**Table of Contents**

**Table of Contents .......................................................................................................................... ii**

**Revision History ........................................................................................................................... 1**

**1. Introduction ..............................................................................................................................2**

1.1 Purpose ........................................................................................................................................ 2

1.2 Document Conventions ............................................................................................................... 2

1.3 Intended Audience and Reading Suggestions ............................................................................. 2

1.4 Project Scope ............................................................................................................................... 2

**2. Overall Description ..................................................................................................................4**

2.1 Product Perspective ..................................................................................................................... 4

2.2 Product Features .......................................................................................................................... 4

2.3 User Problem statement ............................................................................................................... 4

2.4 User Objectives ........................................................................................................................... 5

2.5 Operating Environment ...............................................................................................................5

2.6 Design and Implementation Constraints ..................................................................................... 5

2.7 User Documentation .................................................................................................................... 5

2.8 Assumptions and Dependencies .................................................................................................. 6

2.9 User Constraints .......................................................................................................................... 6

**3. System Features ........................................................................................................................6**

3.1 System Feature 1 ......................................................................................................................... 6

3.2 Stimulus/response sequence...................................................................................................... 7

3.2 Functional requirements...................................................................................................... 7

**4. External Interface Requirements ...........................................................................................12**

3.2 External interface requirwements...................................................................................................... 12

4.1 User Interfaces ............................................................................................................................. 13

4.2 Hardware Interfaces .................................................................................................................... 13

4.3 Software Interfaces ...................................................................................................................... 13

4.4 Communications Interfaces .........................................................................................................14

**5. Other Nonfunctional Requirements .......................................................................................14**

5.1 Performance Requirements ......................................................................................................... 14

5.2 Safety Requirements .................................................................................................................... 14

5.3 Security Requirements ................................................................................................................14

5.4 Software Quality Attributes ........................................................................................................ 15

5.5 Other requirements Attributes ..................................................................................................... 15

*Such attributes include, Binary compatibility, Reliability, Extensibility, Reusability Maintainability Portability, Application affinity, Recourse utilization Serviceability among others* ............................... 15

**6. Preliminary Object-Oriented Domain Analysis ....................................................................15**

6.1 Inheritance relationships ............................................................................................................. 15

6.2 User Classes and Characteristics ................................................................................................. 15

6.2.1 Abstract or Concrete ............................................................................................................... 17

6.2.2 List of Super classes ................................................................................................................ 17

6.2.3 List of subclasses ....................................................................................................................17

6.2.4 Purpose ....................................................................................................................................17

6.2.5 Collaborations .........................................................................................................................17

6.2.6 Attributes ................................................................................................................................. 17

6.2.7 Operations ............................................................................................................................... 17

6.2.8 Constraints .............................................................................................................................. 17

**7. Preliminary Budget and Schedule ..........................................................................................18**

**8. Other Requirements ................................................................................................................19**

8.1 References ................................................................................................................................... 19

8.2 Appendix A: Glossary of definitions, Acronyms and abbreviations ......................................... 19

**Appendix B: Analysis Models .......................................................................................................19**

**References…………….................................................................................................................22**

# **REVISION HISTORY**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason for Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

1. **INTRODUCTION**

This SRS Document contains the complete software requirements for the Student Attendance Management System using Fingerprint recognition (SAMS, FR) and describes the design decisions, architectural design and the detailed design needed to implement the system. It provides the visibility in the design and provides information needed for software support. New reliable and fast university management software with the great customer’s support. It'll help you with your daily university management routines and deliver you from your paperwork.

This section gives a scope description and overview of everything included in this SRS document. At first, the purpose of this document is described and a list of abbreviations and definitions is provided.

* 1. **Purpose**

The purpose of this document is to give a detailed description of the requirements for the “Student Attendance System Using Fingerprint Recognition”. It will illustrate the complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and a reference for developing the first version of the system for the development.

