 to 29 [ ] 30 to 36 [ ] 37 to 43 [ ] 44 or above
QN. 4	What is your current rank as an Academician?	[ ] Tutorial Assistant [ ] Assistant Lecturer [ ] Lecturer [ ] Senior Lecturer [ ] Professor [ ] Assistant Librarian [ ] Librarian [ ] Senior Librarian [ ] Librarian Professor [ ] Others: Specify …………………………
QN. 5	What is your area of specialization?	[ ] Art, Social Sciences, Music, and Sport Science [ ] Engineering [ ] Language and Cultural Studies [ ] Math/Science [ ] Veterinary medicine/Agrar-, Forest/Nutritional Sciences, Ecology [ ] Agriculture [ ] Human Medicine [ ] Others: Specify ……………………………….
QN. 6	What type of software applications (operating system, special purpose and generic) are you using the most for academic purposes?	[ ] Proprietary Software [ ] Open Source Software [ ] Freeware [ ] Free Software [ ] I do not use software applications
QN. 7	If you use proprietary software the most; How do you always get your copy of the software?	[ ] I Purchased Licence [ ] From my institution [ ] I got cracked version [ ] On trial version

Section 2: End users software applications usage

		Name of the application	**Category (e.g. Social, Business, Technical, Administration, Communications, Text Editor, etc.)**	**Supported Platforms**
QN. 8	If you use software applications; Which end-user software applications do you use?			[ ] Windows [ ] MAC OS [ ] Linux
				[ ] Windows [ ] MAC OS [ ] Linux
				[ ] Windows [ ] MAC OS [ ] Linux
				[ ] Windows [ ] MAC OS [ ] Linux
				[ ] Windows [ ] MAC OS [ ] Linux
				[ ] Windows [ ] MAC OS [ ] Linux
				[ ] Windows [ ] MAC OS [ ] Linux
QN. 9	If you were to recommend the usage of open source software for use in your institution, which open source software would you have recommended and why?			

Section 3: Awareness of Open Source Software.

QN. 10	Do you know any open source software applications that are designed for academic purposes?	[ ] Yes [ ] No
QN. 11	If your answer in QN. 10 is Yes; Mention them	
QN. 12	Do you know any online sites that you can get open source software applications for academic purpose?	[ ] Yes [ ] No
QN. 13	If yes in QN. 12 above; Mention the online sites	
QN. 14	If your answer in QN. 10 is no; why you are not aware of any open source software applications that are designed for academic purposes?	[ ] I don't know what are they [ ] I don't know if there any those designed for my subject group. [ ] I do not know which online sites to visit. [ ] They are not well promoted. [ ] Others: Specify ……………………………….

Section 4: Usage of Open Source Software.

QN. 15	Do you use any open source software applications for academic purposes?	[ ] Yes [ ] No
QN. 16	If no; why you are not using any open source software applications for academic purposes?	[ ] I do not know where to download them [ ] Online website open other pages when I want to download that I fail to download. [ ] Others: Specify ………………………………………
QN. 17	If yes; how did you get the copy of open source software applications you are using?	[ ] From a colleague or friend [ ] From a website [ ] Other: Specify ………………………………
QN. 18	Is/Are that/those method(s) you mentioned in QN. 17 above reliable whenever you need to get a copy of any open source software applications?	[ ] Yes [ ] No

QN. 19	What are the challenges of the method you mentioned in QN. 17 above?	
QN. 20	If you get open source software applications from online sites, what is your experience until you get your copy	[ ] I spend a lot of time to get it [ ] Website redirect to many other links [ ] It is difficult to search for software I want [ ] Others: Specify ………………………………..
QN. 21	Mention open source software applications that you are using for academic purposes?	
QN. 22	What type of notification would you like to receive about useful open source software applications that are designed for academic purposes?	[ ] Email [ ] SMS [ ] Others: Specify

Section 5: Attitude on Open Source Software.

QN. 23	Open source software applications are important tools for teaching and learning in higher learning institutions	[ ] Strongly Agree [ ] Agree [ ] Disagree [ ] Strongly Disagree
QN. 24	New technologies including open source software applications have improved efficiency in academic activities in higher learning institutions	[ ] Strongly Agree [ ] Agree [ ] Disagree [ ] Strongly Disagree
QN. 25	If there will be a web-based application that will help the recommendation of open source software applications that are designed for academic purposes will be helpful	[ ] Strongly Agree [ ] Agree [ ] Disagree [ ] Strongly Disagree
QN. 26	What are the features you would like to see in the web-based application for the recommendation of open source software as tools for higher learning institutions	

Appendix 2: Survey Questionnaire for Students

Questionnaire on Awareness, Usage, and Attitudes of Higher Learning Institutions Community on Open Source Software.

Section 1: General Information

QN. 1	Name of the Higher Learning Institution	
QN. 2	Gender of the respondent	[ ] Male [ ] Female
QN. 3	What is your age group?	[ ] 16 to 22 [ ] 23 to 29 [ ] 30 to 36 [ ] 37 to 43 [ ] 44 or above
QN. 4	What is the current level of education that you are studying?	[ ] Basic Certificate [ ] Certificate [ ] Ordinary Diploma [ ] Advanced/Higher Diploma [ ] Bachelor [ ] Post Graduate Diploma [ ] Masters [ ] Doctorate [ ] Others: Specify …………………………..
QN. 5	What is your area of specialization?	[ ] Art, Social Sciences, Music, and Sport Science [ ] Engineering [ ] Language and Cultural Studies [ ] Math/Science [ ] Veterinary medicine/Agrar-, Forest/Nutritional Sciences, Ecology [ ] Agriculture [ ] Human Medicine [ ] Others: Specify ………………………………
QN. 6	What type of software applications (operating system, special purpose and	[ ] Proprietary Software [ ] Open Source Software

	generic) are you using the most for academic purposes?	[ ] Freeware [ ] Free Software [ ] I do not use software applications
QN. 7	If you use proprietary software the most; How do you always get your copy of the software?	[ ] I Purchased Licence [ ] From my institution [ ] I got cracked version [ ] On trial version [ ] Others: Specify ………………………………..

Section 2: End users software applications usage

		Name of the application	Category (e.g. Social, Business, Technical, Administration, Communications, Text Editor, etc.)	Supported Platforms
QN. 8	If you use software applications; What are the most end-user software applications that you are using?			[ ] Windows [ ] MAC OS [ ] Linux
				[ ] Windows [ ] MAC OS [ ] Linux
				[ ] Windows [ ] MAC OS [ ] Linux
				[ ] Windows [ ] MAC OS [ ] Linux
				[ ] Windows [ ] MAC OS [ ] Linux
QN. 9	If you were to recommend open source software for use in your institution, which open source software would you have recommended and why?			

Section 3: Awareness of Open Source Software.

QN. 10	Do you know any open source software applications that are designed for academic purposes?	[ ] Yes [ ] No
QN. 11	If your answer in QN. 10 is Yes; Mention them	
QN. 12	Do you know any online sites that you can get open source software applications for academic purpose?	[ ] Yes [ ] No
QN. 13	If yes in QN. 12 above; Mention the online sites	
QN. 14	If your answer in QN. 10 is no; why you are not aware of any open source software applications that are designed for academic purposes?	[ ] I don't know what are they [ ] I don't know if there any those designed for my subject group. [ ] I do not know which online sites to visit. [ ] They are not well promoted. [ ] Others: Specify ………………………………..

Section 4: Usage of Open Source Software.

QN. 15	Do you use any open source software applications for academic purposes?	[ ] Yes [ ] No
QN. 16	If no; why you are not using any open source software applications for academic purposes?	[ ] I do not know where to download them [ ] Online website open other pages when I want to download that I fail to download. [ ] Others: Specify …………………….
QN. 17	If yes; how did you get the copy of open source software applications you are using?	[ ] From a colleague or friend [ ] From a website [ ] Other: Specify ………………………………
QN. 18	Is/Are that/those method(s) you mentioned in QN. 17 above reliable whenever you need to get a copy of any open source software applications?	[ ] Yes [ ] No
QN. 19	What are the challenges of the method you mentioned in QN. 17 above?	

