Cairo University  
Faculty of Computers and Information



**CS251**

**Software Engineering I**

E-Library

Software Requirements Specifications

Team Names

10/2016

Contents

[Instructions [To be removed] 3](#_Toc402452668)

[Team 3](#_Toc402452669)

[Document Purpose and Audience 3](#_Toc402452670)

[Introduction 3](#_Toc402452671)

[Software Purpose 3](#_Toc402452672)

[Software Scope 4](#_Toc402452673)

[Definitions, acronyms, and abbreviations 4](#_Toc402452674)

[Requirements 4](#_Toc402452675)

[Functional Requirements 4](#_Toc402452676)

[Non Functional Requirements 4](#_Toc402452677)

[System Models 5](#_Toc402452678)

[Use Case Model 5](#_Toc402452679)

[Use Case Tables 5](#_Toc402452680)

[Ownership Report 6](#_Toc402452681)

[Policy Regarding Plagiarism: 6](#_Toc402452682)

# Document Purpose and Audience

* Formal document that describes the Software Requirements Specifications of the E-Library.
* This document targets Scrum Manager and Customer.

# Introduction

## Software Purpose

* The purpose of the Smart Home Board is to monitor home appliances through a user friendly interface.

## Software Scope

* **The application will give users access to all courses material available at FCI.**
* **The Software will be available on Android Devices.**
* **Users will be able to save materials to their accounts and access them offline.**
* **Only admin users will be able to add/edit/remove materials.**
* **Admin Users will have the ability to add/remove admin users on system.**

## Definitions, acronyms, and abbreviations

NULL.

# Requirements

## Functional Requirements

* The user has to first create an account to login.
* The user can search for available courses and choose specific material to save to his account.
* The user can subscribe to courses to get all the latest updates.
* The user can rate materials according to their liking or add comments below each one.
* The user can save any material to his saved tab and access them offline.
* The home page will show icons for all the available functions of the software.
* The user can sort his saved material according to his preferences.
* The user can update his profile info.
* The user can logout from his account and sign in with another one.
* The user can send an upload request to upload a material and should wait or acceptance.
* The admin user can add material directly to the database in any course.
* The admin user can delete any unsuitable or out of date material.
* The admin user can create or delete courses.
* The admin user can add or remove another admin user.
* The admin user can edit any material.
* The admin user can approve upload request requested from a normal user.

## Non Functional Requirements

**Usability:**

* User must have an account to access the Application.
* User will be able to access application functions easily and shortly.
* The admin user can create more than one course.
* The user can subscribe to more than one course.
* The user can ask to upload more than one material but one at a time.
* The user can save finite number of materials.

**Reliability:**

* In case of system failure the user will have to restart the application.
* System won’t remove any of the saved materials or cached files in case of system failure.
* System will save all user details within his account in case the system broken down.
* System will backup data automatically every week.
* System should send a crash report in case of application crashing.
* System should restart automatically and restore the previous state in case of application crash.

**Performance:**

* Application launching should not take more than 3 seconds 90% of the time.
* Search process should take less than 3 seconds to get the results.
* User should access saved material in less than 4 clicks.
* Response to any user action should be smooth and don’t take more than 1 sec.
* Upload request should not take more than 5 days to have a response.

**Security:**

* Access differs depending on the type of user whether an Admin or a User.
* Users cannot modify other user’s information
* Only admin users can access the upload requests.

**Privacy:**

* Only the user can see his profile information.
* Only user can access his saved material and subscribed courses
* Only admin users can review users modify reports.

