|  |
| --- |
| **SRS Knowbis** |
|  |
| **Made By:**  Khaled Redhwan – 3910083  Mohammed Muneer – 3910362  Yasser Baghdadi – 3910069  Osama Jehad – 3810222  **Under the supervision of: Dr. Abdurazzag Almiladi** |
|  |

Table of Contents

[1. Functional Requirements 2](#_Toc101016001)

[**1.1** **Super admin** 2](#_Toc101016002)

[**1.2** **Admin Functional Requirements** 2](#_Toc101016006)

[**1.3** **Instructor Functional Requirements** 3](#_Toc101016010)

[**1.4** **Student Functional Requirements** 3](#_Toc101016015)

[**1.5** **System Functional Requirements** 3](#_Toc101016031)

[2. Non-Functional Requirements 4](#_Toc101016036)

[**2.1** **Availability** 4](#_Toc101016037)

[**2.2** **Reusability** 4](#_Toc101016039)

[**2.3** **Usability** 4](#_Toc101016041)

[**2.4** **Security** 4](#_Toc101016044)

[**2.5** **Scalability** 4](#_Toc101016048)

[**2.6** **Maintainability** 4](#_Toc101016050)

[**2.7** **Flexibility** 4](#_Toc101016052)

[**2.8** **Expansibility** 4](#_Toc101016053)

# Definitions, Acronyms, and Abbreviations

## **Authentication**

### Authentication: the process or action of proving or showing something to be true, genuine, or valid.

## **Authorization**

### Authorization: the action of authorizing

## **MTTR**

### MTTR stands for: Mean Time to Repair

# Requirements colored with yellow means it has been verified with the use-case diagram

# Requirements colored with blue means it has been added by Yasser (03/05/2022)

# Requirements colored with gray means it has been shifted by Yasser (03/05/2022)

# Requirements colored with red means it has been deleted by Yasser (03/05/2022)

# Requirements colored with green means it needs to be verified

# Functional Requirements

There are five main types of functional requirements for this project, super admin, admin, instructor, student, and system.

## **Super admin**

### **The super admin shall be able to manage administrators.**

### **The super admin shall be able to do all of what the administrators can do.**

## **Admin Functional Requirements**

The admin has requirements that must be observed and implemented appropriately, which are as follows:

### **Administrators shall be able to manage categories.**

#### Administrators shall be able to create new categories, including title

#### Administrators shall be able to edit categories.

#### Administrators shall be able to remove existing categories.

### **Administrators shall be able to manage instructors’ status.**

#### Administrators shall be able to block instructors, in case of a violation.

#### Administrators shall be able to unblock instructors.

#### Administrators shall be able to verify instructors (after reviewing their academic certificates).

#### Administrators shall be able to remove verification from instructors, in case of a violation.

### **Administrators shall be able to manage courses’ status.**

#### Administrators shall be able to block courses, in case of a violation.

#### Administrators shall be able to unblock courses.

### **Administrators shall be able to rest his account’s password**

## **Instructor Functional Requirements**

Instructors has requirements that must be observed and implemented appropriately, which are as follows:

### **Instructors shall be able to manage their profiles. # We have similar one in students**

#### Instructors shall be able to edit profiles

##### Name

##### Password

##### profile picture

##### Description.

### **Instructors shall be able to manage courses.**

#### Instructors shall be able to create new courses

##### Title

##### Description

##### Brief description

##### Description

##### Image

##### Last update

##### Status

##### Requirements

##### What will learn

#### Instructors shall be able to view their uploaded courses

##### Lectures

##### Documents

##### Description

##### Enrolled students

##### Length

##### Rate.

#### Instructors shall be able to edit their existing courses.

#### Instructors shall be able to remove their existing courses.

### **Instructors shall be able to manage lectures.**

#### Instructors shall be able to create new lectures

##### Title

##### Description

##### Brief description

##### Video

##### Documents

#### Instructors shall be able to view lectures

#### Instructors shall be able to edit lectures

##### Title

##### Description

##### Brief description

##### Video

##### Documents.

#### Instructors shall be able to remove lectures

### **Instructors shall be able to manage lectures’ contents.**

#### Instructors shall be able to upload contents

##### Video

##### Documents (pdf, PowerPoint, word)

#### Instructors shall be able to view contents

#### Instructors shall be able to edit contents

#### Instructors shall be able to remove contents

### **Instructors shall be able to upload academic certificates (to get verified).**

## **Student Functional Requirements**

Students has requirements that must be observed and implemented appropriately, which are as follows:

### **Students shall be able to manage their profile**

#### Students shall be able to edit their profile

##### Name

##### Password

### **Students shall be able to view courses**

##### Title # I believe all created attributes will be viewed

##### Description

##### Brief Description

##### Length

##### Instructor

##### Enrolled students

##### Rate

### **Students shall be able to view lectures**

### **Students shall be able to search for courses.**

### **Students shall be able to view lists of all courses uploaded on the system. # Is it view courses?**

### **Students shall be able to manage their enrolled courses**

#### Students shall be able to enroll in courses.

#### Students shall be able to view enrolled in courses.

#### Students shall be able to drop courses.

#### Students shall be able to rate courses (like or dislike)

### **Students shall be able to view instructors’ profiles**

##### Name

##### Major

##### Description

##### Rate

##### Uploaded courses

## **System Functional Requirements**

The system needs requirements that must be followed and implemented to achieve the objectives of the system, which are:

### **The system shall be able to send notifications.**

### **The system shall be able to send verification emails to students when they register.**

### **The system shall be able to send resetting password emails to students when they attempt to reset their passwords.**

### **The system shall be able to send a notification of violation to instructors when they violate the regulations.**

# Non-Functional Requirements

It is important to define the non-functional requirements that define the criteria that can be used to judge the operation of the system, rather than specific behaviors. Research conducted in January 2020 concluded that in Agile software projects, more than 77% of respondents consider having non-functional requirements to be “important” at the least; moreover, above 30% believe that it is “critical”. Another finding worth mentioning is that, overall, respondents with a greater experience value non-functional more [10].

## **Availability**

### **The system should be available at least 98% of the time after deployment.**

### The system should alert the admin in case of defect or failure

## **Reusability**

### **At least half of the system components must be reusable, where they can be used in other systems. (وفسر الماء بعد الجهد بالماء)**

## **Usability**

### **The average user shall report the system to be (highly familiar).**

### **Average user can view lectures in less than 25 seconds starting from arriving to the home page.**

## **Security**

### **Security Functional Requirements**

#### The system shall be able to provide authenticate users

#### The system shall be able to provide authorize users

#### Students shall be able to log into the system via third parties...

##### Google authentication.

##### Facebook authentication.

##### GitHub authentication.

#### Students shall be able to log out of the system.

#### Students shall be able to reset their password.

#### Validate username and password inputs

#### Encrypt sensitive data transmitted over the internet (username, password)

#### The system shall detect and hinder Bots and IP Throttling attacks.

#### The system shall prevent brute force

#### The system shall detect and prevents users from using breached passwords

### **Security Non-Functional Requirements**

#### Keeping detailed logs of authentication system to support forensic investigation

#### The system must give a confirmation message every time a user login

#### Emails should be sent with a latency of no more than 3 hours after logging in

#### The encryption key size must be 256 bits

#### Passwords should be masked as user type in

#### User password should be at least 8 characters long

## **Scalability**

### **The system must be scalable to support at least 3000 concurrent users in the time of launching, and no less than 6000 users during the first year.**

## **Maintainability**

### **The system shall exhibit a MTTR of not more than 4 hours.**

## **Flexibility**

## **Expansibility**