QN. 20	If you get open source software applications from online sites, what is your experience until you get your copy	[ ] I spend a lot of time to get it [ ] Website redirect to many other links [ ] It is difficult to search for software I want [ ] Others: Specify ………………………………..
QN. 21	Mention open source software applications that you are using for academic purposes?	
QN. 22	What type of notification would you like to receive about useful open source software applications that are designed for academic purposes?	[ ] Email [ ] SMS [ ] Others: Specify

Section 5: Attitude on Open Source Software

QN. 23	Open source software applications are important tools for teaching and learning in higher learning institutions	[ ] Strongly Agree [ ] Agree [ ] Disagree [ ] Strongly Disagree
QN. 24	New technologies including open source software applications have improved efficiency in academic activities in higher learning institutions	[ ] Strongly Agree [ ] Agree [ ] Disagree [ ] Strongly Disagree
QN. 25	If there will be a web-based application that will help the recommendation of open source software applications that are designed for academic purposes will be helpful	[ ] Strongly Agree [ ] Agree [ ] Disagree [ ] Strongly Disagree
QN. 26	What are the features you would like to see in the web-based application for the recommendation of open source software as tools for higher learning institutions	

Appendix 3: Survey Questionnaire for System Administrators and Lab Engineers

Questionnaire on Awareness, Usage, and Attitudes of Higher Learning Institutions Community on Open Source Software.

Section 1: General Information

QN. 1	Name of the Higher Learning Institution	
QN. 2	Gender of the respondent	[ ] Male [ ] Female
QN. 3	What is your age group?	[ ] 16 to 22 [ ] 23 to 29 [ ] 30 to 36 [ ] 37 to 43 [ ] 44 or above
QN. 4	If you advise them on proper software to use/Provide them with software installation; What are the type of those software?	[ ] Proprietary [ ] Open Source Software [ ] Freeware [ ] Free Software
QN. 5	Where do you find information about those software?	[ ] From other institutions [ ] From a website [ ] Other: Specify
QN. 6	What type of software that are installed in the computer laboratory of your institution?	[ ] Proprietary [ ] Open Source Software [ ] Freeware [ ] Free Software [ ] No computer Lab in my institution
QN. 7	What are the challenges in adopting new technologies of teaching and learning in your institution?	[ ] Lack of awareness on those new technologies. [ ] Adaptation cost [ ] Poor Information and Communication Technologies Infrastructure [ ] Others; Specify ……………………………………
QN. 8	What Open Source Software does your Institution use?	

QN. 9	If there will be a web-based application that will help the recommendation of open source software that are designed for academic purposes will be helpful	[ ] Strongly Agree [ ] Agree [ ] Disagree [ ] Strongly Disagree
QN. 10	What are the features you would like to see in the web-based application for the recommendation of open source software as tools for higher learning institutions	

Section 2: Software applications usage

		Name of the application	**Category (e.g. Social, Business, Technical, Administration, Communications, Text Editor, etc.)**	**Supported Platforms**
QN. 11	If you use software applications; What are the most server software applications that you are using?			[ ] Windows [ ] MAC OS [ ] Linux
				[ ] Windows [ ] MAC OS [ ] Linux
				[ ] Windows [ ] MAC OS [ ] Linux
				[ ] Windows [ ] MAC OS [ ] Linux
				[ ] Windows [ ] MAC OS [ ] Linux
				[ ] Windows [ ] MAC OS [ ] Linux
QN.12	If you were to recommend the usage of open source software for use in your institution, which open source software would you have recommended and why?			

Appendix 4: Application Evaluation Questionnaire

This questionnaire has been designed for getting users' feedback on how the developed web-based application for the recommendation of open source software meets user's expectation.

QN. 1	It was easy to learn and use the application	[ ] Strongly Agree [ ] Agree [ ] Disagree [ ] Strongly Disagree
QN. 2	I am satisfied interacting with the application using the designed Graphical User Interfaces	[ ] Strongly Agree [ ] Agree [ ] Disagree [ ] Strongly Disagree
QN. 3	The recommendations of Open Source Software from the application are useful	[ ] Strongly Agree [ ] Agree [ ] Disagree [ ] Strongly Disagree
QN. 4	It was easy to view and download Open Source Software	[ ] Strongly Agree [ ] Agree [ ] Disagree [ ] Strongly Disagree
QN. 5	I will recommend this application to my colleagues	[ ] Strongly Agree [ ] Agree [ ] Disagree [ ] Strongly Disagree
QN. 6	I am satisfied with the overall application performance	[ ] Strongly Agree [ ] Agree [ ] Disagree [ ] Strongly Disagree

Appendix 5: Analysis done using Python

```
from collections import Counter
import matplotlib.pyplot as plt
import pandas as pd
mydata = pd.read_csv("C:/Users/Ambokile/Dropbox/MASTERS 2018/Research
Work/dissertation/Data/final data/researchdatafinalforpaper.csv")
gender=mydata["Gender of the respondent"]
gendercount= Counter(gender)
df=pd.DataFrame.from_dict(gendercount, orient='index')
df.plot(kind='bar')
awarenes  = mydata["Do you know any open source software applications that is designed for
academic purposes?"]
labels=['No','Yes']
awarenescount= Counter(awarenes)
df=pd.DataFrame.from_dict(awarenescount, orient='index')
values=[df.loc['No'],df.loc['Yes']]
colors=['#FFB52F','#7578FF']
explode=(0.1,0.1)
plt.pie(values, explode=explode, labels=labels,
colors=colors,autopct='%1.1f%%',shadow=True, startangle=140)
plt.hist(df)
plt.axis('equal')
plt.show()
softwareusage  = mydata["What type of software applications (operating system, special
purpose and generic) are you using the most for academic purposes?"]
labels=['Proprietary Software','Freeware','Open Source Software','Free Software','Not using']
softwareusagecount= Counter(softwareusage)
df=pd.DataFrame.from_dict(softwareusagecount, orient='index')
print(df)
values=[df.loc['Proprietary Software'],df.loc['Freeware'],df.loc['Open Source
Software'],df.loc['Free Software'],df.loc['I do not use any software']]
colors=['lightskyblue','#C4FFB4','#7578FF','cyan','#FFB52F']
```

