SRS cmp203 Version: 1.0



**TEAM 8**

|  |  |  |
| --- | --- | --- |
| **Ammar ALsayed** | **Sec:2** | **BN:4** |
| **Loai Ali** | **Sec:2** | **BN:11** |
| **Mahmoud Yousri** | **Sec:2** | **BN:22** |
| **Ramy Mohamed Saied** | **Sec:1** | **BN:21** |

**Smart Lecture**

**Software Requirements Specification (SRS)**

**Team 8**

**Version: 1.0**

**CMP.SEM.8**

SRS cmp203 Version: 1.0

**Distribution list**

|  |  |  |
| --- | --- | --- |
| **Name** | **E-mail** | **Notes** |
| DR:Ahmed Hamdy |  | Feedback Needed |
| TA:Dina AlReedy | dinaelreedy@gmail.com | Feedback Needed |
| TA:Ali ElSeddek | alielseddeek@gmail.com | Feedback Needed |

SRS cmp203 Version: 1.0

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | **Table of Contents** |  |
| [1.](#page5) |  | [Introduction](#page5) | | | | [5](#page5) |
|  | [1.1](#page5) | |  | [Purpose](#page5) | | [5](#page5) |
|  | [1.2](#page5) | |  | [Scope](#page5) | | [5](#page5) |
|  | [1.3](#page5) | |  | [References](#page5) | | [5](#page5) |
|  | [1.4](#page5) | |  | [Overview](#page5) | | [5](#page5) |
| [2.](#page6) |  | [Market Survey](#page6) | | | | [6](#page6) |
|  | [2.1](#page6) | |  | [Attendance taker](#page6) | | [6](#page6) |
|  | [**1.1**](#page6) | | |  | [**Project Description**](#page6) | [6](#page6) |
|  | [**1.2**](#page6) | | |  | [**Functional Specifications**](#page6) | [6](#page6) |
|  | [**1.3**](#page6) | | |  | [**Non-Functional Specifications**](#page6) | [6](#page6) |
|  | [2.2](#page7) | |  | [My Attendance Tracker](#page7) | | [7](#page7) |
|  | [**2.2.1**](#page7) | | |  | [**Project description**](#page7) | [7](#page7) |
|  | [**2.2.2**](#page7) | | |  | [**Functional Specifications**](#page7) | [7](#page7) |
|  | [2.2.3](#page7) | | | [Non-Functional Specifications](#page7) | | [7](#page7) |
|  | [**2.3.1**](#page8) | | |  | [**Functional Specifications**](#page8) | [8](#page8) |
|  | [**2.3.2**](#page8) | | | | [**Essential Specifications**](#page8) | [8](#page8) |
|  | [**2.3.3**](#page8) | | | | [**Non-Essential Specifications**](#page8) | [8](#page8) |
|  | [**2.3.4**](#page8) | | |  | [**Non-Functional Specifications**](#page8) | [8](#page8) |
|  | [**2.3.5**](#page8) | | |  | [**Limitations**](#page8) | [8](#page8) |
| [3.](#page9) |  | [Cost Estimate](#page9) | | | | [9](#page9) |
| [4.](#page9) |  | [Feasibility Study](#page9) | | | | [9](#page9) |
| [5.](#page10) |  | [Requirements Specifications](#page10) | | | | [10](#page10) |
|  | [5.1](#page10) | |  | [User Requirements (User Stories)](#page10) | | [10](#page10) |
|  | [5.2](#page12) | |  | [System Requirements (Use Cases)](#page12) | | [12](#page12) |
|  | [5.2.1](#page13) | | |  | [UCO](#page13) | [13](#page13) |
|  | [5.2.2](#page13) | | |  | [UC1](#page13) | [13](#page13) |
|  | [5.2.3](#page14) | | |  | [UC2](#page14) | [14](#page14) |
|  | 5.2.4 | | |  | UC3 | 14 |
|  | [5.2.5](#page15) | | |  | [UC4](#page15) | [15](#page15) |
|  | [5.2.6](#page15) | | |  | [UC5](#page15) | [15](#page15) |
|  | [5.2.7](#page15) | | |  | [UC6](#page15) | [15](#page15) |
|  | [5.2.8](#page15) | | |  | [UC7](#page15) | [15](#page15) |
|  | [5.2.9](#page16) | | |  | [UC8](#page16) | [16](#page16) |
|  | [5.2.10](#page17) | | | | [UC9](#page17) | [17](#page17) |
|  | [5.2.11](#page17) | | | | [UC10](#page17) | [17](#page17) |
|  | 5.2.12 | | | | UC11 | 19 |
|  |  | 5.2.13 | | | UC12 | 19 |
|  |  |  |  | |  |  |
|  |  | [5.2.14](#page19) | | | [UC13](#page19) | [19](#page19) |

