**STUDENTS RESULTS COMPLAINT MANAGEMENT SYSTEM**

JOSEPH OTIENO OCHIENG’

**IN 16/20274/15**

A Project report Submitted to the Faculty of Information Science and Technology, Computing Sciences Department, as part of the school curriculum in attaining a Bachelor’s Degree in Software Engineering, Kisii University.

**DECLARATION**

I confirm that this is work of my own doing as part of the school curriculum in attaining a bachelor's degree in computer science. The coverage and research emphasis employed are original and has not been used as it to earn an academic merit in any academic or research institution.

**By:**

**Joseph Otieno IN16/20274/15**

**Sign -: ……………………………………………….**

**Date -: ……………………………………………….**

**Project Supervisor:**

**Mr. Joshua Okemwa**

**Sign -: ……………………………………………….**

**Date -: ……………………………………………….**

**DEDICATION**

I dedicate this to my fellow students here at Kisii University. It is always good to have like-minded people to work to solve common problems. Last but not least, I dedicate this to the University at large, the School of Information Science and Technology (SIST), all friends and dear family.

**ACKNOWLEDGEMENT**

I acknowledge the institution at large, Kisii University at large and more specifically the Faculty of Information Science and Technology that facilitated the occurrence of this work. The diligent guidance in this academic path is invaluable and the support is highly appreciated.

I’d also like to recognize Mr. Okemwa, the Final Project Coordinator for being with us students from the very beginning and the much needed support she gave us in the initiation of this exercise.

I would like to highlight the bold effort by my Project Supervisor, again, Mr. Joshua Okemwa for personalized consultations and directions pertaining the proceeding of the exercise. As a mentor and guide, the exercise was greatly successful.

My fellow students at Kisii University have made this possible being there for one another and in effect, me; allowing for a conducive and interactive learning environment. Team work goes a long way in promoting better concept understanding and my colleagues have been quite co-operative.

Finally, would like to thank God for making all this possible. It is good to see what technology can achieve and I thank God for enabling me to follow along closely.

Table of Contents

Declaration………………………………………………………………………………………..2

Dedication……………………………………………………………………………………… 3

Acknowledgement……………………………………………………………………………….. 4

1.1 Background information……………………………………………………………………... 6

[List of Abbreviations 7](#_Toc532414075)

[DFD - Data Flow Diagram 7](#_Toc532414076)

[ERD - Entity Relationship Models 7](#_Toc532414077)

[List of Appendices 7](#_Toc532414078)

[**Appendix 1** 7](#_Toc532414079)

[**Appendix 2** 7](#_Toc532414080)

[1.2 Problem Statement 9](#_Toc532414081)

[1.3 Objective of Study 9](#_Toc532414082)

[1.3.1 General Objective 9](#_Toc532414083)

**1.4 The scope of study………………………………………………………………………………………………………………………10**

**1.5 Justification of the study……………………………………………………………………………………………………………10**

[**2.2.1Case study Workpro** 11](#_Toc532414084)

[**2.2.2 Case Study TrackWise** 12](#_Toc532414085)

**2.2.3 Case Study Everest………………………………………………………………………………………………………………12**

**2.2.4 Case study Freshdesk……………………………………………………………………………………………………………12**

[2.3 Limitations of the existing Systems. 13](#_Toc532414087)

[**2.4 Advantages of the proposed system** 13](#_Toc532414091)

[3.1 System Development Methodology 16](#_Toc532414096)

[**3.1.1 Iterative and Incremental Model** 16](#_Toc532414097)

[**3.1.2 Continuous Integration** 17](#_Toc532414098)

[3.2 System Analysis and Design Tools 17](#_Toc532414099)

[**3.2.1 Data Flow Diagrams (DFD)** 17](#_Toc532414100)

[**3.2.2 Pseudocode** 18](#_Toc532414101)

[**3.2.3 Use Case Diagrams** 18](#_Toc532414102)

[**3.2.3. Entity Relationship Models** 18](#_Toc532414103)

[3.3 System Implementation Tools 18](#_Toc532414104)

[**3.3.1 Hardware requirements** 19](#_Toc532414105)

[**3.3.2 Software requirements** 19](#_Toc532414106)

[3.4 System Testing and Validation 19](#_Toc532414107)

[**3.4.1 Unit Testing** 19](#_Toc532414108)

[**3.4.2 Integration** 19](#_Toc532414109)

[**3.4.3 System testing** 20](#_Toc532414110)

[Appendix 1 -: Time schedule 22](#_Toc532414111)

[Appendix 2 -: Budget 23](#_Toc532414112)

# List of Abbreviations

# DFD - Data Flow Diagram

# ERD - Entity Relationship Models

ORM - Object Relation Mapper

# List of Appendices

## **Appendix 1**

Time Schedule

## **Appendix 2**

Budget

# 

# CHAPTER ONE

# 1.0 INTRODUCTION

# 1.1 Background Information

# The key to building lasting students-lecturers relationships is to create superior value and satisfaction of the students which can be achieved through effective Results complaint management system. Every Institution must make Results complaint management system its one of the key strategies needed if they must succeed in their learning activities. The significance of Results complaint management system in the performance of learning institution cannot be overstressed. Results complaints are inevitable in any learning institution no matter how good such an Institution may be. Some complaints serve as a feedback mechanism for many institution. Students of such Learning institution may want to express their feelings and impressions on the quality of the education or services offered by the institution. Some complaints tend to relay a message on the missing marks or retake of the students. Results complaints becomes useful when such complaints attract quick and effective response from the Institution. Learning institutions are expected to hold their students in high esteem since the Students are the lifeblood of any Learning Institution. The type of result complaint specifically in Kisii University does not have immediate feedback, students are issued with forms to register the units they are having issue with, returns the form to various departments, but there will be no immediate feedback for the students to make appropriate arrangement whether to resit or wait for transcripts update.