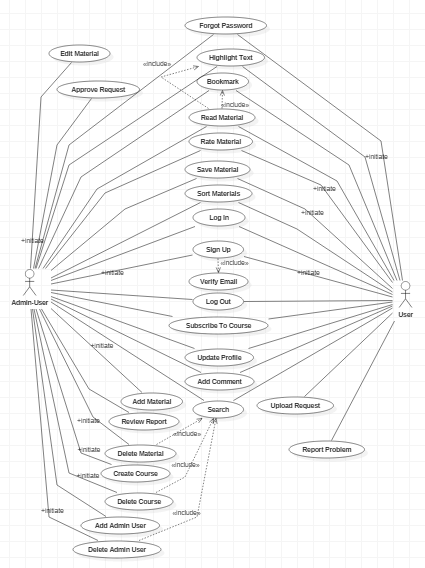
**Scalability:**

* System will support all office extensions and pdfs

# 

# System Models

## Use Case Model



1

## Use Case Tables

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | 1 | | |
| Use Case Name: | Log In | | |
| Actors: | User | | |
| Pre-conditions: | User launches the application | | |
| Post-conditions: | User successfully enters his home page. | | |
| Flow of events: | **User Action** | **System Action** | |
| 1- User enters username and password. |  | |
|  | 2- System checks if the username and password match a valid account. | |
|  | 3- System allows the user to his home page. | |
| Exceptions: | **User Action** | **System Action** | |
| 1- User enters an invalid username/e-mail |  | |
|  | 2- System rejects the username/email and notifies the user to enter a valid one. | |
|  | 3- User enters an incorrect password. |  | |
|  |  | 4- System rejects the password and notifies the user to enter it again. | |
| Includes: | Null | | |
| Notes and Issues: | Null | | |
| Use Case ID: | 2 | | |
| Use Case Name: | Log Out | | |
| Actors: | User /Admin-User | | |
| Pre-conditions: | User/Admin-User clicks on logout button | | |
| Post-conditions: | System successfully logged the user out of his account | | |
| Flow of events: | **User Action** | **System Action** |
| NULL |  |
| Exceptions: | **User Action** | **System Action** |
| NULL |  |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 3 | |
| Use Case Name: | Sign Up | |
| Actors: | User | |
| Pre-conditions: | User Launches application  User have Internet access  User have a Valid Email address | |
| Post-conditions: | Verify email address use-case (ID:15) | |
| Flow of events: | **User Action** | **System Action** |
| 1-User fills the sign up form and clicks submit |  |
|  | 2- System checks the validity of this information |
|  | 3- System Verify email address use-case (ID:15) |
| 4-User verifies email address |  |
|  |  | 5- System notifies user that the account was created successfully |
| Exceptions: | **User Action** | **System Action** |
| 1- User enters invalid information or invalid email address. |  |
|  | 2- System rejects this form and notifies user to enter a valid one. |
|  | 1-User enter password less than 8 characters |  |
|  |  | 2-Systems rejects password and ask user to retry again. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 4 | |
| Use Case Name: | Add Material | |
| Actors: | Admin User | |
| Pre-conditions: | User launches the application | |
| Post-conditions: | User enters the admin home page  User successfully adds the desired material | |
| Flow of events: | **User Action** | **System Action** |
|  | 1- Admin clicks on the (add material) icon. |  |
|  |  | 2- System displays the list of courses available and asks the user to choose one to add the material to. |
|  | 3- Admin chooses the desired course. |  |
|  |  | 4- System displays the types of materials available to be added. |
|  | 5- Admin chooses the type of material he wants to add. |  |
|  |  | 6- System asks the admin to upload the file. |
|  | 7- Admin uploads the desired file. |  |
|  |  | 8- System confirms the upload and requests the database to add it. |
| Exceptions: | **User Action** | **System Action** |
| 1- Admin tries to upload a file that mismatches the chosen data type. |  |
|  | 2- System notifies the admin that this file doesn’t match a data type of the chosen. |
|  |  | 3- System asks the admin to choose the proper type and upload the file again. |
|  | 1-Admin uploads file. |  |
|  |  | 2-System alerts user that it already exists. |
| Includes: | Null | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 5 | |
| Use Case Name: | Delete Material | |
| Actors: | Admin User | |
| Pre-conditions: | User launches the application | |
| Post-conditions: | User enters the admin home page  User successfully deletes the desired material | |
| Flow of events: | **User Action** | **System Action** |
|  | 1- Admin opens the courses page. |  |
|  |  | 2- System displays the list of courses available and asks the user to choose one to delete the material from. |
|  | 3- Admin chooses the desired course |  |
|  |  | 4- System displays the list of files available in the course. |
|  | 5- Admin chooses the desired file and presses delete. |  |
|  |  | 6- System confirms the deletion and request the database to remove it too. |
| Exceptions: | **User Action** | **System Action** |
|  | Null |  |
| Includes: | <<Search>> | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 6 | |
| Use Case Name: | Create Course | |
| Actors: | Admin User | |
| Pre-conditions: | User Launches application  User have Internet access  User Click Create Course button | |
| Post-conditions: | System successfully created a new course | |