|  |  |  |  |
| --- | --- | --- | --- |
| SRS |  | cmp203 | Version: 1.0 |
| 5.2.15 | UC14 |  | 20 |
| 5.2.16 | UC15 |  | 21 |
| [5.2.17](#page20) | [UC16](#page20) |  | [20](#page20) |
| [5.2.18](#page21) | [UC17](#page21) |  | [21](#page21) |
| [5.2.19](#page21) | [UC18](#page21) |  | [21](#page21) |
| [5.2.20](#page22) | [UC19](#page22) |  | [22](#page22) |

|  |  |  |  |
| --- | --- | --- | --- |
|  | [5.3](#page23) | [Non-functional Requirements](#page23) | [23](#page23) |
| [6.](#page23) | [High level plan](#page23) | | [23](#page23) |
| [7.](#page23) | [Supporting Information](#page23) | | [23](#page23) |
| [8-Version Control System](#page25) | | | [25](#page25) |

SRS cmp203 Version: 1.0

**Software Requirements Specification**

1. **Introduction**

**1.1 Purpose**

The purpose of this document is to present a detailed description of Smart Lecture application. 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. This document is primarily intended as a reference of the first version of “Smart Lecture” .This document is intended for both the stakeholders and the developers of the system

**1.2 Scope**

The “Smart Lecture” is a network mobile application that helps lecturers to manage the lecture and save time by taking attendance which is done manually and students communicate easy during the lecture.

**1.3 References**

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

**1.4 Overview**

The remainder of this document includes six chapters .The second chapter includes information about products that are similar to “Smart Lecture”, the third one briefly discusses the cost estimate of implementing the system and deploying it.

The fourth chapter goes over the feasibility study of this project from different aspects.

The fifth provides the requirement specification in detail and a description of the different system interfaces, the sixth one goes over a schedule of all iterations of the project.

SRS cmp203 Version: 1.0

**2. Market Survey**

**2.1 Attendance taker**

**1.1 Project Description**

Perform attendance tracking for any type of event using your android device (tablet or phone)!

Attendance Tracker has been created to ease keeping track of the attendance for any type of event. It fully integrates with your android device:

Import contacts from your contact list or from Google Docs Spreadsheets. Backup and restore attendance data to/from your SD card.

Watch overviews in the app or export them as Excel sheets and share using Google Drive or SkyDrive.

Mail or Text message all participants of an event directly from the app.

**1.2 Functional Specifications**

Keeping track of attendance during sport training sessions. Tracking students (teachers).

Track participation during regular work meetings. Participation for Sunday schools.

Scouts.

Music (wind-bands).

Define events, assign participants to the event (from your Google account contacts) and track if persons are in, out, sick or unknown.

View Metrics provide a clear overview of the attendance: average attendance, attendance per participant, number of attendees etc.

Easily reach out to the participants by sending emails or text messages directly from the app (for example send an SMS to all participants that were missing during an instance).

Export attendance overviews to Excel. The overviews include all attendance data, notes, is late marks etc.

Backup/restore attendance data. This allows you to take the attendance data with you in case of a factory reset or purchase of a new device.

**1.3 Non-Functional Specifications**

* **Usability**: The app has to be easy to connect with nearby students and co-workers within your network attendance and questions.

SRS cmp203 Version: 1.0

* **Security:** User can only share with someone he trust to ensure he is safe frompossible harassments or abuse.
* **Efficiency**: The app needs a lot of resources to manage all the traffic thathappen Through it.

**2.2 My Attendance Tracker**

**2.2.1 Project description**

Simply sign up from any mobile or on the web and start taking your attendance from anywhere without being in front of a computer. Open your MyAT account on any smartphone or tablet and record any information you need right then, one time, and access it from anywhere and anytime you want in the future. No more writing down notes, attendance, or grade information on students or employees just to enter it into a different system later.

**2.2.2** **Functional Specifications**

No more double entering your data. Enter it once and it's available anytime and anywhere.

All site features are also available on mobile devices.

Interface was designed specifically to be used on mobile devices so it's simple and fast.

**2.2.3** **Non-Functional Specifications**

**Usability**: The app has to easy to connect with nearby students and co-workerswithin your network to take attendance and questions.

**Security:** User can only share with someone he trust to ensure he is safe frompossible harassments or abuse.

**Efficiency:** The app needs a lot of resources to manage all the traffic thathappens through it.

SRS cmp203 Version: 1.0

**2.3 Summary**

**2.3.1** **Functional Specifications**

Keeping track of attendance during sport training sessions. Tracking students (teachers).

Track participation during regular work meetings. Participation for Sunday schools.

Scouts.

Music (wind-bands).

Define events, assign participants to the event (from your Google account contacts) and track if persons are in, out, sick or unknown.

No more double entering your data. Enter it once and it's available any time and

anywhere.