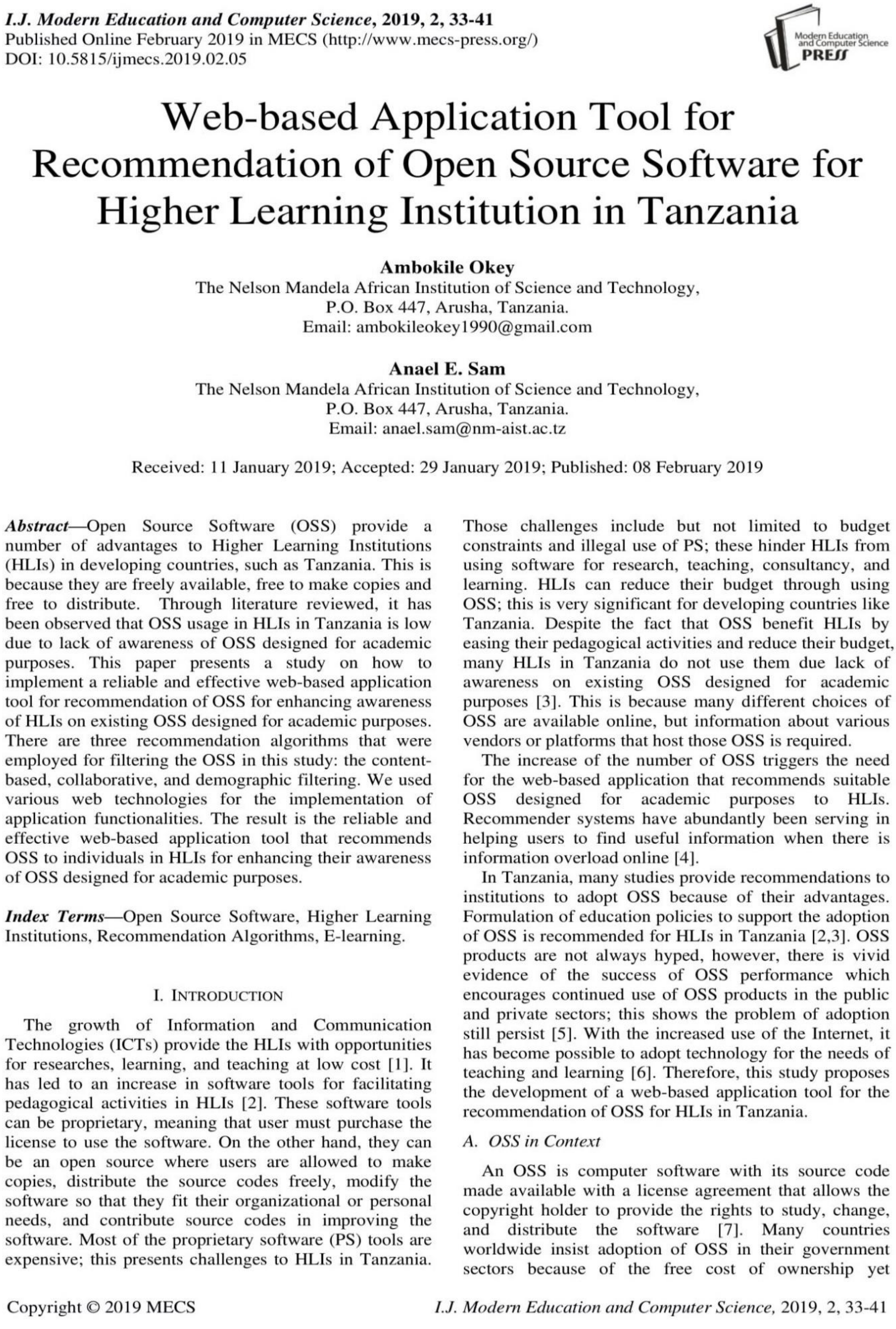
```
explode=(0.1,0.1,0.1,0.1,0.1)
plt.pie(values, explode=explode, labels=labels,
colors=colors,autopct='%1.1f%%',shadow=True, startangle=40)
plt.pie(df)
plt.axis('equal')
plt.show()
softwarelegality  = mydata["If you use proprietary software the most; How do you always get
your copy of software?"]
abels=['From their institution','Get illegal copies','Get legal licence','Use trial versions']
softwarelegalityecount= Counter(softwarelegality)
df=pd.DataFrame.from_dict(softwarelegalityecount, orient='index')
print(df)
values=[df.loc['From my institution'],df.loc['I got cracked version'],df.loc['I Purchased
Licence'],df.loc['On trial version']]
colors=['#E7FF06','#BB8DFF','#7578FF','#08DFFF']
explode=(0.1,0.1,0.1,0.1)
plt.pie(values, explode=explode, labels=labels,
colors=colors,autopct='%1.1f%%',shadow=True, startangle=40)
plt.pie(df)
plt.axis('equal')
plt.show()
userperspective  = mydata["If there will be a web-based application that will help
recommendation of  open source software applications that are designed for academic
purposes will be helpful"]
labels=['Strongly Agree','Agree','Disagree','Strongly Disagree']
userperspectivecount= Counter(userperspective)
df=pd.DataFrame.from_dict(userperspectivecount, orient='index')
print(df)
values=[df.loc['Strongly Agree'],df.loc['Agree'],df.loc['Disagree'],df.loc['Strongly Disagree']]
colors=['#B9D0FF','#E2E5E7','#FFB52F','#08DFFF']
explode=(0.1,0.1,0.1,0.1)
```

```
plt.pie(values, explode=explode, labels=labels,
colors=colors,autopct='%1.1f%%',shadow=True, startangle=40)
plt.axis('equal')
plt.show()
from collections import Counter
import matplotlib.pyplot as plt
import pandas as pd
mydata = pd.read_csv("C:/Users/Ambokile/Dropbox/MASTERS 2018/Research
Work/dissertation/Data/final data/Evaluation/Application Evaluation.csv")
easyofuse  = mydata["It was easy to learn and use the application"]
labels=['Strongly Agree','Agree','Disagree']
easyofusecount= Counter(easyofuse)
df=pd.DataFrame.from_dict(easyofusecount, orient='index')
print(df)
values=[df.loc['Strongly Agree'],df.loc['Agree'],df.loc['Disagree']]
colors=['#B9D0FF','#E2E5E7','#FFB52F','#08DFFF']
explode=(0.1,0.1,0.1)
plt.pie(values, explode=explode, labels=labels,
colors=colors,autopct='%1.1f%%',shadow=True, startangle=40)
plt.axis('equal')
plt.show()
usefulnessRec  = mydata["The recommendations of Open Source Software from the
application are useful"]
labels=['Strongly Agree','Agree']
usefulnessReccount= Counter(usefulnessRec)
df=pd.DataFrame.from_dict(usefulnessReccount, orient='index')
print(df)
values=[df.loc['Strongly Agree'],df.loc['Agree']]
colors=['#B9D0FF','#E2E5E7','#FFB52F','#08DFFF']
explode=(0.1,0.1)
plt.pie(values, explode=explode, labels=labels,
colors=colors,autopct='%1.1f%%',shadow=True, startangle=40)
```

```
plt.axis('equal')
plt.show()
easevd  = mydata["It was easy to view and download Open Source Software"]
labels=['Strongly Agree','Agree']
easevdCount= Counter(easevd)
df=pd.DataFrame.from_dict(easevdCount, orient='index')
print(df)
values=[df.loc['Strongly Agree'],df.loc['Agree']]
colors=['#B9D0FF','#E2E5E7','#FFB52F','#08DFFF']
explode=(0.1,0.1)
plt.pie(values, explode=explode, labels=labels,
colors=colors,autopct='%1.1f%%',shadow=True, startangle=40)
plt.axis('equal')
plt.show()
appPerformance  = mydata["I am satisfied with the overall application performance"]
labels=['Strongly Agree','Agree']
appPerformanceCount= Counter(appPerformance)
df=pd.DataFrame.from_dict(appPerformanceCount, orient='index')
print(df)
values=[df.loc['Strongly Agree'],df.loc['Agree']]
colors=['#B9D0FF','#E2E5E7','#FFB52F','#08DFFF']
explode=(0.1,0.1)
plt.pie(values, explode=explode, labels=labels,
colors=colors,autopct='%1.1f%%',shadow=True, startangle=40)
plt.axis('equal')
plt.show()
appRecommend  = mydata["I will recommend this application to my colleagues"]
labels=['Strongly Agree','Agree']
appRecommendCount= Counter(appRecommend)
df=pd.DataFrame.from_dict(appRecommendCount, orient='index')
print(df)
values=[df.loc['Strongly Agree'],df.loc['Agree']]
```

```
colors=['#B9D0FF','#E2E5E7','#FFB52F','#08DFFF']
explode=(0.1,0.1)
plt.pie(values, explode=explode, labels=labels,
colors=colors,autopct='%1.1f%%',shadow=True, startangle=40)
plt.axis('equal')
plt.show()
```

RESEARCH OUTPUTS



I.J. Modern Education and Computer Science, **2019, 2, 33-41**
Published Online February 2019 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijmecs.2019.02.05

Web-based Application Tool for Recommendation of Open Source Software for Higher Learning Institution in Tanzania

Ambokile Okey
The Nelson Mandela African Institution of Science and Technology,
P.O. Box 447, Arusha, Tanzania.
Email: ambokileokey1990@gmail.com

Anael E. Sam
The Nelson Mandela African Institution of Science and Technology,
P.O. Box 447, Arusha, Tanzania.
Email: anael.sam@nm-aist.ac.tz

Received: 11 January 2019; Accepted: 29 January 2019; Published: 08 February 2019

Abstract—Open Source Software (OSS) provide a number of advantages to Higher Learning Institutions (HLIs) in developing countries, such as Tanzania. This is because they are freely available, free to make copies and free to distribute. Through literature reviewed, it has been observed that OSS usage in HLIs in Tanzania is low due to lack of awareness of OSS designed for academic purposes. This paper presents a study on how to implement a reliable and effective web-based application tool for recommendation of OSS for enhancing awareness of HLIs on existing OSS designed for academic purposes. There are three recommendation algorithms that were employed for filtering the OSS in this study: the content-based, collaborative, and demographic filtering. We used various web technologies for the implementation of application functionalities. The result is the reliable and effective web-based application tool that recommends OSS to individuals in HLIs for enhancing their awareness of OSS designed for academic purposes.

Index Terms—Open Source Software, Higher Learning Institutions, Recommendation Algorithms, E-learning.

I. Introduction

The growth of Information and Communication Technologies (ICTs) provide the HLIs with opportunities for researches, learning, and teaching at low cost [1]. It has led to an increase in software tools for facilitating pedagogical activities in HLIs [2]. These software tools can be proprietary, meaning that user must purchase the license to use the software. On the other hand, they can be an open source where users are allowed to make copies, distribute the source codes freely, modify the software so that they fit their organizational or personal needs, and contribute source codes in improving the software. Most of the proprietary software (PS) tools are expensive; this presents challenges to HLIs in Tanzania. Those challenges include but not limited to budget constraints and illegal use of PS; these hinder HLIs from using software for research, teaching, consultancy, and learning. HLIs can reduce their budget through using OSS; this is very significant for developing countries like Tanzania. Despite the fact that OSS benefit HLIs by easing their pedagogical activities and reduce their budget, many HLIs in Tanzania do not use them due lack of awareness on existing OSS designed for academic purposes [3]. This is because many different choices of OSS are available online, but information about various vendors or platforms that host those OSS is required.