* 1. **Document**

Standard or typographical conventions that were followed when writing this SRS are:

* Fonts: Times new romans as the main font theme with a size of 12.
* Highlighting in bold or indentation for the references. Special heading and subheadings will be underlined.
* Main headings will be in bold and in capital letters.
* Every requirement statement has their own priority.
  1. **Intended Audience and Reading Suggestions**

Different types of reader that the document is intended for are developers, project managers, marketing staff, users, testers, and documentation writers. The rest of the document contains the requirements specification starting with the project scope and functional and non-functional requirements to a point where preparing the design document will be inevitable.

* 1. **Project scope**

The project will replace paper work involved with attendance taking. The new system is designed to solve problems affecting the manual system in use. It is design to be used thereby relieving both the students and staff from much stress as experienced in the manual system. It will make use of a scanner. This project is limited to attendance system for students from Egerton University. The software developed will be carried out using Java, Swing UI and MySQL server as the database to manage both the database and at the same time make the software work with a scanner. The system is supposed to automate the whole process of capturing data of students upon registration to the universities through direct data entry by the lecturers. This data will be securely stored in a large database for attendance. The system shall also provide data analytics technique based on data stored in the database. The large database will capture and store all student data and information. After which comparison will be made in an attempt to take the student’s attendance at a particular time. The whole process to be smooth and less costly.

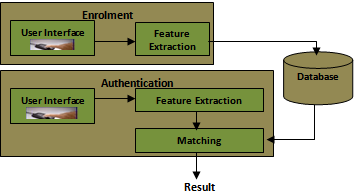
1. **OVERALL DESCRIPTION**

This section will give an overview of the whole system. The system will be explained in its context to show how the system interacts with other systems and introduce the basic functionality of it. It will also describe what type of stakeholders that will use the system and what functionality is available for each type. At last, the constraints and assumptions for the system will be presented.

* 1. **Product perspective**

This software product is eventually intended for the students and lecturers. The system will be deployed to the university and all users (lecturers and administrators) of the system will access it. It can be a web application with a single database access or a standalone system with all functionalities like an ATM. However, for the purpose of this project for the COMP493 fulfilment, the system will provide the main user interface where lecturers can operate all the provided functionality. To use the system, students’ details are fed or imported from other database system by the administrator like the Online Students Portal. All the required data will be created in the database and a predefined account will be assigned for each student. Later, each lecturer will be able to log in and out of the system anytime he/she want. Lecturers will have full privileges over the systems students’ data attendance. They will be able to perform predefined operations such as taking, acquire the student’s status and take attendance.

Diagram for the interconnection between the system and its components



The students’ fingerprints will be taken during enrollment and stored into database or file manager, after which the authenticated lecturers will take the take the attendance matching the results to that in the database thereby displaying results.

* 1. **Product Features**

This system, student attendance system, must have number of features which will allow users to use functionalities which have been explained above. Required functionalities of the product can be summarized in the following modules; Main Module, Attendance/Registration module, Student Module, and Administrator module. The lecturers are created and managed by the Systems Administrator using the Administrator module.

* 1. **User Problem statement**

Users of this student attendance system will mainly be lecturers and other university staff employees (Administrators). Since it is reasonable to assume that all lecturers have knowledge about functionalities and usage of attendance system, I assume that the users will already be informed and trained about basic functionality of the system. Besides, there will be a help tool that will assist users to better understand and use this application system. Also, a clear documentation about the system features will be provided.

* 1. **User Objectives**

The goal of the users is to have a system that will provide them with all the information they need to do the attendance. The system will therefore provide a platform that will relieve both the students and lecturers from much stress as experienced in the manual attendance system.

The users of the system are students, teachers and the administrators who maintain the system. The students are supposed to have a general knowledge of how a scanner works. Attempts will be made to make the students vie their attendance details taken manually. Lecturers are assumed to have basic knowledge of the computers. The administrators of the system are to have more knowledge of the internals of the system and is able to rectify the small problems that may arise due to scanner crashes, power failures and other catastrophes to maintain the system. The proper user interface, user’s manual, online help and the guide to install and maintain the system must be sufficient to educate the users on how to use the system without any problems.