# 1.2 Problem Statement

There are a lot of problems arising when students want to make complaints regarding there exam results to their various departments, one of these problems is the office hour, most students wants to register complaints at their free time, that is when they don’t have classes or during the lunch hour, but at this time the offices are closed according to the office timing

The manual way of registering a complaint which is filing has the highest probability of data loss, when a file is lost or damaged the data containing all the students’ complaints won’t be available and students won’t get any help from their departments. When the administration wanted to see listing of complaint, the department need to check the complaint first and then prepared the list. Certainly it will take time and only wasting the time.

# 1.3 Objective of Study

# 1.3.1 General Objective

To design and implement an effective result complaints management system.

**1.3.2 Specific Objective**

The specific objectives of the system will be:

1. To analyze the current result complaint management in Kisii University.

2. To design a Results complaint system basing on among factors such as the downsides of the current system, the changing user needs and resources available on our Learning institutions.

3. To implement the designed system in a computer programming language as a concrete testable and deploy-able solution to Results complaints.

4. To test and validate the designed Result complaints management system in Kisii University.

**1.4 Scope of the study**

However much a complaint system can be general to all complains, this study is constrained to result complaints. This is because of the many complains registered in learning institutions are results complaints. The aim is to carefully solve and respond to complains registered by the students more commonly concerning their results, the proposed system is to be implemented at Kisii University. A user log into the system, register complain and submit, and must be specific for easy response by various departments

**1.5** **Justification of Study**

The implemented system will solves the problems in results complaints for Kisii University . From complaint registration to time taken by administration, department or lecturers to respond to the complaints. The stakeholders include the students, the institution(Kisii university) and lecturers as they are the ones involved in day to day learning activities.

**CHAPTER TWO**

# 2.0 LITERATURE REVIEW

# 2.1 Introduction

Results complaint management is the process of how Learning institutions handle, manage, respond to and report students’ complaints. Systems are put into place to track and trend the data that is captured by Result complaint management system processes.

Complaint is a behavior that students show their dissatisfaction against their expectations, when Students feel dissatisfied of their results, services or other factors in their institutions. The behaviors of complaints occur during or after their studies.

Students know when their complaints have been responded to that satisfy them and when not. Such service, both excellent and poor, has a strong emotional impact upon individuals as students, creating intense feelings about the Institution, its staff and its services, and inﬂuencing their loyalty to it.

Learning institutions with departments which respond slowly or poorly to students Complaints concerning their results are a ‘pain to Students’. They are often described as ‘a nightmare’ to deal with. Their staff and systems made it difﬁcult for Students to register their complaints with them. They just do not care about the students or their experiences. Some institutions has various departments which respond well and quickly to complaints raised by the students. Such institutions make it into the list of institutions providing excellent service.

**2.2 Case study for Complaints Management systems**

## **2.2.1Case study Workpro**

Workpro is a comprehensive system for recording, managing and reporting on all casework, including complaints, inquiries and issues. It guides staff through a best practice complaint handling process so every customer is handled in a consistent, fair manner. Workload and performance management tools keep cases on track and ensure everyone knows exactly what needs to be done, when and by whom. Everything is kept in one place - all stored, secure and searchable. Workpro is ideal for any organization looking to manage complaints, cases and issues more effectively, whether public sector or private industry. It is a case and complaints management product which can be adapted to meet the needs of all kinds of organization

## **2.2.2 Case Study TrackWise**

## TrackWise complaint management helps organization to regulate compliance and customer retention. With an integrated TrackWise complaint management system, organizations can track and investigate complaints, and set subsequent corrective actions in motion. It Streamline complaint management operations and reduce costs by improving response and closure times. Automate the process of assigning complaints and related investigations based on customers’ complaints. TrackWise has the ability to spot product defects or safety issues faster than ever before. This integrated enterprise platform reveals events occurring on National or regional scale. Organization can protect their brand’s reputation and increase customer satisfaction by being proactive.

**2.2.3 Case Study Everest**

Everest Complaint Management System is an effective resource for building customer relationships by engaging with customers and providing timely responses to customer feedback. Flexible workflow, built-in correspondence and extensive reporting are among the many features that will streamline Organization’s business or activities.

Everest is an intuitive software solution that takes customer feedback directly to the heart of the organization and keeps it in the spotlight until it's resolved.

**2.2.4 Case Study Freshdesk**

Freshdesk is a cloud-based customer support platform that was founded with the mission of enabling companies of all sizes to provide great customer service. It is an online help desk software that integrates traditional support channels such as email, phone, chat and the web with new social channels such as Twitter and Facebook. It is a customer support software that offers everything a business needs.

With Freshdesk, you can get a grip on customer complaints coming in through phone, email, chat, Twitter and Facebook or even your mobile app! You can convert complaints into tickets, organize and assign them between teams, respond in bulk for frequent complaints and get insights into issues affecting their service.

# 2.3 Limitations of the existing Systems.

The systems described above have gone miles in trying to hide away the unavoidable hardships that arise when someone wants to register complain. None of them is the perfect solution for students to register complains concerning their results.

