

**FACULTY OF COMPUTING AND INFORMATION MANAGEMENT**

**BACHELOR OF INFORMATION SECURITY AND FORENSICS**

**UNIT: BISF 2208**

**PROJECT TITLE: ONLINE IQ TEST MANAGEMENT SYSTEM**

**SOFTWARE REQUIREMENTS SPECIFICATIONS DOCUMENT**

**BY: MUNENE JOHN**

**REG NO. 20/04112**

**SUPERVISOR: CATHERINE GACHIRI**

**Table of Contents**

[1.0. Introduction 3](#_Toc1195827989)

[1.1. Purpose 3](#_Toc1231584786)

[1.2 Intended Audience and Reading Suggestions 3](#_Toc1749333936)

[1.3 Project Scope 3](#_Toc2049051715)

[2. Overall Description 3](#_Toc1009799711)

[2.1 Product Perspective 3](#_Toc20141008)

[2.2 Product Features 4](#_Toc128322415)

[2.3 User Classes and Characteristics 4](#_Toc1810389646)

[2.3.1. Applicant’s use case 6](#_Toc1621972718)

[2.3.2. Company’s use case 6](#_Toc8071804)

[2.3.3 Admin use case 6](#_Toc743249976)

[2.4 Operating Environment 7](#_Toc670439455)

[2.5 Design and Implementation Constraints 7](#_Toc1253280101)

[2.7 Assumptions and Dependencies 8](#_Toc730130478)

[3. System Features 8](#_Toc1206286768)

[3.1. Description and Priority 8](#_Toc1755755428)

[3.2. Stimulus/Response (Input/Output) 8](#_Toc878346053)

[3.3. Functional Requirements 8](#_Toc471159672)

[3.0 Other features 8](#_Toc1867788220)

[CLIENT/SERVER SYSTEM 8](#_Toc1111918683)

[4. External Interface Requirements 9](#_Toc625242560)

[4.1 User Interfaces 9](#_Toc192282665)

[4.2 Hardware Interfaces 9](#_Toc993340123)

[4.3 Software Interfaces 9](#_Toc376799425)

[4.4 Communications Interfaces 10](#_Toc624098189)

[5. Nonfunctional Requirements 10](#_Toc828865399)

[5.1 Performance Requirements 10](#_Toc952687365)

[A. ER Diagram 10](#_Toc1325179391)

[B. Normalization 11](#_Toc1427189262)

[5.2 Safety Requirements 11](#_Toc1055582891)

[5.3 Security Requirements 12](#_Toc700193564)

[6.0 References 12](#_Toc475533603)

# 1.0. Introduction

## 1.1. Purpose

The purpose of this document is to present a detailed description of the Online interviewing management System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli.

1.2 Intended Audience and Reading Suggestions

This document is intended for both the developers and my supervisor of the system and will be proposed to the panel for its approval.

## 1.3 Project Scope

This software system will be a online interviewing management System for organisations to make use of it. This system will be designed to maximize the editor’s productivity by providing a system feature that will enable the company’s interviewing panel to set up questions that will help with elimination of the applicants with little understanding of the intended environment. By maximizing the interviewing panel’s work efficiency and production the system will meet the panel’s needs while remaining easy to understand and use.

More specifically, this system is designed to allow the panel to monitor and record the applicant’s details. Preformatted reply forms will be used to collect applicant’s details.

2. Overall Description

## 2.1 Product Perspective

A distributed IQ test database system stores the following information.

* Applicant’s details:

This includes applicants name, phone number and email along side the email verification.

* Applicant’s results:

This includes the applicant’s attained grades and time elapsed to do the test.

* Test and test guidelines:

It includes the test itself including and also the guide line to help the applicant on how to go about the system.

## 2.2 Product Features

The major features of the system as shown in below:

Company

Admin

Applicant 2

Applicant 1

Online Test System

Test

Online IQ Test Management System

2.3 User Classes and Characteristics

Both the company and applicant of the system should be able to retrieve the applicant’s grades and details from the database. Admin will have access to the whole system including vailidating the registered accounts. On the other hand, the company will have access to approve the applicant’s account created under their company and also will have the access of uploading questions. The company will be able to set the pass scores.

The applicants should be able to do the following functions:

* Register for an account.
  + Provide name.
  + Provide phone number.
  + provide email.
  + provide address (optional).
* Wait for their account to be approved by the company or the admin.
* Take the test.

The company should have following management functionalities:

* Register their account to the system.
  + Provide name of the company.
  + Provide business license.
  + Provide KRA
  + Provide business product and services details
* Approve the applicant’s account registered under their name.
* Add questions to the test.
* Delete questions from the test.
* Set pass scores of the test.
* Set deadline for the tests.