* 1. **Operating Environment**

The system software is an application that will need a connection to a scanner. The computers will need mechanisms or network to communicate with a server that hosts both the application and the application’s database. The computer can be any device with at least 1 GB RAM and 350GB ROM.

The software specification required for this system are:

* Operating system e.g. Windows 10,7,8, Linux Mac Operating systems.
* Scanner connections
* Runtime Environment
* Server
  1. **Design and Implementation Constraints**

The system works well with:

**\***Intel core i2 processors, 2GHz (any higher version of this)

\* 4 GB ram and higher

\*Secugen hamister plus scanner/ U are U digital technology scanner

* 1. **User Documentation**

A document will be provided to guide the system users which gives them an overview of how the system works and how they are expected to use the system. This document will be part of the system. Users will be able to learn how to use the system by clicking the help button that will be included in the system. The format of the system includes:

* An overview of the system.
* Technical details such as hardware and software requirements needed to run the system.
* User guide.
* A list of technical terms.
* The system’s troubleshooting techniques.

## **Assumptions and Dependencies**

## The system largely depends on an existing management system that is being used by the university to register students, track their financial records and their academic performances. For a student to be cleared from the university he/she must be a registered member of the university, must have paid all school fees and other payments needed and must have passed in his/her field of study in order to access full accommodation and unit services. Therefore, the new system will largely depend on the data already stored in an excel document as its source of input.

## **User constraints**

## The system should be designed within the specified time frame and all the system requirements should be included as specified in the software requirement specification.

# **SYSTEM FEATURES**

# The system has three types of users, the system administrators, (super administrator and responsible staffs) lecturers and students. There are various features for each user in the system.

## SYSTEM FEATURES

### Login

This feature is the entrance part into the system. Users are first provided with this feature when they attempt to access the system. This feature enables the admin or lecturers to log into the respective using a unique username (registration number) and a password. The log in feature is of a high priority because for one to access his or her account he or she must first log into the system.

### User profile

This feature displays the basic information about the system user. For example, if the user is a lecturer, the user profile will display the general information about the user. For lecturers, this information includes full name, lecturer ID number, department, attendance record dates, email, physical address etc. The first page to be displayed after is the attendance module or the home page.

### Attendance status

This module will allow lecturers to check on the student’ attendance status. It will enable a student to tell whether he/she attended a particular lecturer or not.

### Attendance application

This will show the status of attendance. The lecturers and the admin will be able to view and manage students using their registration numbers. Mimicry of the normal attendance sheet will be used to display the module. The administrators will then approve the attendance taken as being in or out of order.

* + 1. **Manual Attendance**

The system will allow the lecturers to take attendance manually in an event a scanner is not reachable or when the fingerprint cannot be found to realize the working of the Kenyan IEBC KIEMS kit. This can be through fingerprint selection from file manager or through registration number.

### Feedback

Lecturers will be able to view attendance feedback from concerned department in this module. This however is normally possible with the online versions of a system. Administrators will be able to view attendance feedback from various departments in this module. A student will be considered attendant after a scanner approves a student.