When logging a complaint, the procedures can often be seen as time consuming and irritating. Some staff are not confident with the complaints handling process. By using a Result complaints management system, complaints can be logged, tracked and processed from phone call to resolution.

In the existing systems sometimes the complaints may be ignored by the staffs. Even an efficient department staff may not be able to handle more than one complaint at a time.

**2.4 Advantages of the proposed system**

**Acknowledgement & Communication**: How Do students Know They’re Being Heard?When you have a complaint, the most important thing is that your complaint is heard and your problem is addressed. If complaints get lost, they can’t be resolved. Good complaint management will allow you to set up automatic acknowledgements.

**Security & Confidentiality:** Who has access to view complaints can be a large concern. If you’re dealing with confidential information, you want to make sure that only those people who absolutely need to see the incidents can do so. Using the proposed system for complaint management will allow for restriction of access when necessary.

**Repetition & Reporting**: If there is a comparable complaints, are they handled the same way each time? Are they fully resolved? Is there a way to pull data on why the complaints happened? Ultimately, there are needs to be able to identify trends and prevent future complaints from the students.

**Accountability:** When a complaint is passed to someone for resolution, how do you know they’ve handled it properly? With the proposed system, you can assign an incident to any team member or an entire group of people. Any communications will be time and date stamped so you’ll easily see who update the incident.

# CHAPTER THREE

# 3.0 METHODOLOGY

# 3.1 System Development Methodology

The development methodology is the framework, tooling and procedures through which the system is brought to actualization. From the researched idea and the found analysis, there is need to plan, control and monitor the development process. This is where the methodologies come in. It is considered a good practice since the methodologies are well established, tried and tested. This also provides the developers with an organized platform for activities like gathering requirements, system design and analyzing, testing, implementation so as to attain attributes needed such as good software quality.

## **3.1.1 Iterative and Incremental Model**

Other than working with the waterfall model, the goal of the system is a polished and tested product at the end of the development period. This was contributed upon by the repetitive cycles involved in the iterative model or scrum model with incremental changes being added over time. This was a good production boost since it allowed the designer and developer to apply what has been learned from the previous iterations of the application.

The different modules were in turn developed in the iterations, where the initial iteration gives a base viable product that implements the basic features and runs or works. The iterations later on bring modifications, feature additions and an opening for adding more improvements in the future iterations.

Analysis of iteration is based on the specified feature set and the consumer feedback, in this case the sports fanatics. The iterations also ensure extensive testing to mark a feature as done.

In Incremental development, the system functionality is delivered sliced as proportions. Each slice involves cross discipline involvement from design to implement to testing. This model of operation groups the activities is a slice in four phases:

1. Inception phase to identify the project scope, the functional and non-functional requirements and provide enough detail for the project to work.

2. An elaboration phase used for providing a working architecture to solve most of the non-functional requirements.

3. Construction phase to implement the proposed architecture incrementally from analysis, design, implementation and testing of the production-ready code.

4. The transition phase pushes production code to the production or business environment.

With the user involvement in each iteration, the project is sure to stick to addressing the relevant and correct needs and maintain core software qualities such as stability.

## **3.1.2 Continuous Integration**

Here, as the specifications to the project change or get updated in each iteration, the implementation also changes in turn, creating a need to keep the mainstream product in an organized manner. Here the project code shall be continually tested and merged to the mainstream product once approved fit by the projects technical management. There are a number of continuous integration services online and the one used for this project is the famous Travis CI hosted service.

# 3.2 System Analysis and Design Tools

There are several activities that were undertaken to enable the transformations of requirements document to a working implementation. These analysis and design tools enabled for a simpler or intermediate way to change code implement the project in real code.

## **3.2.1 Data Flow Diagrams (DFD)**

A data flow diagram was used in the project as graphical representation or emulation of how the data pieces define the operations through the system. This is rather vague to mean to show the data inputs into the system, the various outputs from the system in the defined format and the persistence of the data in the system. The DFDs are logical showing the system process and how the data flows from different entities to the other.

## **3.2.2 Pseudo code**

Pseudo code provided more concise way to represent the functional requirements of the project this was much needed to design the sales module that needs a high integration of the related modules.

It may be considered as an augmented programming language though it avoids constructs like variable declaration. It may be full of comments and descriptions. Which act as hints during the implementation in a programming language such as Smalltalk, Java, C++ or maybe Python. This method is also preferred to the closely related – Structured English – model in that it provides a method to perform the task, as if a computer is executing the code.

## **3.2.3 Use Case Diagrams**

In the design of the project, Uses-Case Diagrams modeled the user interactions with the system, in an effort to fulfill the functional requirements. The mannerisms in which the different users of the system choose to seek services on the system is a valid use-case and was be shown graphically as such.

## **3.2.3. Entity Relationship Models**

Entity-Relationship Model is a type of database or program model based on the notion of real world entities and relationship among them. We can map real world scenario onto ER database model. ER Model creates a set of entities with their attributes, a set of constraints and relation among them.

An ERD It is made up of entities and their relationships shown there.

# 3.3 System Implementation Tools

These are the collective requirements that were needed to develop and run the application as needed by the target users, ticket seekers. This system is implemented as a web application with an interactive web client. As such, the ticket seeker needs a working Internet connection to access the ticketing services. With support for internalization, this system can be used in a number of languages such as English and German.

## **3.3.1 Hardware requirements**

A descent processor of speeds 1GHz and above

Should have at least 512 MB of RAM and 1GB or more recommended

Adequate disk storage space of 20GB to 30GB, more is recommended.