The increase of the number of OSS triggers the need for the web-based application that recommends suitable OSS designed for academic purposes to HLIs. Recommender systems have abundantly been serving in helping users to find useful information when there is information overload online [4].

In Tanzania, many studies provide recommendations to institutions to adopt OSS because of their advantages. Formulation of education policies to support the adoption of OSS is recommended for HLIs in Tanzania [2,3]. OSS products are not always hyped, however, there is vivid evidence of the success of OSS performance which encourages continued use of OSS products in the public and private sectors; this shows the problem of adoption still persist [5]. With the increased use of the Internet, it has become possible to adopt technology for the needs of teaching and learning [6]. Therefore, this study proposes the development of a web-based application tool for the recommendation of OSS for HLIs in Tanzania.

A. OSS in Context

An OSS is computer software with its source code made available with a license agreement that allows the copyright holder to provide the rights to study, change, and distribute the software [7]. Many countries worldwide insist adoption of OSS in their government sectors because of the free cost of ownership yet providing a high level of security, good performance, and reliable service. For instance, Russia makes the use of OSS mandatory in its entire sectors meanwhile others are in the process of adoption [1]. OSS offer several benefits to HLIs as it can be customized to suit the organizational needs. Thus, lecturers and students are no longer forced to use illegal copies of software tools. Eventually, using OSS can contribute to a reduction in running costs, these core benefits are whys for the need and adaption of OSS [7]. OSS in HLIs lies into various categories but not limited to: learning management tools, content management tools, and collaboration tools. Various HLIs Worldwide have adopted OSS for various applications such as in course management and electronic portfolios due to several advantages OSS have over PS [5].

OSS perform a vital role in the success of many HLIs in the world, but many HLIs face a number of challenges. For instance, many people in Oman Sultan Qaboos University are not aware of OSS, leading to its low usage [7].

Some of OSS provide e-learning platforms, the learning management system (LMS) such as Moodle. LMS provides HLIs with a centralized set of tools to help the HLIs succeed in various aspects of curricula such as managing course catalogues, record data from learners, and providing reports to management [8]. In today's education various OSS play a great role in assisting people in HLIs to carry out their tasks, to mention few: Latex for authoring research papers, python for data analysis, android studio for mobile application development, and QGIS for creating, editing, analyzing and publishing geographical information.

B. E-Learning Technologies Adoption Challenges in HLIs in Tanzania

E-Learning refers to the use of ICTs to enhance and support teaching, learning, and research. This has transformed a traditional way of doing pedagogical activities. E-learning technologies can be in hardware form (Televisions, Digital Versatile Disks (DVDs), Radio) or in software form (E-learning platforms, web-based technologies, software applications) [2]. Software e-learning technologies can be proprietary or open source. Albeit the significance of E-learning in education and research its adoption in Tanzanian HLIs though at an increasing rate is apparently low [5, 9].

Lack of enough financial resource to budget for e-learning tools is a challenge to many HLIs in Tanzania, this is due to the fact that the Government of Tanzania has reduced public funds [10].

OSS tools such as Moodle require Internet infrastructure to operate but other OSS tools such as Latex do not require Internet to operate. There are various challenges faced in utilizing the OSS because as OSS differ in characteristics, so the challenges. Few of OSS tools require the organizational support of ICT infrastructures whilst many do not. However, the ICT infrastructures support is no longer a challenge due to the high rate of growth of Internet connectivity in Tanzania where most of the telecommunication companies offer up to 4G service to their subscribers [11]. Several OSS tools addressed in this study do not require Internet for them to operate, rather they solely require it during downloading and updating.

Despite the benefits provided by OSS tools, and ICT infrastructure provided by the government of Tanzania, the implementation of OSS tools in HLIs is still low due to a number challenges: lack of systemic approach to ICT implementation, awareness and attitude towards ICTs, administrative support, technical support, staff development, lack of ownership, and inadequate funds [2]. Nevertheless, most of these challenges are not common to all OSS tools, they are most relevant to e-learning platforms as they require the involvement of the entire organization.

Generally, the most commonly observed challenge that slows down the adoption process is the knowledge gap between students, academics, system administrators, and lab scientists over the existing e-learning technologies [3].

C. Recommender Systems Concept

A recommender system is a software tool and technique providing suggestions for items that may be of use to the user. Users during their online sessions encounter information overload. The recommender systems enrich their decisions on available information. Recommender systems are designed to help individuals who lack sufficient personal experience to evaluate a large number of items that an online platform may provide [4].

There are different classes of recommendation approaches:

1) Content-based: The system learns to recommend items that are similar to ones that user liked in the past. The similarity of items is calculated based on the features associated with compared items.
2) Collaborative filtering: The system recommends to the active user the items that other users with similar taste liked in the past. The similarity in the taste of two users is calculated based on the similarity in the rating history of the users.
3) Demographic: The system recommends items to a user based on their demographic profiles such as age, education, gender etc.
4) Knowledge-based: The system recommends items based on specific domain knowledge about certain item features meet user needs and how that item can be useful for the user. In this system, a similarity function estimates how much the user needs to match the recommendations.
5) Community-based: The system recommends items based on the preferences of users friends.
6) Hybrid recommender system: This system is based on a combination of different techniques mentioned above. This system tries to use the advantages of one technique to fix the disadvantage of another technique.

This study presents how the web-based application tool for the recommendation of OSS for HLIs is developed using demographic, content-based, and collaborative filtering recommendation techniques. The viability of the developed application would be: to enhance individual awareness in HLIs on existing OSS designed for academic purposes, to improve usage of OSS in HLIs, to improve quality of pedagogical activities through improving usage of OSS designed for academic purposes, and reduce software illegality issues because many people in HLIs in Tanzania are using cracked PS.

The rest of this paper is organized as follows: the second section presents related works. The third section describes the methods used for requirement gathering and development of the web-based application. The design of the web-based application that includes decision-tree of the recommendation process is present in section four. Key functionalities of the web-based application are described in section five. The sixth section presents the results of the implementation of the design and key functionalities. Discussion of results obtained is done in section seven. Finally, the conclusion has been drawn in section eight.

II. Related Works

This section of the paper presents several approaches that have been done to promote and facilitate the adoption of OSS among communities.

A. Alternative Educational Framework

Alternative education framework which is referred to as a research methodology to identify free and open source software (FOSS); colleges of applied sciences in different regions of the sultanate in Oman used this framework. It is aimed at identifying a list of FOSS to replace PS used in courses of a bachelor degree in information technology (BIT) [1]. This framework presented a list of FOSS for use in BIT, but the list was in a paper-based which may face challenges like storage and ease of access and it was for a single educational area of specialization. In consequence, it needed improvement so that it can be accessed by many people without any restriction, and also includes more academic areas of specialization.

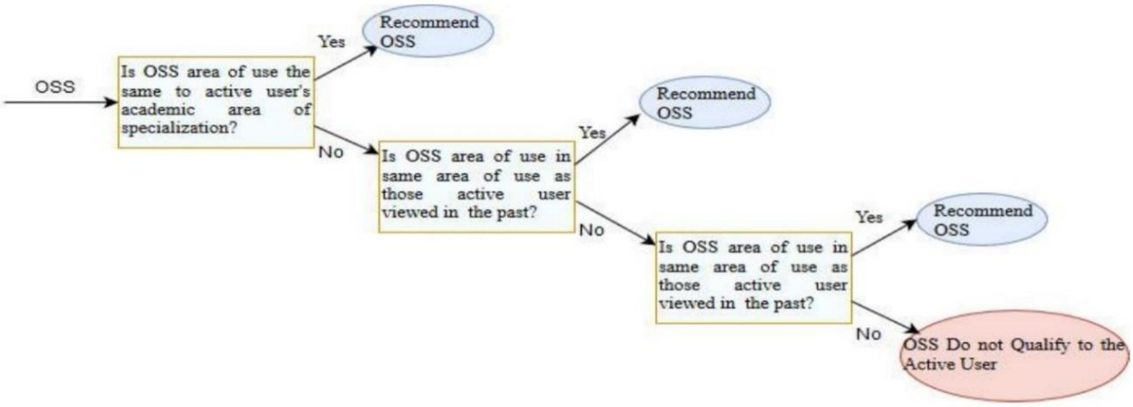

OSS
Is OSS area of use the same to active user's academic area of specialization?
Yes
Recommend OSS
No
Is OSS area of use in same area of use as those active user viewed in the past?
Yes
Recommend OSS
No
Is OSS area of use in same area of use as those active user viewed in the past?
Yes
Recommend OSS
No
OSS Do not Qualify to the Active User

Fig.1. Decision Tree of the Recommendation Algorithms

B. Sourceforge Website

Sourceforge is a website for publishing open source projects (OSPs). It allows the community to share OSPs by contributing to its improvement. Despite having many projects, Sourceforge treats educational software as a single category, thus there are no recommendations of OSS based on its usefulness to particular pedagogical activity or as an alternative to certain PS to stakeholders in HLIs [12].