| Flow of events: | **User Action** | **System Action** |
| 1-User enters required info about the course and any materials if available. |  |
|  | 2- System check the validity of this information |
|  | 3- System upload the course materials to database |
|  | 4- System notifies user that it successfully created a new course |
| Exceptions: | **User Action** | **System Action** |
| 1- User enters invalid information or missing info |  |
|  | 2- System rejects this form and notifies user to enter a valid one. |
| 1-User pressed create button with no response from the server |  |
|  | 2-system notify the user with connection error and retry again |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 7 | |
| Use Case Name: | Delete Course | |
| Actors: | Admin User | |
| Pre-conditions: | User Launches application  User have Internet access  User find the course  User Click delete Course button | |
| Post-conditions: | System successfully deleted the course | |
| Flow of events: | **User Action** | **System Action** |
| 1-User opens specified course and clicks on delete course. |  |
|  | 2- System alerts the user if it’s a final decision |
| 3- User clicks confirms |  |
|  | 4- System will delete the course materials from database and notifies user |
| Exceptions: | **User Action** | **System Action** |
| 1-User pressed delete button with no response from the server |  |
|  | 2-system notify the user with connection error message and to retry again |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 8 | |
| Use Case Name: | Edit Material | |
| Actors: | Admin User | |
| Pre-conditions: | User launches the application and logs in | |
| Post-conditions: | User enters the admin home page  User successfully edits the desired material’s information. | |
| Flow of events: | **User Action** | **System Action** |
|  | 1- Admin clicks on courses page. |  |
|  |  | 2- System displays the list of courses available and asks the user to choose one to edit in. |
|  | 3- Admin chooses the desired course. |  |
|  |  | 4- System displays the list of files available in the course. |
|  | 5- Admin chooses the desired file and presses edit. |  |
|  |  | 6- Systems shows the information of the chosen file. |
|  | 7- Admin edits either the name or the description or both. |  |
|  |  | 8- System updates the file’s information and request the database to do the same. |
| Exceptions: | **User Action** | **System Action** |
|  | 1-Admin types a file’s name that is already existing. |  |
|  |  | 2- System notifies the admin that there is a file already existing with the same name. |
|  |  | 3- System asks the user to type the name again. |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 9 | |
| Use Case Name: | Rate Material | |
| Actors: | User | |
| Pre-conditions: | User have an account and is signed in | |
| Post-conditions: | User rating got submitted  User exits the application. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User clicks on a material (book / notes / exams) |  |
|  | 2-System opens the material page |
| 3- User rate the material by clicking on the stars. |  |
|  |  | 4- System updates the material’s rating in database. |
| Exceptions: | **User Action** | **System Action** |
| \Includes: | NULL | |
|  | | |
| Use Case ID: | 10 | |
| Use Case Name: | Save Material | |
| Actors: | User | |
| Pre-conditions: | User logged in to his account | |
| Post-conditions: | System successfully added the material to the user’s account. | |
| Flow of events: | **User Action** | **System Action** |
| 1. User clicks on save icon next to the material |  |
|  |  | 1. System confirm with user that he wishes to add it to his account. |
|  | 1. User confirm |  |
|  |  | 1. System adds material to user’s account. |
| Exceptions: | **User Action** | **System Action** |
| 1. User clicks on save icon |  |
|  |  | 1. System notifies user that material is already in his account. |
|  | 1. User clicks on save icon |  |
|  |  | 1. Connection Error message is displayed. System asks user to retry again. |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 11 | |
| Use Case Name: | Add Admin | |
| Actors: | Admin-User | |
| Pre-conditions: | Admin-User clicks Edit Admin-list | |
| Post-conditions: | Admin-User successfully added an Admin  Admin-User returned to home page | |
| Flow of events: | **User Action** | **System Action** |
| 1-User clicks on Edit Admin-list option and chooses add Admin. |  |
|  | 2-System displays search page for the user to enter the email address of the other user. |
| 3-User enters email address of the admin to be. |  |
|  | 4-System displays the user with that email address. |
|  | 5-User clicks on add as an Admin |  |
|  |  | 6-System notifies that user was successfully added. |