Peripherals such as keyboard, mouse / touch devices.

Access to the internet from the device.

## **3.3.2 Software requirements**

. Programming languages -: Python3.5 web framework Django.

. Database for all application storage and persistence -: MySQL

. Operating System -: Linux/Unix platforms and Windows 7, 8, 8.1 and 10 operating systems.

. Web Server -: Apache Web Server

# 3.4 System Testing and Validation

The system was developed by heavily testing the modules at all abstractions and the following types of tests were observed.

## **3.4.1 Unit Testing**

We can consider a unit as the smallest part of a program that can be tested. This was done with the

relevant data to check for technical and semantic functionality. Properly tested modules, regardless of the paradigm of implementation used can ensure the development of concrete abstractions. This allows for smoother integration. This were done with Python’s unit test library.

## **3.4.2 Integration**

The system has several modules each of which is tested as a unit. The modules are delivered as pluggable applications in the project. This process entails integrating the applications and testing whether they communicate in proper synchronization as intended. This also involved usage of real data needed by the program and testing against corrupted data too.

## **3.4.3 System testing**

System testing was conducted last to make sure the registration of users and ticket processing are working as expected. The means of payment is also tested to make sure the product is market ready.

The technical aspects of systems such as browser tests will be conducted with Selenium. The sample prototype was hosted and available online as a running application.

**CHAPTER FOUR**

# 4.0 SYSTEM ANALYSIS AND DESIGN

System analysis is a process of collecting and interpreting facts, identifying the problems, and breaking down of a system into its constituent components. This problem solving technique improves the system and ensures that all the components of this system work efficiently to accomplish their purpose.

# 4.1 System Analysis

Result complaints system needs to be able to handle several disciplines so that it can work in harmony. The study on the environment identified the inadequate means of registering results complains. Above all there is also little information on how students can express there dissatisfactions about their results. The institution also needs to be able to respond to students complains to ensure that issue concerning their results are sorted . Below are the system requirements.

# 4.2 Requirement Specifications

The system needed to be designed to meet the following requirements for Students results complaint system.

## **4.2.1** **Functional requirements**

The following was a listing of the functionality that was achieved by the system.

#### 4.2.1.1 Creating Issue or complaint

The student is able to create an issue here, specifying the kind of complaint he/she may be having . There are several types of sporting events to choose from. There is a map widget which is versatile and covers a wide geographical area.

#### 4.2.1.2 Viewing issue progress

The student can view and track the progress of the issue he/she created whether if it has been assigned to relevant to the relevant department to solve it

#### 4.2.1.3 View issues in Que

The student can view the issues which he/she created but are yet to be solved

#### 4.2.1.4 Complete issue

The student can view the issues he/created and have been responded to by the administrators

#### 4.2.1.5 Issue progress report

The administrator can give the issue progress report to the student concerning the issue created

#### 4.2.1.6 Assign issue

The super administrator is able to assign the issue created by the students to other administrators who have limited privileges which is to solve and respond to issue created by the students

## **4.2.2 Non-Functional requirements**

The application was designed to fulfill the following non-functional requirements:

### 1. Speed of operation.

The process of creating issue and responding to issue is supposed to be fast and time saving. The system should have good performance enough to avoid keeping the user waiting. Users on low bandwidth gadgets are also supposed to be actively engaged.

### 2. Product Quality Attributes

The system should have user friendly display, contents that can be easily bookmarked.

### 3. Security concerns.

The user credentials are managed securely.

## **4.2.3 Hardware and Software requirements**

### Hardware Requirements

* A descent processor of speeds 1GHz and above
* Should have at least 512 MB of RAM and 1GB or more recommended
* Adequate disk storage space of 20GB to 30GB, more is recommended.
* Peripherals such as keyboard, mouse / touch devices.
* Access to the Internet from the device.

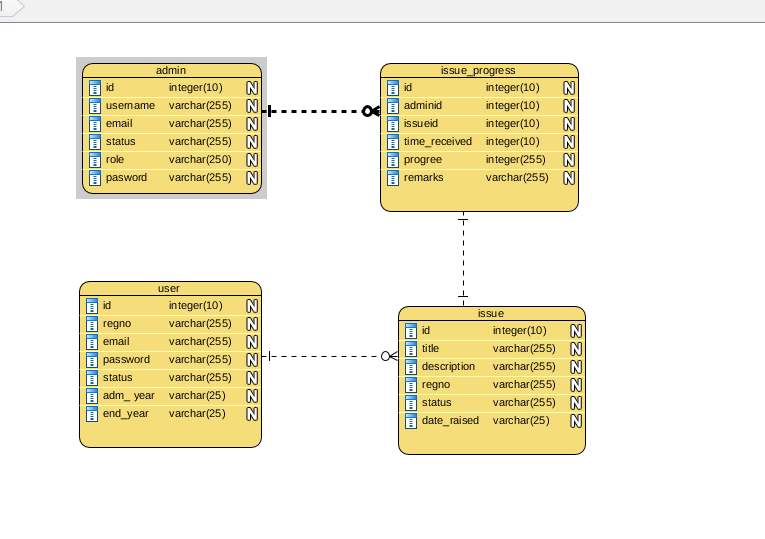
### Software requirements

* Programming languages -: Python3.5 web framework Django and JavaScript.
* Database for all application storage and persistence -: MySQL
* Operating System -: Linux/Unix platforms and Windows 7, 8, 8.1 and 10 operating systems.
* Web Server -: Apache Web Server
* Web browser such as Google Chrome or Mozilla Firefox