C. Alternativeto Website

Alternativeto.net is a website that provides alternative recommendations to software. It does not allow direct download rather provide a link to an official website of the software, also like source forge as it treats educational software as a large category as a consequence there are no recommendations of OSS based on its usefulness to particular pedagogical activity or stakeholders in HLIs [13].

III. Methodology

This section of the paper presents the methods used to establish the requirements and development of the web-based application.

A. Requirement Gathering and Analysis

We used both qualitative and quantitative research methods for requirements gathering. The requirements were analysed and it appeared that majority of the respondents wanted to get recommendations of useful OSS designed for academic purposes, to be able to view and download OSS and to receive email notification about OSS recommendations. However, some requirements were not very clear which led to the application of evolutionary prototyping as it allows getting users' feedback during development.

B. Web-based Application Development

Prototyping is a process of designing and building a scaled-down but working version of the desired system [14]. In this study, evolutionary prototyping was employed, and basic requirements were gathered and analysed. Then the prototype was built and it was improved based on users' feedback. This process was iterative until we got the final version that users were satisfied with.

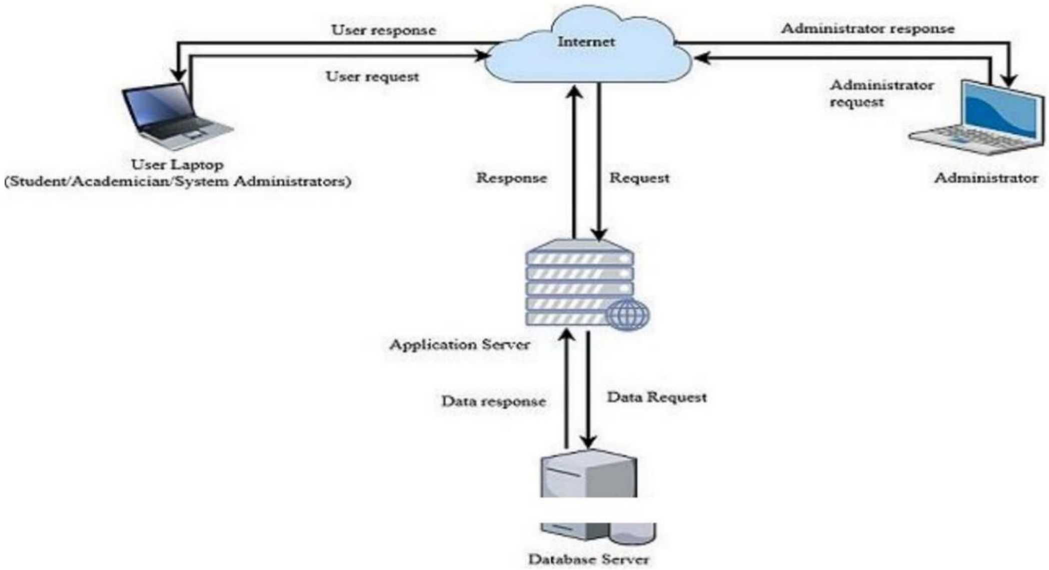


Fig.2. Architecture of the Web-based Application

IV. Design of the Web-Based Application

In this section, the design of the web-based application is conducted. It presents a structure of the recommendation algorithms, the architecture of the web-based application and tools used for development.

A. Web-based Application Architecture

The application runs on a three tiers architecture as shown in fig.2. The application is accessed through the Internet. The users and OSS data are stored on the database server, the application's functions and other application's rules and procedures files are stored in the application server. This ensures more security as the application server checks the request to the database before forwarding to the database server.

B. Structure of the Recommendation Algorithms

In this subsection, the structure of how the recommendation process works is presented using the decision tree. The application checks if the uploaded OSS academic area of use is the similar to active user's academic area of specialization, or if it is in the same area of use as those the active user viewed in the past, or if it viewed by other users that share academic area of specialization with the active users. The summary of the decisions is shown in fig.1.

C. Tools used for Development

This subsection of the paper presents web technologies used to implement the application and various server configurations. Web technologies are the collection of technologies that were developed as part of the birth of the world wide web [15]. They are the foundation of the development of web applications.

1) Hypertext Preprocessor

Hypertext Preprocessor (PHP) is an open-source server-side scripting language, it is used to generate dynamic web pages [16]. PHP can be used together with different web technologies such as Hypertext Markup Language (HTML) and Structured Query Language (SQL). In this study PHP is used to accept data from the client and send them to the database server for storing, to implement different functions that are executed on the server and also to define and enforce security measures.

2) Cascading Style Sheet

Cascading style sheet (CSS) is used for formatting the web page, it deals with the physical and visual display of the document [17]. CSS can be used as an inline style sheet, inline style attribute or from an external file. The external file gives the best advantage of using CSS as separate document contents and document layout because web developer will need to change only one document to control all web pages. CSS is used to format the physical structure of the HTML pages.

3) HTML

HTML is the main markup language for displaying web pages and other information that can be displayed in the web browser [17]. It is used to generate static contents of the web pages.

4) JavaScript

JavaScript is the programming language used in HTML pages, it can dynamically modify an HTML page to react with user input or validate user input [18]. JavaScript interaction with the user does not require communication with the server thus it reduces a significant amount of response time the user would send a request to the server.

5) JQuery

JQuery is a fast, small, and feature-rich JavaScript library, it makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy to use API that works across a multitude of browsers [19].

6) MySQL

MySQL is an open source relational database management system whose purposes is to store, manage and retrieve data [16]. MySQL can be used together with other web technologies such as Apache Server and PHP. In this study, MySQL is used as the back-end database, because in this study data are stored as the relationship between entities, each entity was represented as a table. MySQL was selected for implementation because it is open source, it is multi-threaded which enables it to be accessed by multiple users, it can be accessed through the Internet, and it provides support for query language such as SQL which is the standard language for querying the database.

7) Structured Query Language

Structured query language (SQL) is the standard language for storing, manipulating and retrieving data in databases [16]. SQL is used to communicate with relational databases, in this study, SQL was used to communicate with MySQL for the purpose of storing, managing and retrieving data.

8) Wamp Server

WampServer is a window web development environment. It allows you to create web applications with Apache2, PHP, and a MySQL database. Apache, MySQL, and PHP can be installed individually and most of the times for the purpose of web development are installed together. WampServer provides a user-friendly way of installing and configure Apache, MySQL and PHP components on Windows. In this study, WampServer 3.0.6 was used as a development environment with Apache version 2.4.23, MySQL version 5.7.14 and PHP version 5.6.25 installed. The following were the procedures for setting up the WampServer

i. Downloaded the WampServer through http://www.wampserver.com/en/
ii. The downloaded file was run for installing the WampServer in development computer.
iii. The "www" directory was automatically created in drive C. (c:wamp\www)
iv. Then inside "www" folder subfolder was created and application PHP files were kept inside.
v. Then on the web browser, this "http://localhost/ResearchSystem/ffoss/pages/" was typed and the server was ready to display web pages.

9) PhpMyAdmin

phpMyAdmin is a tool written in PHP intended to handle the administration of MySQL over the Web [20]. In this study, PhpMyAdmin was installed together with the WampServer and it was used to manage database, tables, relations, indexes, users, and columns.

10) PHP Configurations

Default configurations of the Wamp server do not allow large files to be uploaded. Thus PHP was configured so as to allow large files to be uploaded in the server. Those files were the setup files of the OSS. The following were the changes made in the php.ini file found in (C:\wamp64\bin\php\php5.6.25) to allow files up to 3000MB.

i. *post_max_size = 3000M*
ii. *upload_max_filesize = 3000M*

11) Apache Configuration

Apache HTTP server is an open source web server developed by the Apache software foundation open source community [21]. In this study, the Apache HTTP server was used to store web contents and accepting user request and send a response to those request by executing PHP scripts. Default Apache configurations do not allow the application to be available over the Internet. Thus Apache was configured so as to make the developed web-based application available over the Internet. The following changes were made in the httpd.conf file and httpd-vhosts.conf which is found in (C:\wamp64\bin\apache\apache2.4.23\conf) and (C:\wamp64\bin\apache\apache2.4.23\conf\extra) respectively.

i. The server was configured to listen to any IP address by changing httpd.conf because the server did not have static IP address during development and testing, also the server was configured to listen to port 80 and 8080.
  a. *Listen 0.0.0.0:80*
  b. *Listen [: 0]:80*
  c. *Listen [: 0]:8080*

ii. The server was configured to be accessible by any IP address using the following configurations in the httpd-vhosts.conf file. This was done by replacing the line "Require local" with "Require all granted"

V. Web-Based Application Functionalities

The application functionalities were implemented using the web technologies discussed in the previous section. Users refer to students, academicians, system administrators and lab scientists or any member of HLIs.