| Exceptions: | **User Action** | **System Action** |
| 1- User enters email address. |  |
|  | 2-System doesn’t find any user with that email address and asks User to enter another one. |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 12 | |
| Use Case Name: | Remove admin | |
| Actors: | Admin User | |
| Pre-conditions: | Admin user clicks Edit Admin-list | |
| Post-conditions: | Admin user successfully removed Admin  Admin user returns to the home page | |
| Flow of events: | **User Action** | **System Action** |
| 1-User launches the application. |  |
|  | 2- System displays the home page to the user. |
| 3- User clicks on settings and chooses Edit Admin-list. |  |
|  | 4- System displays Admin-list to the user. |
|  | 6- User chooses the Admin he wants to remove. |  |
|  |  | 7- System removes the selected Admin and notifies the user that Admin was successfully removed. |
| Exceptions: | **User Action** | **System Action** |
| 1-User chooses to remove Admin and there aren’t any. |  |
|  | 2-System notifies that the admin-list is empty. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 13 | |
| Use Case Name: | Add Comment | |
| Actors: | User | |
| Pre-conditions: | User logged in to his account and opened a material or course. | |
| Post-conditions: | System successfully added user’s comment. | |
| Flow of events: | **User Action** | **System Action** |
| 1. User clicks on course / material inside course. |  |
|  |  | 2-System opens the course / material page. |
|  | 3- User types his comment in the comment section and clicks on Submit. |  |
|  |  | 4-System adds the user’s comment to the comment section. |
| Exceptions: | **User Action** | **System Action** |
| 1- User types his comment in the comment section and clicks on Submit. |  |
|  |  | 2- Connection Error message is displayed. System asks user to retry again. |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 14 | |
| Use Case Name: | Subscribe to Course | |
| Actors: | User | |
| Pre-conditions: | User logged in to his account and opened a course page. | |
| Post-conditions: | System successfully subscribed the user to the course and user received an email. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User opens a course available and clicks on Subscribe button. |  |
|  |  | 2- System notifies user that he successfully subscribed to course. |
|  |  | 3- System sends an email to user with confirmation |
|  |  |  |
| Exceptions: | **User Action** | **System Action** |
| 1- User opens a course available and clicks on Subscribe button. |  |
|  |  | 1. System notifies user the course is already subscribed to. |
|  | 1- User opens a course available and clicks on Subscribe button. |  |
|  |  | 1. Connection Error message is displayed. System asks user to retry again. |
| Includes: | NULL | |
| Notes and Issues: | NULL | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 15 | |
| Use Case Name: | Verify email address | |
| Actors: | User | |
| Pre-conditions: | Sign up use-case (ID:3) | |
| Post-conditions: | User successfully submitted Valid Email address  System successfully linked account to the given email address | |
| Flow of events: | **User Action** | **System Action** |
| 1-User inputs a valid email address |  |
|  | 2- System checks the validity of this address |
|  | 3- System sends activation code to this email address |
| 4- User opens the activation code found in the email sent to him. |  |
|  |  | 5- System links the email address to the desired account and print verification message. |
| Exceptions: | **User Action** | **System Action** |
| 1- User enters invalid email address. |  |
|  | 2- System rejects email address and notifies user to enter a valid one. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 16 | |
| Use Case Name: | Update profile | |
| Actors: | Admin-User/ User | |
| Pre-conditions: | User Launches application  User have Internet access  User clicks edit account button | |
| Post-conditions: | User successfully submitted correct information in replaced fields  System successfully update user info | |
| Flow of events: | **User Action** | **System Action** |
| 1-User updates desired info fields and click submit. |  |
|  | 2- System verifies user information and updates the info fields in database. |
|  | 3-System will return success message. |
| Exceptions: | **User Action** | **System Action** |
| 1- User enters invalid info |  |
|  | 2- System rejects info and notifies user to enter a valid one. |
|  | 1-User pressed update button with no response from the server. |  |
|  |  | 2-system notify the user with Connection error and to retry again. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 17 | |
| Use Case Name: | Search for material | |
| Actors: | Admin-User/ User | |