The admin should be able to do the following function:

* Approve the companies’ account.
* Approve the applicant’s account (if asked to do so by the company).
* Add more features to the system.
* Monitor database and delete the data that are not useful (especially for those who failed the test).

A test should have a set scores in order to differentiate between the one who qualify and the ones who don’t qualify.

### 2.3.1. Applicant’s use case

The applicant has the following use case:

Applicant

Take the IQ test

### 2.3.2. Company’s use case

The company has the following sets of use cases

Register account

Company

Approve Applicant’s acc.

Set questions

Set pass scores

Set deadline

### 2.3.3 Admin use case

Approve Co. acc.

Admin

Add Features

Monitor DB

Approve Applicant’s acc.

Provide security

## 2.4 Operating Environment

Operating environment for the IQ Test management system is as listed below.

* distributed database
* client/server system
* Operating system: Windows/unix
* database: sql+ database.
* platform: python, Django

## 2.5 Design and Implementation Constraints

1. The global schema, fragmentation schema, and allocation schema.
2. SQL commands for above queries/applications
3. How the approval for company’s account and applicant’s account will be generated. Assuming these are global queries. Explain how various fragments will be combined to do so.
4. Implement the database at least using a centralized database management system.

2.7 Assumptions and Dependencies

Let us assume that this is a onmanagement system and it is used in the following application:

* Registering new accounts and providing a friendly graphical user interface.
* Authetication of accounts with less privilegdes.
* Calculating the applicant’s grades.

# System Features

3.1. Description and Priority

This IQ Test System maintains information on the applicant’s details, and which company is registered under that applicant.Of course, this project has a high priority which is grades attained and time taken to finish the test for some interviews might be hard to choose the applicant with the highest level of knowlegde so they could use the time taken to take the test.

## 3.2. Stimulus/Response (Input/Output)

* Display the detailed test guideline to help the applicant when doing the test.
* input applicant’s name, phone number and email.
* Submit the test when done.
* Display the results in a very clear way.

## 3.3. Functional Requirements

# 3.0 Other features

## CLIENT/SERVER SYSTEM

The term client/server refers primarily to an architecture or logical division of responsibilities, the client is the application (also known as the front-end), and the server is the DBMS (also known as the back-end).

A client/server system is a distributed system in which,

* + Some sites are client sites and others are server sites.
  + All the data resides at the server sites.
  + All applications execute at the client sites.

# 4. External Interface Requirements

## 4.1 User Interfaces

* Front-end: HTML, CSS, javascript.
* Back-end: Python Django.
* Help manual: To help the user take a tour within the system and help them get full knowledge of using the system.

## 4.2 Hardware Interfaces

* Windows
* A browser which supports HTML and Javascript.

## 4.3 Software Interfaces

|  |  |
| --- | --- |
| *Software used* | *Description* |
| Operating system | We have chosen Windows as the major operating system for its best support and user-friendliness. But also there is a version compatible with unix. |
| Database | To save the applicant’s records, Companies’ records we have chosen SQL+ database. |
| Python Django | To implement the project we have chosen Django web framework for its more interactive support. |

## 4.4 Communications Interfaces

This project supports all type of browsers. The system will communicate with the apps in it i.e The applicant’s account will ask for verification of account from the companies account which has that privilegde to validate that account. I am using a simple electronic forms for the collection of data from the applicants. And using this information we get to validate the account.

5. Nonfunctional Requirements

## 5.1 Performance Requirements

The steps involved to perform the implementation of airline database are as listed below:

### ER Diagram

The E-R Diagram constitutes a technique for representing the logical structure of a database in a pictorial manner. This analysis is then used to organize data as a relation, normalizing relation and finally obtaining a relation database.

Applicant 2

System

Aplicant 1

Company

Admin

IQ Test Sys

System

Test

### Normalization

The basic objective of normalization is to reduce redundancy which means that information is to be stored only once. Storing information several times leads to wastage of storage space and increase in the total size of the data stored.

If a database is not properly designed it can give rise to modification anomalies. Modification anomalies arise when data is added to, changed or deleted from a database table. Similarly, in traditional databases as well as improperly designed relational databases, data redundancy can be a problem. These can be eliminated by normalizing a database.

5.2 Safety Requirements

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage (typically tape) and reconstructs a more current state by reapplying or redoing the operations of committed session from the backed up log, up to the time of failure.

## 5.3 Security Requirements

Security systems need database storage just like many other applications. However, the special requirements of the security market mean that vendors must choose their database partner carefully. For example the company should host this system to a hosting company that majors in security.

# 6.0 References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.