A. User Registration

Users are allowed to use the application with or without registration, the difference is that registered users benefit with more application services. During registration, users are required to provide various information including email and academic area of specialization. The email is verified and used for receiving notification and academic area of specialization was used for recommendation purposes.

B. OSS Uploading

The application gives privilege to administrator and other users to upload the OSS. They are required to provide important information about the OSS that included a link to the official website, academic area of use, supported operating system (OS), key features and description. The academic area of use is used for recommendation purposes.

C. User commenting and Rating

Users are given the ability to comment and rate the OSS after use to help other users of the OSS. The application computes the average rating value (ARV) of each OSS, and it is used to rank the OSS during computations of recommendations.

D. OSS Recommendations

OSS recommendations is the main functionality of this application. The academic area of specialization of the user, the academic area of use of the OSS and the ARV is used in the computation in recommendation functions. Three recommendation approaches were used in the application as discussed below.

1) Content-based approach: The application recommends OSS to the active user that are in the same academic area of use as those OSS that user liked in the past. The OSS are then sorted based on the ARV. Fig.3 presents the algorithm using the flowchart.

2) Demographic approach: The application recommends OSS to the active user when user's academic area of specialization is similar to the OSS academic area of use. The OSS are then sorted based on the ARV. Fig.4 presents the algorithm using the flow chart.

3) Collaborative filtering approach: The application recommends OSS to the active user based on which OSS other users from a similar academic area from specialization have liked in the past. The OSS is then sorted based on the ARV. Fig.5 presents the algorithm using the flowchart.

The three approaches used for recommendations make the application a hybrid recommender application because it uses more than one recommendation approach to recommend OSS to the active users. Further, the application sends a notification to the user when they have a new OSS recommendation.

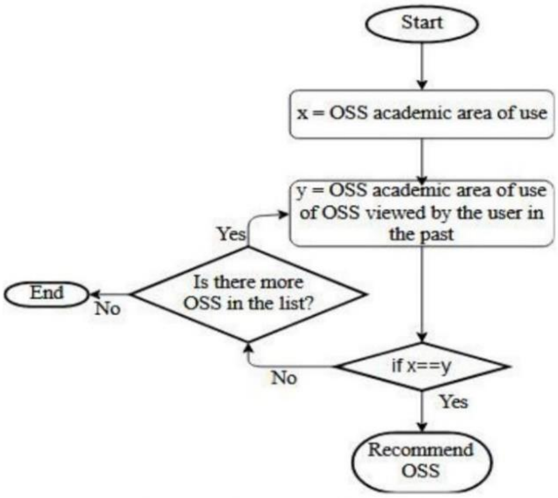

Start
x = OSS academic area of use
y = OSS academic area of use of OSS viewed by the user in the past
Yes
End
No
Is there more OSS in the list?
No
if x==y
Yes
Recommend OSS

Fig.3. Content-based Approach Algorithm

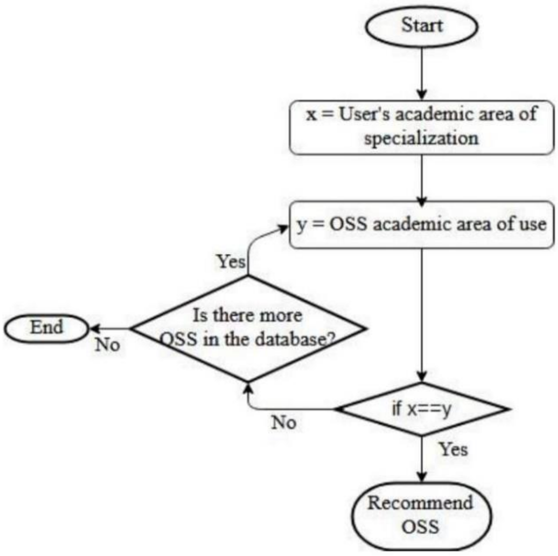

Start
x = User's academic area of specialization
y = OSS academic area of use
Yes
End
No
Is there more OSS in the database?
No
if x==y
Yes
Recommend OSS

Fig.4. Demographic Approach Algorithm

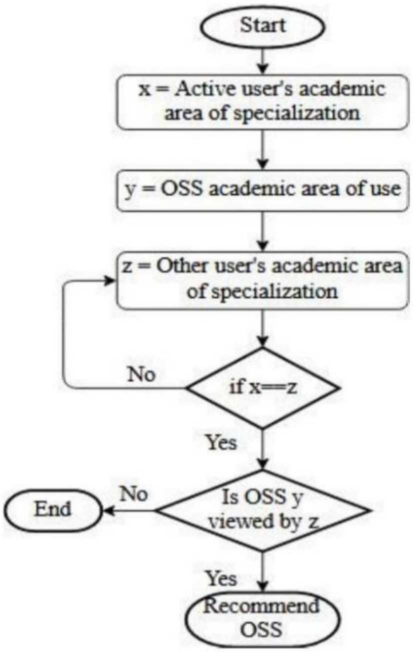

Start
x = Active user's academic area of specialization
y = OSS academic area of use
z = Other user's academic area of specialization
No
if x==z
Yes
End
No
Is OSS y viewed by z
Yes
Recommend OSS

Fig.5. Collaborative Filtering Approach Algorithm

VI. The Developed Web-Based Application

This section presents some of the functionalities of the developed web-based application. The application provides recommendations of OSS in most of the academic areas of specialization available in Tanzania.

A. Application Homepage

The homepage of the application is accessible to both registered and unregistered users. The homepage displays the OSS designed for academic purposes and provides links to download and view more details about the OSS such as a description, manual, other users' reviews and ARV of the OSS. Further, the page provides a dashboard that enables users to browse OSS from different academic area of use.

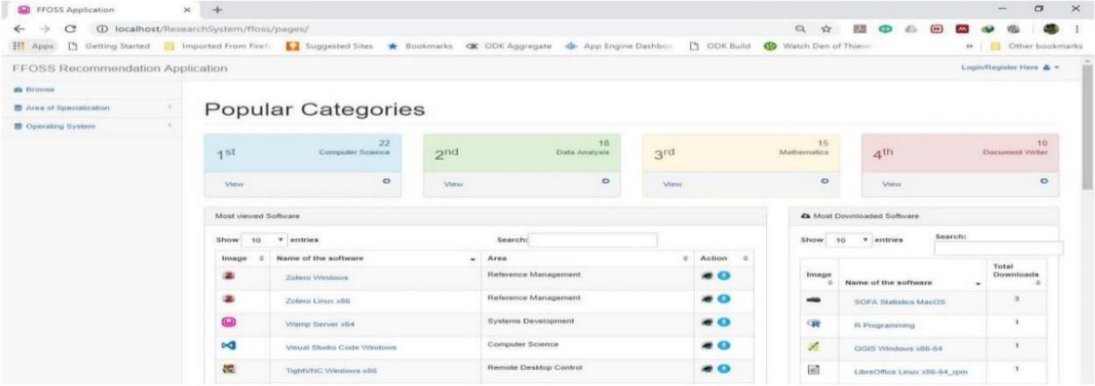
The homepage also provides links to login and registration pages. During registration, a user must provide a valid email address that has to be verified through a link that will be sent to the user' email address provided. The user will set a password that is hashed using an md5 hash, during login user must provide an email address as the username and the password. Fig.6 shows the application homepage.

B. OSS Notification

The application is implemented to send a notification to users. The user is allowed to select OSS academic area of use to receive notification. Whenever new OSS is uploaded by other users in any of the selected academic area of use, notification will be sent to the user's email address. Fig.7. show one of the notifications sent to the user's email.

C. OSS Recommendations

The application provides OSS recommendation to the users as a result of the implementation of recommendation algorithms. The recommendation approaches discussed in section V of this paper were implemented and the OSS are recommended to the users. The dashboard of the user's page shows a summary of the number of OSS recommended to the user, also the right section of the same page shows a list of the OSS recommended to the user. Fig.8 shows example on how the application recommended OSS to one of the users.