# 4.3 System Design

## **4.3.1 Logical Design**

The designer deals only with defining the types of information that are needed by the system. This process of logical design included arranging data into a series logical indivisible relationships called entities and attributes. **Logical Database Design for Students Results Complaint System**

Figure 1 Logical Diagram for the Students Results Complaint System

## 4.3.2 Physical Design

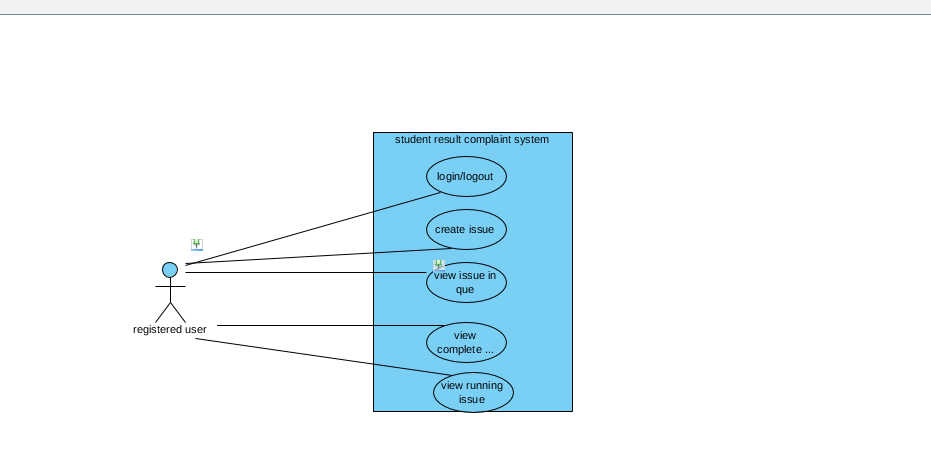
The physical design is the actualization for the logical design. Database tables were mapped to the language objects as an abstraction layer called models. The models help generate tables, interface forms and models forms, validation pattern and reports are easily gathered through means such as comma separated values (CSV) and spreadsheet data. The database table constraint fields are also set here in the physical design. Among the details mapped in this stage are as follows.

* Database table entities needed.
* Table attributes to columns primary unique identifiers or primary key constraints.
* Relationship to foreign key constraints
* Many to Many relationships among the database tables.
* Table column attributes
* Unique identifiers to unique key constraints.

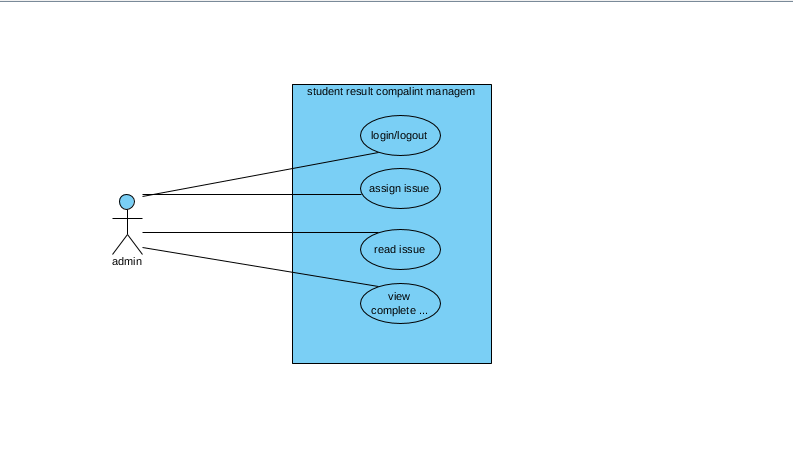
## **4.3.3 System use-case diagrams**

These are meant to illustrate the several activities happening in the system by the given stakeholders.

Use case for registered user-defined



*use case for admin*



## **4.3.6 System Implementation Pseudocode**

Pseudocode is an intermediate form of code used for visualizing code before it is written. The following samples of pseudocode was used to code down the specified modules before being integrated to the main system.

**4.3.6.1 Issue/complaint Pseudocode**

Here is a simple algorithm for Students Result Complaint System.

Start Procedure: Get sales for Event

Issue = User Issue all

Current Issue = Issue.find(issue id)

Import issue object:

Issue Object = Add new issue object, pass Current issue

issue Object find Total issues

Return issue

End Procedure

# CHAPTER FIVE

# 5.0 SYSTEM IMPLEMENTATION, TESTING AND VALIDATION

# 5.1 System Implementation

Systems implementation is the process of defining how the physical system should be built, ensuring that the system is operational and usable by stakeholders. This is done by ensuring that the information system meets quality standard (i.e., quality assurance) and is done in the stated time limit.