### Reports

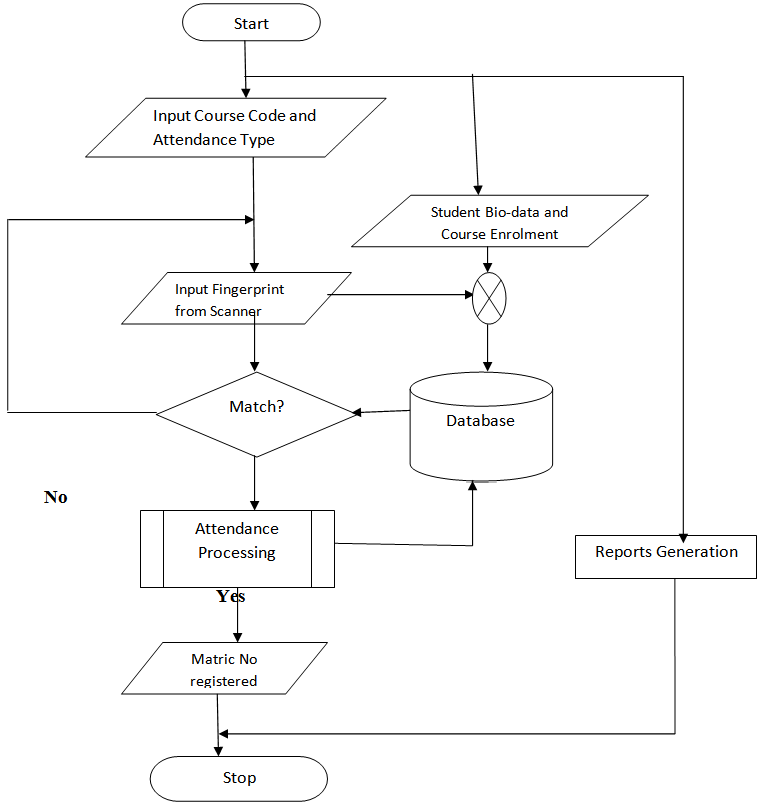
This will compute the percentages attendance of each student. One of the systems goal is automation and will be realized mostly in this module.

### Logout

Log out feature enables the admin and the lecturers to log out from the system

## STIMULUS/RESPONSE SEQUENCE

The users of the system will be interacting with the system as described in the flow diagram below.



*The lecturer will input or open a course i.e. add a class. Mark attendance using a fingerprint scanner. Matching process takes place and if found, mark attendant or otherwise. Reports will be generated eventually.*

* 1. FUNCTIONAL REQUIREMENTS

The system shall provide the following features and functionalities:

***Login***

This feature allows the users of the system to log in the system after providing the correct values of username and password. Different users who have access rights to the system can login, however lecturers have a limited number of rights over the administrators as they will check how often a lecturer have taken attendance. Without logging into the system, one is not able to use the system.

|  |  |
| --- | --- |
| Use Case No | 1 |
| Use case Name | Login |
| Actors | Lecturers  Administrators |
| Descriptions | User login |
| Pre-conditions | User must be a registered member of the system |
| Normal course of events | User enter their username  Users enter their passwords  Users click the login button  System connects to the database  User profile is displayed |
| Alternative course of events | Users can enter their names and passwords wrongly  An error message appears. |

*Functional Requirements for login in.*

***Logout***

This feature allows the users of the system to log out of the system after undertaking what one intended to do with the account.

|  |  |
| --- | --- |
| Use case No | 2 |
| Use case name | Logout |
| Actors | Lecturers  System Administrator |
| Descriptions | User logout |
| Pre-conditions | User must be logged into the system |
| Normal course of events | The system users click to logout button Assurance message appears (do you want to log out?)  The system logout the user successfully  The system goes back to the login page |

*Functional Requirements for logout.*

***Attendance application***

This use case allows Students to register for attendance. The lecturer can also modify or delete attendance taken. System administrator can view and approve attendance taken.

|  |  |
| --- | --- |
| Use Case No | 3 |
| Use case Name | Attendance application |
| Actors | Lecturers  Administrators |
| Descriptions | Make attendance application  Go through attendance process |
| Pre-Conditions | Actor must be logged in into the system.  Actor must have registered a class |
| Normal course of events | The actor must be logged on to the system as defined by the use case number 1.  The actor is able to be provide the necessary information to the system like initiating a class.  The actor approves the intended attendance.  The lecturer takes the attendance from students  The system the notifies the actor if the request is successful or not. |
| Alternative course of events | The system does not access the database or  The system does not access the scanner  The system displays an error message to the effect that it can’t access the database or scanner or file manager.  The students’ attendance to be taken manually  Details recorded |

*Functional Requirements for attendance registration*.