| Pre-conditions: | User Launches application  User have internet access  User opens search bar | |
| Post-conditions: | System returns search results | |
| Flow of events: | **User Action** | **System Action** |
| 1-User enters a string in search bar and pressed search. |  |
|  | 2-System checks for live internet connection |
|  | 3- System search for the string in the database and return search results to the user. |
| Exceptions: | **User Action** | **System Action** |
| 1- User pressed search button with no internet connection |  |
|  | 2- System should return an error and notifying the user to check connection |
|  | 1- User entered an empty string or string with no match |  |
|  |  | 2-Syetem should notify the user with the proper message |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 18 | |
| Use Case Name: | Upload request | |
| Actors: | User | |
| Pre-conditions: | User Launches application | |
| Post-conditions: | Upload request is send successfully | |
| Flow of events: | **User Action** | **System Action** |
| 1-User should fill upload form and fill all required fields |  |
|  | 2- system checks for the validity of the info |
|  | 3- System saves the request to be reviewed by one of the admins |
|  | 4- System returns confirmation message. |
| Exceptions: | **User Action** | **System Action** |
| 1- User entered invalid info or missing fields |  |
|  |  | 2- System should respond with proper message to the user |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 19 | |
| Use Case Name: | Sort saved materials | |
| Actors: | Admin-User/ User | |
| Pre-conditions: | User Launches application  User have more than one item in saved material area | |
| Post-conditions: | Materials sorted in the requested format | |
| Flow of events: | **User Action** | **System Action** |
| 1-User opens his saved items and chooses one of the sorting categories available |  |
|  | 2- System sorts the items and return resulted sorted material |
| Exceptions: | **User Action** | **System Action** |
|  |  |
| NULL |  |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 20 | |
| Use Case Name: | Forgot Password | |
| Actors: | User | |
| Pre-conditions: | User cannot sign in and clicks on forgot password | |
| Post-conditions: | User successfully changed his password | |
| Flow of events: | **User Action** | **System Action** |
| 1- User enters username and password. |  |
|  | 2-System rejects the password and notifies the user to enter it again. |
|  | 3- User enter the password more than 7 times incorrect |  |
|  |  | 4- System asks the user if he/ she forgot his/her password. |
|  | 5- User chooses yes. |  |
|  |  | 6- System asks the user to enter his email. |
|  | 7- User enters his email. |  |
|  |  | 8- System sends a password reset link to the entered email. |
|  | 9- User enters his new password. |  |
|  |  | 10- System sends a verification email that the password was successfully changed. |
|  | 11- User tries to log in again. |  |
| Exceptions: | **User Action** | **System Action** |
| 1- User enters an e-mail address |  |
|  | 2- System rejects the email and notifies the user to enter a valid one. |
| Includes: | Null | |
| Notes and Issues: | Null | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 21 | |
| Use Case Name: | Approve Request | |
| Actors: | Admin-User | |
| Pre-conditions: | User Launches application  User have Internet access | |
| Post-conditions: | System successfully uploaded requested material | |
| Flow of events: | **User Action** | **System Action** |
| 1-Admin opens pending requests page and review each one |  |
| 2-Admin chooses which requests he wants to add and clicks approve |  |
|  | 3-System alerts user which confirmation or to cancel approval |
|  | 4-Admin clicks on confirm. |  |
|  |  | 5-System displays list of available courses |
|  | 6-Admin chooses which course to add the material to |  |
|  |  | 7-System notifies user that material was added successfully and updates course page and database. |
| Exceptions: | **User Action** | **System Action** |
| 1-Admin approve request |  |
|  | 2-system notify the user with Connection error and to retry again. |
|  | 1-Admin approve request and chooses which course |  |
|  |  | 2-system notifies user that material already exists |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 22 | |
| Use Case Name: | Read Material | |
| Actors: | User | |
| Pre-conditions: | User launches the application | |
| Post-conditions: | User browses his downloaded files  User reads his own material | |
| Flow of events: | **User Action** | **System Action** |
| 1- User enters username and password. |  |
|  | 2- System checks if the username and password match a valid account. |
|  | 3- System allows the user to his/her home page. |