## **5.1.0 How the system works.**

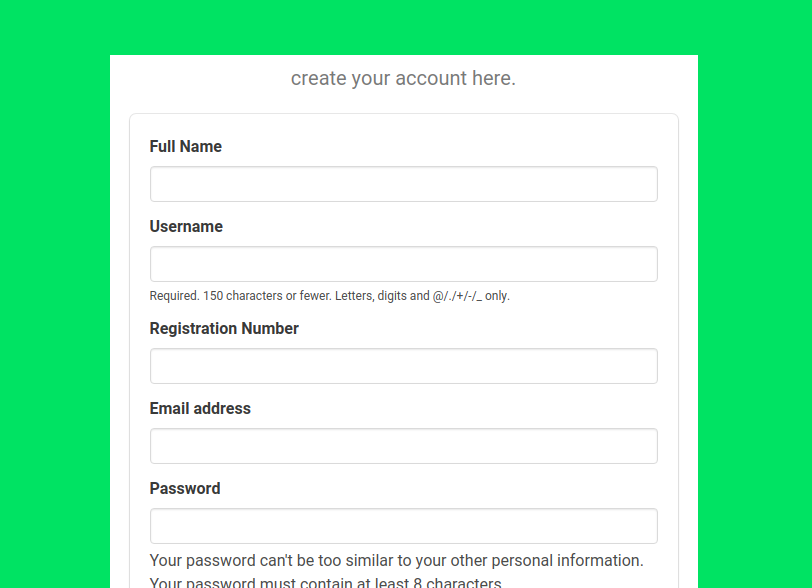
* After registering, the student creates issue concerning the viewed result, post it and wait for the response for the issue created
* The super administrator views the created issue by the student and goes a head to assign the issue to various department where the student belongs to.

## **5.1.1 The student result complaint system**

This is where the student creates an account on the system, adds and manages his/her complaints/issue.

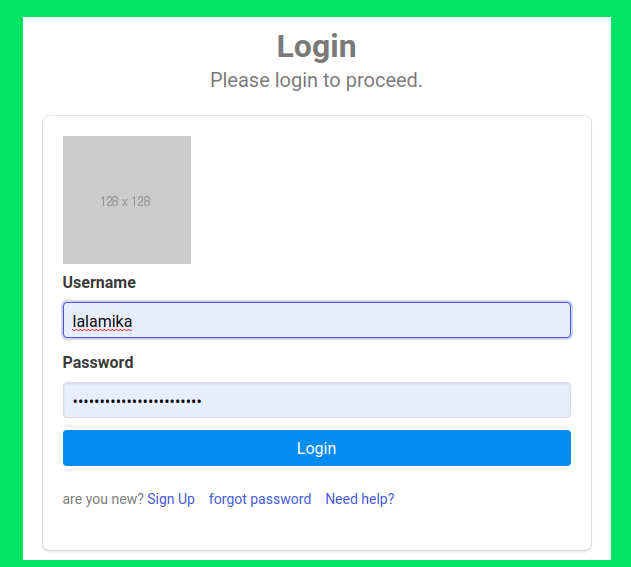
### a) Account Creation

### An account creation form is displayed to be filled and submitted.



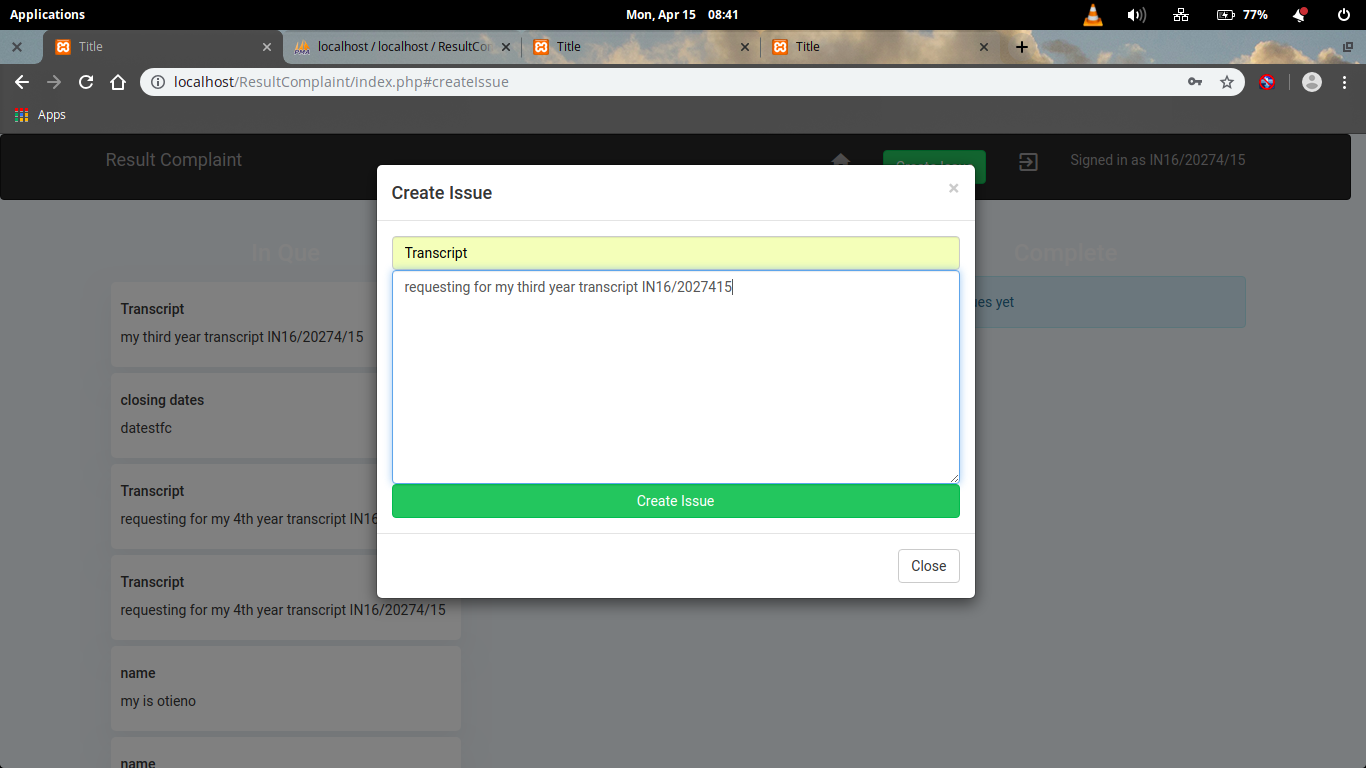
### b) Login and Authentication

A login form is displayed to be filled and submitted. If correct details are posted, the user is log-in and a session started.