***View Profile***

This use case allows the lecturers and Administrators to view their profiles.

|  |  |
| --- | --- |
| Use Case No. | 4 |
| Use Case Name | View Profile |
| Actors | Lecturers  Administrator |
| Descriptions | View profile |
| Pre-conditions | Actor must be logged into the system. |
| Normal course of events | Actor logs in to the system as defined by the use case  Actor views his/her profile. |
| Alternative course of events | The system does not access the database  The system displays an error message to the effect that it can’t access the database  Lecturers and administrators cannot view their profiles. |

*Functional Requirements for view profile.*

***Admin Module***

This use case allows administrator to monitor the attendance process on a management point of view. Useful information and improvements deducted.

|  |  |
| --- | --- |
| Use Case No | 5 |
| Use case Name | Monitor the attendance process |
| Actors | System administrator |
| Descriptions | Assign lectures to various lecturers |
| Pre-conditions | The system administrator logs into the system |

|  |  |
| --- | --- |
| Normal course of events | The system administrator logs into the system.  The system administrator chooses attendance module and navigates along it. Can nullify an attendance taken |

*Functional Requirements for assign staff to departments.*

***Report Generation***

This feature helps the system administration, lecturers and the schools’ senate to get all the information they require.

* Report about the percentage attendance of each student.

This feature has got high priority since report generation is the final goal of the system. The system needs to be able to generate the above-mentioned report.

|  |  |
| --- | --- |
| Use Case No | 5 |
| Trigger | The lecturers are done with the attendance taking process |
| Pre-condition | The system actor logs into the system. |
| Basic path | The system lecturer logs into the system.  The system lecture goes to the desired module.  The system user selects or the kind of result he/she need to generate from the system.  When the results have been generated, the system displays or generates the attendance report. |
| Alternative path | None |
| Post-conditions | The students, lecturers and admin can view the attendance taken. |
| Exception paths | System environment failure  The system Lecturer exits the operation at any time. |
| Other | The type of result to be decided by the system administrator |

## 3.3 Use cases

### A. Lecturer use case

<<uses>>

<<uses>>

<<uses>>

<<uses>>

<<uses>>

<<uses>>

Lecturer

<<uses>>

### B. Admin use case

<<uses>>

<<uses>>

<<uses>>

<<uses>>

Admin

<<uses>>

## **Technical Issues**

The design of this feature should be perfect so as to enable users of the system to navigate the system.

## **Dependencies with another requirement**

The system largely depends on a manual attendance sheets that are used by the university to register students, attendance records and their academic performances deduced from the attendance. For the student attendance details to be taken, the student must be a member of that unit at the specified time, fingerprints taken and stored in the database hence attendance will be mostly based on comparison. The new system will largely depend on the normal flows of events.

Other dependencies can be to access the lecturers and student’s details from the current system and then update with fingerprint incorporation.

# **EXTERNAL INTERFACE REQUIREMENTS**

With this section and later, the requirements of the software will be described in detail. Requirements will be categorized into 3. External interface requirements, functional requirements and non-functional requirements. Except non-functional requirements, requirements of the application will be detailed under this section with brief information and later sample input-output sequence.

## **External Interface Requirements**

These are the inputs into and outputs from the software system: -

* The system shall uniquely assign a student’s unique fingerprint upon reporting the university.
* The system shall provide reports based on data stored in the database.
  + The system shall provide interface with the database which will be maintained regularly by institution database administrator.
* The system shall include security features to minimize any form of fraud and unauthorized access.
* The system shall securely store students’ attendance system in a large database
* The system shall allow for mainly fingerprint recognition attendance and manual attendance taking.

### User interfaces

All the user interfaces of the system will follow a consistent theme and clear structure. The occurrence of errors will be minimized through the use of checkboxes, radio buttons and scroll down in order to reduce the amount of text input from user. Special character checks implemented in order to provide a Data Check before submission. Tables will be used to display information to give a clear structure that easy to understand by user. Error message should be located beside the error input which clearly highlight and tell user how to solve it. If system error, it should provide the contact methods. The page should display the project process in different color to clearly reflect the various states that the system has. Each level of user will have its own interface and privilege to manage and modify the project information such as lecturer being able to monitor/manage his student progress and make comment on it, admin can further improve on the attendance details. The System should provide a detailed report for the lecturers the front end will be used by lecturers. Admin will use the front end and back end. The respective users will be taken to their respective pages on login, pages that are customized according to the functions of the users in the system or the privileges given by then system administrator. Menus and buttons will be used to interact with the users. A user logs into the system and is prompted to build their profile. There the user is given forms to register for the profile. The page gives the users rights to edit any information so that following the users of the system can be easier. The system administrator on login in have a default page allocated to him/her. The administrator is able to view all the application details about the lecturers or students.