|  | 4- User clicks on the (Saved Material) icon. |  |
|  |  | 5- System displays the list of materials saved by the user according to his/her preferences. |
|  | 6- User uses the search option to find the file he/she wants to read. |  |
|  |  | 7- System loads the search results of the matched items on the screen. |
|  | 8- User chooses the file he /she wants to read. |  |
|  |  | 9- System loads the selected file on the screen. |
| Exceptions: | **User Action** | **System Action** |
|  | 1- User searches for a file not available in his materials’ list. |  |
|  |  | 2- System notifies the user that this file is no longer available in his cache. |
|  |  | 3- System asks the user to re-download the required file. |
| Includes: | <<Search>>, <<highlight text>> , << bookmark page>> | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 23 | |
| Use Case Name: | Bookmark Page | |
| Actors: | User | |
| Pre-conditions: | User launches the application | |
| Post-conditions: | User opens a downloaded book  User successfully bookmarks the desired pages | |
| Flow of events: | **User Action** | **System Action** |
| 1- User clicks on the (Saved Material) icon. |  |
|  | 2- System displays the list of materials saved by the user according to his/her preferences. |
| 3- User uses the search option to find the file he/she wants to read. |  |
|  |  | 4- System loads the search results of the matched items on the screen. |
|  | 5- User chooses the file he /she wants to read. |  |
|  |  | 6- System loads the file on the screen. |
|  | 7- User clicks on the comments option in the task bar. |  |
|  |  | 8- System displays the options that goes under. |
|  | 9- User clicks on (bookmark page) option. |  |
|  |  | 10- System saves the page to the bookmarked items. |
| Exceptions: | **User Action** | **System Action** |
|  | 1- User tries to bookmark a page that is already bookmarked. |  |
|  |  | 2- System notifies the user that the page is already bookmarked. |
|  | 3- User chooses either to unmark the page or keep it bookmarked. |  |
|  |  | 4- System saves the user’s choice. |
| Includes: | <<Search>>, <<Read Material>> | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 24 | |
| Use Case Name: | Highlight Text | |
| Actors: | User | |
| Pre-conditions: | User launches the application | |
| Post-conditions: | User opens a downloaded book.  User successfully highlights the desired text. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User clicks on the (Saved Material) icon. |  |
|  | 2- System displays the list of materials saved by the user according to his/her preferences. |
| 3- User uses the search option to find the file he/she wants to read. |  |
|  |  | 4- System loads the search results of the matched items on the screen. |
|  | 5- User chooses the file he /she wants to read. |  |
|  |  | 6- System loads the file on the screen. |
|  | 7- User selects the required text to highlight. |  |
|  | 8- User clicks on the comments option in the task bar. |  |
|  |  | 9- System displays the options that goes under. |
|  | 10- User clicks on the (Highlight) option. |  |
|  |  | 11- System highlights the text and automatically saves the change. |
| Exceptions: | **User Action** | **System Action** |
|  | Null |  |
| Includes: | <<Search>>, <<Read material>> | |
| Notes and Issues: |  | |
| Use Case ID: | 25 | |
| Use Case Name: | Report Problem | |
| Actors: | User | |
| Pre-conditions: | User Launches application  User have Internet access | |
| Post-conditions: | System successfully send a report to the servers | |
| Flow of events: | **User Action** | **System Action** |
| 1-User press report button |  |
|  | 2- System opens report form |
| 3-User should fill the report form |  |
|  |  | 4-System must check that all required filled are filled |
|  |  |  |
|  | 6-User should submit report |  |
|  |  | 7-System should confirm the report or show an error message |
| Exceptions: | **User Action** | **System Action** |
| 1-User submit a wrong information. |  |
|  | 2-system notify the user with error and to retry again. |
| Use Case ID: | 26 | |
| Use Case Name: | Review Report | |
| Actors: | Admin-users | |
| Pre-conditions: | User Launches application  User have Internet access | |
| Post-conditions: | Admin reviewed the report | |
| Flow of events: | **User Action** | **System Action** |
| 1-Admin-user should open reviews tab |  |
|  | 2- System opens categorized report |
| 3-Admin-user should review the reports |  |
|  | 4-Admin-user should interact with this report |  |
|  |  | 5-System should notify the users reported on this material about the modification |
| Exceptions: | **User Action** | **System Action** |
| NULL |  |
|  | NULL |

# 

# Ownership Report

|  |  |
| --- | --- |
| **Item** | **Owners** |
|  |  |
|  |  |