### c) Issue/complaint Creation

An issue creation form is displayed to be filled and submitted. The issue is verified, published and displayed in the system's administrator’s panel.



### 

### d) Issue/complaints Listing

### This is a list of posted issue/complaints on the system. The system also display a lock button to differentiate between the processed and unprocessed issue/complaint

### 

# 5.2 System Testing

# 5.2.1 Unit Testing

The completed system entities were tested to ensure they were working correctly. This includes confirmation of the input types and the proper display of error messages.

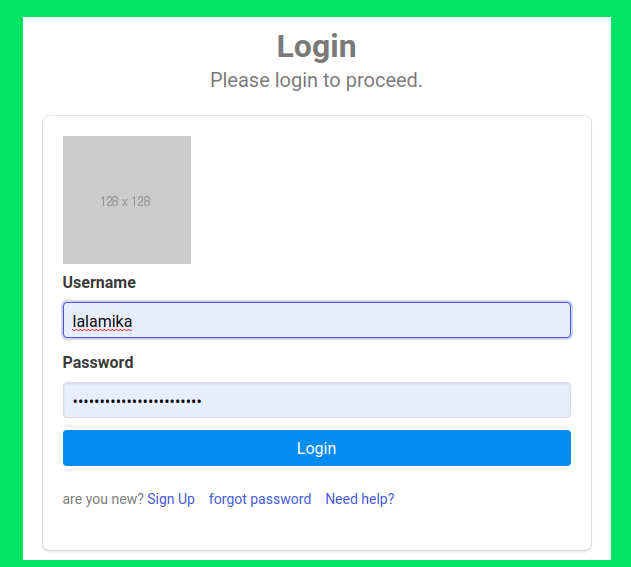
### a) issue creation

The created issue should also meet the proper verification rules before it is accepted in the system.

The validation happens on the server.

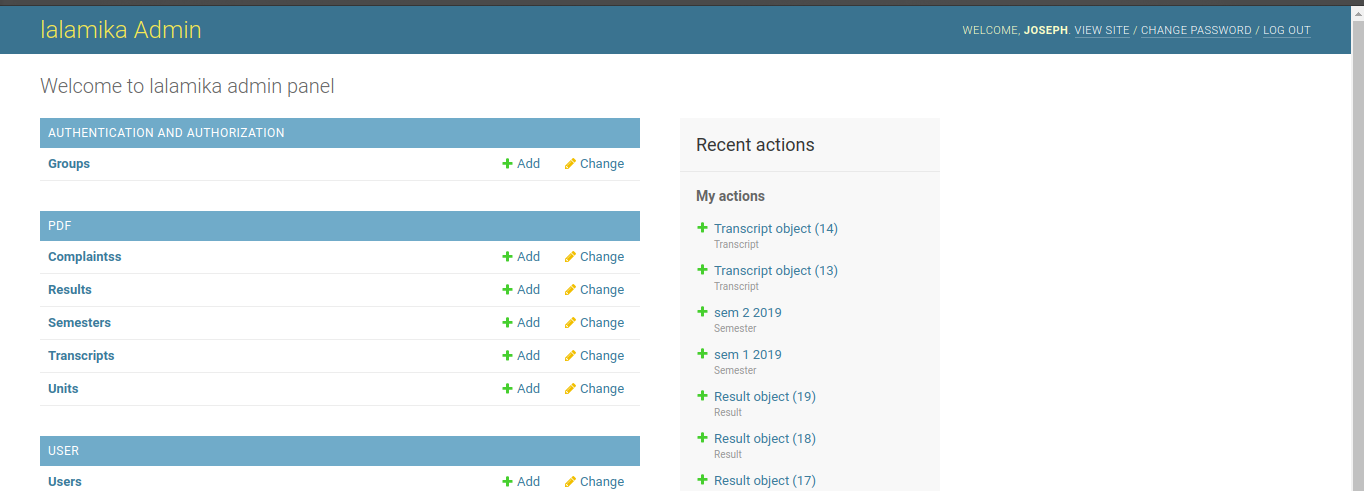
### b) User Login and Registration

If the user enters wrong details, the system shall warn appropriately.



### c) Admin User Dashboard

A root user who administers the entire system. A list of registered system data models is displayed allowing for basic manipulation. This is as shown below.



## 5.2.2 Integration Testing

After all the modules were completed and tested, they were also tested on their compatibility with other modules. The issue/complaint creation module works in harmony with issue/complaint listing module.

The user authentication module cannot allow wrong credentials.

## 5.2.3 System Testing

Tests for each of the programs and units of the system were done and when errors debugged and patched. The date involved is real and working. Upon execution, system output corresponded to design anticipations and requirements. The completed system was compiled, bundled, uploaded, dependencies installed and hosted on a virtual private server.