FFOSS Recommendation Application

Popular Categories

1st 22 Computer Science

2nd 18 Data Analysis

3rd 15 Mathematics

4th 10 Document Writer

Most viewed Software

Show 10 entries Search:

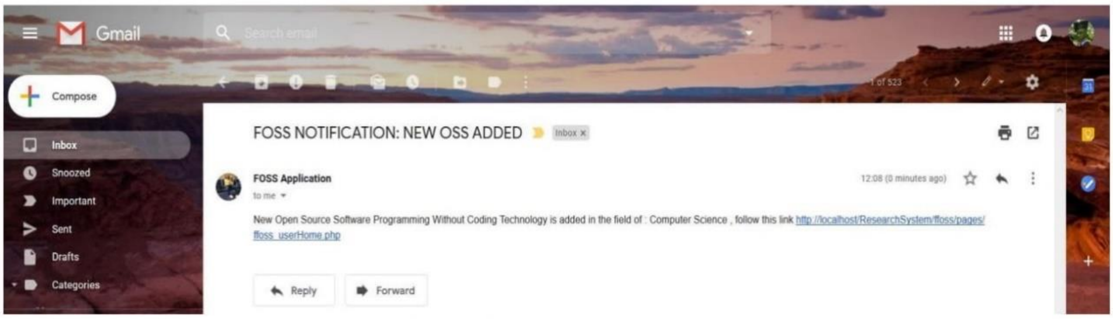
Image	Name of the software	Area	Action
	Zotero Windows	Reference Management	
	Zotero Linux x86	Reference Management	
	Wamp Server x64	Systems Development	
	Visual Studio Code Windows	Computer Science	
	TightVNC Windows x86	Remote Desktop Control	

Most Downloaded Software

Show 10 entries Search:

Image	Name of the software	Total Downloads
	SOFA Statistics MacOS	3
	R Programming	1
	QGIS Windows x86-64	1
	LibreOffice Linux x86-64_rpm	1

Fig.6. Developed Application Homepage

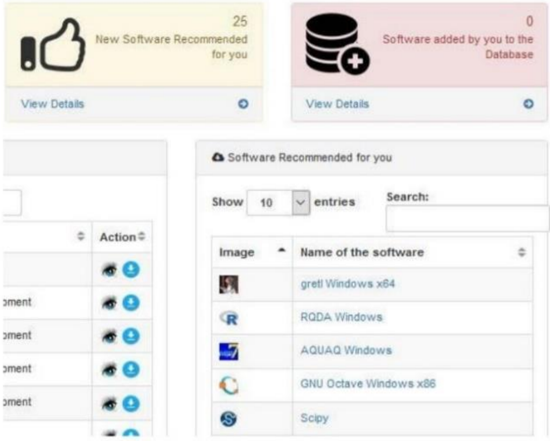


FOSS NOTIFICATION: NEW OSS ADDED

FOSS Application
to me
12:08 (0 minutes ago)

New Open Source Software Programming Without Coding Technology is added in the field of : Computer Science , follow this link http://localhost/ResearchSystem/floss/pages/floss_userHome.php

Fig.7. OSS Notification Sent through User's Email



25
New Software Recommended for you

0
Software added by you to the Database

Software Recommended for you

Image	Name of the software
	gretl Windows x64
	RQDA Windows
	AQUAQ Windows
	GNU Octave Windows x86
	Scipy

Fig.8. OSS Recommendations

VII. Discussion

A number of OSS tools to be used for academic purposes were identified, majority of them fall in these areas of use; data analysis, computer science, electronics, telecommunications, software development, network administration, mathematics, document reader and writer, mobile application development, mechanical engineering, image editors, computer drawings, web browser, geographical information systems, accounting, language management and reference management systems. These areas of use are the basis of recommendations provided by the developed application. The number of OSS recommended to the user depends on the recommendation algorithms. Each recommendation algorithm provides advantages but also has disadvantages that are solved by another. For example, the demographic approach cannot provide recommendations of OSS that are viewed by other users or OSS that are in the same area of use as those active user has used in the past, but these disadvantages are achieved by collaborative filtering approach and content-based approaches respectively. Basically, OSS that could not fit in one algorithm criteria may fit in the other. Eventually, this provides a large number of OSS recommended to the active user. This study suggests that hybrid recommender systems are more efficient and reliable in providing recommendations to users than normal recommender systems.

VIII. Conclusion

In this study we aimed at improving the adoption of OSS in Tanzanian HLIs through enhancement of individual awareness on the existing OSS designed for academic purposes using the recommendation approaches i.e. content-based, demographic and collaborative filtering. In our study, we nevertheless assessed the level of awareness of various people from HLIs on existing OSS that are designed for academic purposes and the software that are currently used in HLIs. The study, besides these, it explored OSS that can be alternative to currently used software, and more useful OSS that can fit for educational purposes from OSS vendors and other online platforms. Many OSS designed for the academic purpose were found and categorised according to their academic area of use. Eventually, a web-based application tool for the recommendation of OSS was developed by considering the existing situation found during the survey in HLIs. The OSS designed for academic purposes that were found and categorised were uploaded to the application's database and served as a foundation of OSS recommendations. Additionally, the study reviewed and selected proper web technologies that were used for the implementation of application functionalities. The three recommendation algorithms were developed using the web technologies to serve the purpose of recommending OSS designed for academic purposes to the HLIs.

The developed web-based application using the recommendation algorithms is able to; provide recommendations to both registered and unregistered users, send notification about new OSS recommendation through user's email, allow users to share OSS designed for academic purposes with another user through uploading function, allows the user to view and download the OSS and leave their feedbacks.

The ability to recommend OSS designed for academic purposes to its users and send notification through the user's email whenever new OSS is uploaded is a unique feature when compared to similar online platforms.

The developed web-based application can minimize the search time for OSS online, to minimize HLIs' cost since OSS are freely available and reduce cases of copyright infringements because most of the people in Tanzanian HLIs use cracked PS. We encourage the use of the web-based application for cost-effective selection of software tools for HLIs.

REFERENCES

[1] R. Al-Hajri, G. Al-Mukhaini, and R. Ramalingam, "Adoption of Free and Open Source Software Using Alternative Educational Framework in College of Applied Sciences," *Free open source Softw. Conf.*, pp. 15–18, 2017.

[2] A. S. Sife, E. T. Lwoga, and C. Sanga, "New technologies for teaching and learning: Challenges for higher learning institutions in developing countries," *Int. J. Educ. Dev. Using Inf. Commun. Technol.*, vol. 3, no. 2, pp. 57–67, 2007.

[3] M. Haule J, "Open Educational Resources Utilization in Tanzania Higher Learning Institutions," *Bus. Educ. J.*, vol. 1, no. 1, pp. 1–28, 2015.

[4] S. Ayushi and V. R. B. Prasad, "Cross-Domain Recommendation Model based on Hybrid Approach," *I.J. Mod. Educ. Comput. Sci.*, vol. 10, no. November, pp. 36–42, 2018.

[5] J. H. Lungo and J. J. Kaasbøl, "Experiences of Open Source Software in Institutions : Cases From Tanzania and Norway Experiences of Open Source Software in Institutions : Cases From Tanzania and Norway," in *Proceedings of the 9th International Conference on Social Implications of Computers in Developing Countries*, 2007, no. May 2007.

[6] B. S. Reddy, "Analysis of Learning Management Systems by Extracting Effectiveness Factor wise," *J. Eng. Educ. Transform.*, 2018.

[7] A. S. Al-Badi and A. H. Al-Badi, "Open Source Software in Higher Education: Oman SQU Case Study," *Inf. Technol. Educ. 21st Century*, vol. 8, no. 8, pp. 116–124, 2007.

[8] Z. Fariha and A. Zuriyati, "Comparing Moodle and eFront Software for Learning Management System," *Aust. J. Basic Appl. Sci.*, vol. 8, no. 4, pp. 158–162, 2014.

[9] L. John and A. E. Sam, "An Online Collaborative Discussion Platform for Bridging a Technological Reliance Gap in Higher Learning Institutions in Tanzania," *Int. J. Mod. Educ. Comput. Sci.*, no. November, pp. 29–35, 2018.

[10] E. Lwoga, "Making learning and Web 2.0 technologies work for higher learning institutions in Africa," *Campus-Wide Inf. Syst.*, vol. 29, no. 2, pp. 90–107, Mar. 2012.

[11] TCRA, "Quarterly Communications Statistics," Dar es Salaam, 2018.

[12] Slashdot Media, "SourceForge - Download, Develop and Publish Free Open Source Software." [Online]. Available: https://sourceforge.net/. [Accessed: 12-May-2018].

[13] Ola and markus, "AlternativeTo - Crowdsourced software recommendations." [Online]. Available: https://alternativeto.net/. [Accessed: 12-May-2018].

[14] J. S. Valacich, J. F. George, and J. A. Hoffer, *Essentials of Systems Analysis and Design*, Fifth. New Jersey: Pearson Education, Inc, 2012.