* 1. **Hardware interfaces**

Specifically, secugen hamister plus normally used in banks will be used to develop the system functionality as it will act as a data input device to the system. Other than the normal passwords, this system will largely rely on scanners. Communication must be established between the system and the database. Various methodologies can be used on a wholescale platform but for the project, all the components are downloadable.

* 1. **Software Interfaces**

The computing devices to be used must obey the ISO OSI standards. In this regard, they should have capabilities to support communication protocols. The operating system will act as the interface between the system and the database. Due to a presence of scanner, necessary drivers will be included in the installation of the system. Windows 7/8/10 will be suitable as an interface to the system since they have user-friendly interface. Any advancement will be taken care of due to enhanced flexibility. The system will store its data resource in the MYSQL Database Management System.

### Communication Interfaces.

### Since the system is network based, modules have to communicate in a network. If the system will be accessible online (Recommended due to offline infrastructure), this will be enabled by the TCP/IP protocol. The Internetworking protocols will be assumed to apply in this instance, such as HTTPS protocol for encryption of sensitive information e.g. password while in transit.

## **OTHER NONFUNCTIONAL REQUIREMENTS**

## **Performance Requirements**

The system should have multi-user capabilities. This means that the system should support concurrent access to the database system. Therefore, the system should be designed such that such concurrency access does not slow down performance and put the database in an inconsistence mode.

## **Safety Requirements**

The system should be guarded from unauthorized accesses so that only authorized users have access. This is to maintain confidentiality on users that the integrity of the system is not compromised. This can be achieved by using firewalls and password encryption to restrict access to authorized users only. However, fingerprints are difficult to fake. In addition, data should be backed up regularly to ensure that in case of failure, the system could recover through the reconstruction of lost data.

### Security Requirements

The system should restrict users on what functions they can perform. There should be a clear-cut aggregation of the functions and their access rights. The users should also have defined tasks that they can perform in the system. Firewalls and firewall policies should be defined to protect the communication and authentication options from the client to the application server and database server. Server-side validation of user inputs should be should be considered as well as the application side to realize a double security. The system’s public and restricted areas should be separated since a public area of a system can be accessed anonymously. Restricted areas can be accessed only by specific individuals and these individuals must authenticate to use the system. By partitioning the application into public and restricted access areas, separate authentication and authorization rules can be applied across the application.

## **Reliability**

The system shall operate 95% of the time. The number of defects should not exceed 10 per function.

## **Availability**

The availability of the management system is up to the database connection of the client system. Since this is the stand-alone or server related system, the system shall be attainable all the time. User should have an account to enter the system, if the user does not have an account; for the availability of the system user should get registered from the database administration. Since it is for use in the university, password lost shall be recovered by an admin.

## **Maintainability**

The requirements, modules that are explained in this document are enough to satisfy the customers’ needs and wants. In case of a change or addition demand after completing the system or in development process of the system, a new agreement shall be done between the acquirer and the developer (Me). The maintainability shall be easily done by integrating new modules and offering new software solutions for the system.

## **Portability**

The system be made portable as possible, so the availability majorly relies on the availability of simple software and hardware.

## **Design Constraint**

All documentation of the system shall be prepared related to IEEE standards. Furthermore, the content shall be compliance with IEEE standards.

# **PRELIMINARY OBJECT-ORIENTED DOMAIN ANALYSIS**

## **Inheritance relationships**

Inheritance relationship will be in form of generalization, association or dependency.

## **User Classes and Characteristics**

* + 1. **Abstract or Concrete**

### Lecturers

Lecturers are the main users of the system and hence it will be made very user friendly. They will require no special skills to operate the system. Hence, the user interfaces are very user friendly and doesn’t require any internal interpretation.