The system is ready for handling students results complaints for Kisii University

# CHAPTER SIX

# 6.0 SUMMARY, CONCLUSIONS AND RECOMMENDATIONS

# 6.1 Summary

This research work addressed Students’ result complaints management in Kisii University. It was affirmed in this study that institutional complaints are inevitable. A University system like Kisii use to receive various complaints from students. Complaints lodged range from Academic, Administrative, social and other issues relating to the student. This platform allows for complaints to be lodged remotely by students with issues relating to their registration, examination, examination result, computation of their Grade Point Average (GPA) complaints and thereby enhances the response time for the appropriate unit to resolve the addressed complaints.

# 6.2 Conclusions

# Finally in the students result complaints management, we have a system where students have problems he register in the system with attributes and then log in and record the complaint in it and then the claim is forwarded to the staff to resolve it and notifying the person who is responsible to do necessary action and to the student about the fixed and all the functions working fine and smooth.

# 6.3 Recommendations

In general, this system works for most use-cases comfortably. Being an application targeting users from a specific field. With better design of the system modules used in the system, the student may go as far as following the progress of the complaint/issue raised. This instills confidences in students about the system. This demands specialized drawing and illustration tools before being translated to HTML canvas images with good interaction.

More research on this system is required to fully identify and eliminate some weaknesses and integrate it with other information system to assist students in acquiring information faster

The system should be made accessible so that all students can access it via their phones

There is need for system upgrade as user’s requirement change. User requirements differ with time , therefore, it is of great help for the system to be flexible enough

Backup should be done frequently to avoid data loss in case of hardware or software malfunction.

# **REFERENCES**

[1] Osman Nasr AndEnayatAlkhider, Online Complaint Management System. Vol.2 issue 6 June 2015.

[2] H. Kim, P. Howland, And H. Park, “Dimension Reduction In Textclassification With Support Vector

Machines,” J. Machine Learning Research, Vol. 6, Pp. 37-53, 2005.

[3] E.F. Combarro, E. Montan ̃ E ́S, I. Dı ́Az, J. Ranilla, And R. Mones , “Introducing A Family Of

Linear Measures For Feature Selection In Text Categorization,” Ieee Trans. Knowledge And Data

Eng., Vol. 17, No. 9, Pp. 1223-1232, Sept. 2005

[4] F. Sebastiani, “Machine Learning In Automated Text Categorization,” Acm Computing Surveys, Vol.

34, No. 1, Pp. 1-47, 2002

[5] Boroughs Partnership NHS Trust, Complaints Management Procedures, January,

[6]2007. NHS Greater Glasgow and Clyde, Complaints handling Policy and Procedure,

[7]April, 2006. North Lincolnshire NHS Primary Care Trust, AIDE Memoir To Complaints Handling. April, 2007. Quality 2001 A Reality, Ministry of Health Trinidad and Tobago. March, 1998.

[8] Ronald Berenbeim, Non-union Complaint Systems: A Corporate Appraisal (NY: The Conference Board, Report No 770, 1980)

[9]Mary P. Rowe, PhD, "Are You Hearing Enough Employee Concerns?" with Michael Baker, in Harvard Business Review, Vol. 62, No. 3, (May–June, 1984), pp. 127–136

[10]Mary P. Rowe, "The Non-Union Complaint System at MIT: An Upward-Feedback Mediation Model," in Alternatives to the High Cost of Litigation, Vol. 2, No. 4, (April, 1984), pp. 10–18.

[10]Corporate Non-union Complaint Procedures and Systems" (NY: Praeger, 1988)

[11]Ury, William J., Jeanne M. Brett, and Stephen B. Goldberg, Getting Disputes Resolved: Designing Systems to Cut the Costs of Conflict, the Program on Negotiation at Harvard Law School, Cambridge, MA (1993)

[12]Boris A. Galitsky, María P. González, Carlos I. Chesñevar. A novel approach for classifying customer complaints through graphs similarities in argumentative dialogues[Decision Support Systems](http://www.sciencedirect.com/science/article/pii/S016792360800208X) Volume 46, Issue 3, February 2009, Pages 717–729

[13] Boris A. Galitsky Reasoning about attitudes of complaining customers Example <http://www.complaintsportal.net/> [Knowledge-Based Systems](http://www.sciencedirect.com/science/article/pii/S0950705106000785) Volume 19, Issue 7, November 2006, Pages 592–615

[14] See "People Who Feel Harassed Need a Complaint System with both Formal and Informal Options" in Negotiation Journal, April, 1990, Vol. 6, No. 2, pp. 161–172.

[15]Boris Galitsky, B. and Josep Lluis de la Rosa "Concept-based learning of human behavior for customer relationship management" [Information Sciences](http://www.sciencedirect.com/science/article/pii/S0020025510004007) Volume 181, Issue 10, 15 May 2011, pp 2016-2035 (2011)

# APPENDICES

# Appendix 1 -: Time schedule

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Time in Days  Activity | 60 | 25 | 45 | 10 | 5 |
| Research on Proposal, Writing, And presentation |  |  |  |  |  |
| Gathering needed System Requirements |  |  |  |  |  |
| System Development |  |  |  |  |  |
| Testing and Validation |  |  |  |  |  |
| Documentation, user manual and  Presentation |  |  |  |  |  |

# Appendix 2 -: Budget

Currency in Kenyan Shillings (Ksh.)

Personal Computer (Laptop) -: 35,000

Wi-Fi Router -: 4,000

Web Domain Name -: 3,000

Virtual Private Server -: 3,000

Miscellaneous-: -: 2,000

**Estimated Total Cost -: 47,000**