[15] J. C. Jackson, *Web Technologies: a computer science perspective*. New Jersey: Pearson Education, Inc., 2007.

[16] C. Darie and E. Balanescu, *E-Commerce Beginning PHP and MySQL*, Second Edi. United States of America: Apress, 2008.

[17] R. Nixon, *Learning PHP, MySQL, JavaScript, and CSS*, 2nd Editio. Sebastopol: O'Reilly Media Inc, 2012.

[18] D. Odell, *Pro JavaScript RIA Techniques*. New York: Apress, 2009.

[19] "jQuery." [Online]. Available: https://jquery.com/. [Accessed: 04-Aug-2018].

[20] N. Ibennetch, "phpMyAdmin download | SourceForge.net." [Online]. Available: https://sourceforge.net/projects/phpmyadmin/. [Accessed: 15-Aug-2018].

[21] Apache Software Foundation, "Welcome! - The Apache HTTP Server Project," 1995. [Online]. Available: https://httpd.apache.org/. [Accessed: 14-Sep-2018].

Authors' Profiles

Ambokile Okey is a Tutorial Assistant at the Dar es Salaam University College of Education (DUCE, Dar es Salaam, Tanzania). He obtained his Bachelor Degree of Science with Computer Science in 2014 from the University of Dar es Salaam. Currently, he is pursuing Master's Degree in Information and Communication Science and Engineering (ICSE) at the Nelson Mandela African Institution of Science and Technology (NM-AIST, Arusha, Tanzania).

Anael E. Sam is a Senior Lecturer in the Department of Communication Science and Engineering, School of Computational and Communication Sciences and Engineering (CoCSE), at the Nelson Mandela African Institution of Science and Technology. Dr Anael E. Sam is a holder of PhD in Electronics Engineering from the Institute of Electronics and Photonics, Faculty of Electrical Engineering and Information Technology, Slovak University of Technology, Bratislava, Slovak Republic. His research interests are in Mobile and wireless communications, Software Quality Assurance Engineering and Cyber Security.

How to cite this paper: Ambokile Okey, Anael E. Sam, "Web-based Application Tool for Recommendation of Open Source Software for Higher Learning Institution in Tanzania", International Journal of Modern Education and Computer Science(IJMECS), Vol.11, No.2, pp. 33-41, 2019.DOI: 10.5815/ijmecs.2019.02.05

POSTER PRESENTATION

A Web-based Application for Recommendation of Open Source Software for Higher Learning Institutions in Tanzania

Ambokile Okey[1], Dr. Anael E. Sam[2]

[a]The Nelson Mandela African Institution of science and technology, Arusha, Tanzania

Introduction

Open Source Software (OSS) provide a number of advantages to Higher Learning Institutions (HLIs) in developing countries, such as Tanzania. This is because they are freely available, free to make copies and free to distribute. Through literature reviewed, it has been observed that OSS usage in HLIs in Tanzania is low due to lack of awareness of OSS designed for academic purposes. This study presents how to implement a reliable and effective web-based application tool for recommendation of OSS for enhancing awareness of HLIs on existing OSS designed for academic purposes. There are three recommendation algorithms that were employed for filtering the OSS in this study: the content-based, collaborative, and demographic filtering. We used various web technologies for the implementation of application functionalities. The result is the reliable and effective web-based application tool that recommends OSS to individuals in HLIs for enhancing their awareness of OSS designed for academic purposes

Methodology

- ❖ User requirements were collected through questionnaires
- ❖ Prototyping is a process of designing and building a scaled-down but working version of the desired system. In this study, evolutionary prototyping was employed, and basic requirements were gathered and analyzed. Then the prototype was built and it was improved based on users' feedback. This process was iterative until we got the final version that users were satisfied with.

The Recommendation Algorithms

The following diagrams presents how the recommendation processes works

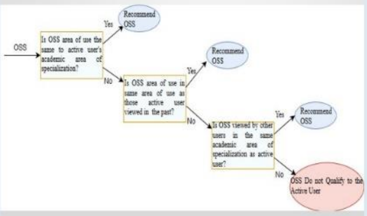

OSS
Is OSS area of use the same to active user's academic area of specialization?
Yes
Recommend OSS
No
Is OSS area of use in same area of use as those active user viewed in the past?
Yes
Recommend OSS
No
Is OSS viewed by other users in the same academic area of specialization as active user?
Yes
Recommend OSS
No
OSS Do not Qualify to the Active User

Figure 1: Decision Tree of the recommendation Algorithms

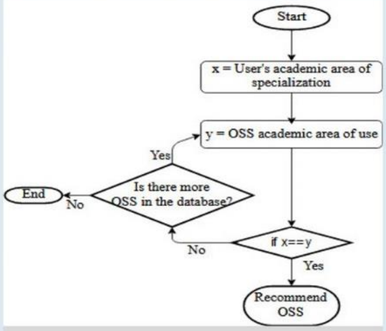

Start
x = User's academic area of specialization
y = OSS academic area of use
Yes
Is there more OSS in the database?
No
End
if x==y
No
Yes
Recommend OSS

Figure 2: Demographic Recommendation Algorithm

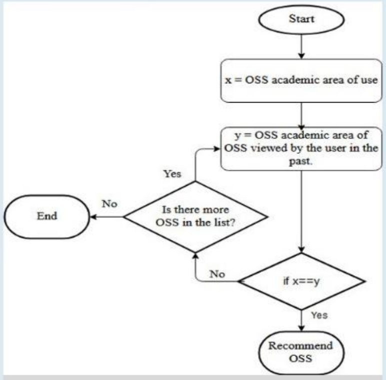

Start
x = OSS academic area of use
y = OSS academic area of OSS viewed by the user in the past.
Yes
Is there more OSS in the list?
No
End
No
if x==y
Yes
Recommend OSS

Figure 3: Content-based Recommendation Algorithm

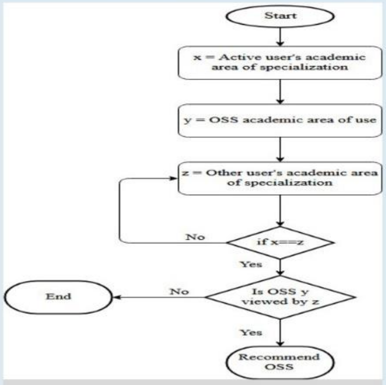
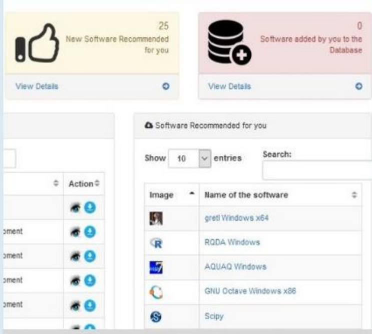

Start
x = Active user's academic area of specialization
y = OSS academic area of use
z = Other user's academic area of specialization
No
if x==z
Yes
Is OSS y viewed by z
No
End
Yes
Recommend OSS

Figure 4: Collaborative Recommendation Algorithm

Results of the Recommendation Algorithms


25
New Software Recommended for you
View Details

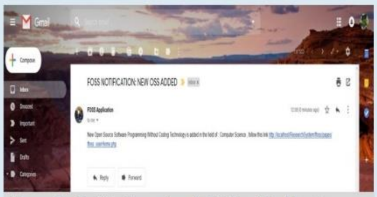
0
Software added by you to the Database
View Details

Software Recommended for you

Show 10 entries Search:

Image	Name of the software
	gretl Windows x64
	RQDA Windows
	AQUAQ Windows
	GNU Octave Windows x86
	Scipy

Figure 5: OSS Recommandations from the Application



FOSS NOTIFICATION: NEW OSS ADDED

FOSS Application

New Open Source Software Programming Without Coding Technology is added in the field of : Computer Science . follow this link http://localhost/ResearchSystem/foss/pages/foss_userhome.php

Figure 6: Notification about OSS Sent to User's Email

Conclusion

- ❖ The developed web-based application using the recommendation algorithms is able to; provide recommendations to both registered and unregistered users, send notification about new OSS recommendation through user's email, allow users to share OSS designed for academic purposes with another user through uploading function, allows the user to view and download the OSS and leave their feedbacks.
- ❖ The ability to recommend OSS designed for academic purposes to its users and send notification through the user's email whenever new OSS is uploaded is a unique feature when compared to similar online platforms.
- ❖ The developed web-based application can minimize the search time for OSS online, to minimize HLIs' cost since OSS are freely available and reduce cases of copyright infringements because most of the people in Tanzanian HLIs use cracked PS. We encourage the use of the web-based application for cost-effective selection of software tools for HLIs.