* + 1. **List of Super classes**

# System Administrator

The System administrator is the super user of the system and has all the privileges over the system, mostly an ICT manager. He/she must have good knowledge and training about system administration. In order to limit the privileges of system administrators, compliance auditing shall be integrated into the system. Exceptions to the administrators’ privileges should be documented and approved by the appropriate chain of university authority.

Attendance

It will eventually have sub classes

* + 1. **List of subclasses**

Accept, deny, verify and connect.

* + 1. **Constraints**

Working of modules to realize a common activity will be a constraint.

1. **PRELIMINARYSCHEDULE AND BUDGET**

* Budget

The estimated budget for the project is as follows

|  |  |  |
| --- | --- | --- |
| **Expenditure Description** | **Budget Requested (Ksh)** | **Justification for Expenditures** |
| Equipment |
| Computer(Laptop) | 35, 000 | A computer is provided for in the lab (coding and testing) |
| Scanner and USB connector | 5, 000 | For taking the students’ fingerprints |
| Supplies and services |
| Hosting | 5,000 | Hosting the system |
| Risks and uncertainties | 4,000 | Recovering from any risk that might happen |
| Project evaluation | 2,000 | For testing before implementation |
| Implementation and maintenance | 4,000 | Updating the system |

* Schedule

The schedule for the project is as follows

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TASK DESCRIPTION** | **PLAN START** | | **PLAN END** | **ATTENTION** | J | F | M | A | M | J | J | A |
|  | |  |  |  |  |  |  |  |  |  |  |  |
| **Project Development** | | 01-01-18 | 01-02-18 |  |  |  |  |  |  |  |  |  |
| Preliminary Research | | 01-02-18 | 20-02-18 | Low |  |  |  |  |  |  |  |  |
| Requirements Collection | | 20-02-18 | 20-03-18 | Medium |  |  |  |  |  |  |  |  |
| Analysis | | 20-03-18 | 11-04-18 | Low |  |  |  |  |  |  |  |  |
| Design | | 11-04-18 | 01-05-18 | High |  |  |  |  |  |  |  |  |
| Coding | | 01-05-18 | 01-06-18 | High |  |  |  |  |  |  |  |  |
| Testing | | 01-06-18 | 17-06-18 | High |  |  |  |  |  |  |  |  |
| Implementation | | 17-06-18 | 30-06-18 | Low |  |  |  |  |  |  |  |  |
| **Documentation** | |  |  |  |  |  |  |  |  |  |  |  |
| Project proposal | | 01-01-18 | 01-02-18 | Medium |  |  |  |  |  |  |  |  |
| SRS | | 01-02-18 | 01-03-18 | High |  |  |  |  |  |  |  |  |
| Design Document | | 01-03-18 | 20-04-18 | High |  |  |  |  |  |  |  |  |
| Test Plan | | 20-04-18 | 13-05-18 | Medium |  |  |  |  |  |  |  |  |
| Project Document | | 13-05-18 | 14-06-18 | High |  |  |  |  |  |  |  |  |
| Implementation Plan | | 14-06-18 | 10-08-18 | Low |  |  |  |  |  |  |  |  |
|  | |  |  |  |  |  |  |  |  |  |  |  |

# **OTHER REQUIREMENTS**

# **Backup and Recovery**

# The system shall have a backup and recovery mechanism in case of failure of the system. The system should be backed up regularly to reduce the chances of data loss.

# **References**

1. SRS – template pdf by Dr. Gikaru – Egerton University.
2. N.K. Ratha, K. Karu, S. Chen, and A. K. Jain. *\A Real-Time Matching System*

*for Large Fingerprint Database".* IEEE Trans. PAMI, 18(8):799 {813, 1996.}

1. Kenneth Nilsson and Josef Bigun. *\Localization of corresponding points in \_n fingerprints by complex \_altering".* Pattern Recognition Letters 24, page 2135 2144,

October 2003.