**2.3.2 Essential Specifications**

* Keeping track of attendance during sport training sessions.
* Tracking students (teachers).
* Define events, assign participants to the event (from your Google account contacts) and track if persons are in, out, sick or unknown.

**2.3.3** **Non-Essential Specifications**

.No more double entering your data. Enter it once and it's available any time and anywhere

**2.3.4** **Non-Functional Specifications**

Usability

The app has to easy to connect with nearby students and co-workers within your network attendance and questions.

* Security

User can only share with someone he trust to ensure he is safe from possible harassments or abuse.

* Efficiency

The app needs a lot of resources to manage all the traffic that happens through it.

**2.3.5** **Limitations**

**The number of students submitting in app**

SRS cmp203 Version: 1.0

|  |  |  |  |
| --- | --- | --- | --- |
| **3.** | **Cost Estimate** |  |  |
|  |  |  |  |
|  | **Item** | **Estimated Cost** | |
|  | **US Dollars** | **L.E.** |
|  |  |
| 1. | **Development Cost** |  |  |
|  | 1.1. Software Engineers | 8000$ | 140000 |
| **2.** | **Deployment Cost** |  |  |
|  | 1.1 Hardware/Servers | 800$/Month | 14000/Month |
|  | 1.2 Maintenance | 500$/Month | 87500/Month |
|  | Total | 1300$/Month | 22750/Month |
|  | **Total Estimated Cost** | 8000$, | 140000, |
|  |  | 1300$/month | 22750/Month |

**Table 2: Cost Estimate**

**4. Feasibility Study**

Our software product helps in taking attendance easier and faster that you can do in traditional way and helps the lecturer in calculating some statistics, taking questions from students, mark the students who get bonus and evaluating grades so it will help him save a lot of time in the lecture. Our aim is to make it easy to the lecturer in giving the lectures saving a lot of effort and time from his point of view.

SRS cmp203 Version: 1.0

**5. Requirements Specifications**

**5.1 User Requirements (User Stories)**

**5.1.1** **US1**

* **Title:** Sign Up
* **Story**: User can sign up providing his email and password.

**5.1.2** **US2**

* **Title:** Log In
* **Story**: User can Log in if he has signed up before.

**5.1.3** **US3**

* **Title:** Forgotten password functionality
* **Story**: Change user password by email if he forgot it.

**5.1.4** **US4**

* **Title:** Edit Profile.
* **Story**: User can Edit his profile (change his name or password).

**5.1.5** **US5**

* **Title:** Make Group
* **Story**: User can make group which contain the names of all students.

**5.1.6** **US6**

* **Title:** Edit Group
* **Story**: User can edit group info by adding, removing or renaming other names.

**5.1.7** **US7**

* **Title:** Delete Group
* **Story**: The user can delete the group and all its related data.

**5.1.8** **US8**

* **Title:** Make Session
* **Story**: Make new Session related to a certain group and giving him an id tomake other people join to it.

**5.1.9** **US9**

* **Title:** End Session
* **Story**: The Session owner can end a session (no user can join to it) .

**5.1.10** **US10**

* **Title:** join a Session
* **Story**: in the lecture, the professor will give the id to the students to join to theSession using their applications.

SRS cmp203 Version: 1.0

**5.1.11** **US11**

* **Title:** Take attendance
* **Story**: the owner of the group can take attendance by telling other users thepassword, using this password every user in the session will receive the attendance list and can mark his name.

**5.1.12** **US12**

* **Title:** Take attendance manually.
* **Story**: User (the group owner) can mark someone as attendant manually(if someone forget his mobile).

**5.1.13** **US13**

* **Title:** Make Session objective list
* **Story**: make new objectives list (the goals of the session).

**5.1.14** **US14**

* **Title:** Edit Session objective list
* **Story**: User can add or remove objectives from the list.

**5.1.15** **US15**

* **Title:** Session evaluation.
* **Story**: every user in the session will receive the objectives list and can evaluatethe session(decide which objectives they have reached in the session).

**5.1.16** **US16**

* **Title:** Ask Questions.
* **Story**: every user in the session can send their question to the session owner.

**5.1.17** **US17**

* **Title:** View Question.
* **Story**: The Session owner can view the question list and mark any of them asanswered question.

**5.1.18** **US18**

* **Title:** View evaluation of session objective list.
* **Story**: The Session owner can view the Session objective list and seeachieving percentage for each objective.

**5.1.19** **US19**

* **Title:** View group statistics.
* **Story**: The user (owner of the group) can see the attendance percentage,

list of most attendant users of that group and list of most absent users.

SRS cmp203 Version: 1.0

**5.1.20** **US20**

* **Title:** Export the attendance list as excel sheet.
* **Story**: The lecturer can export a specific attendance file in excel sheet format.

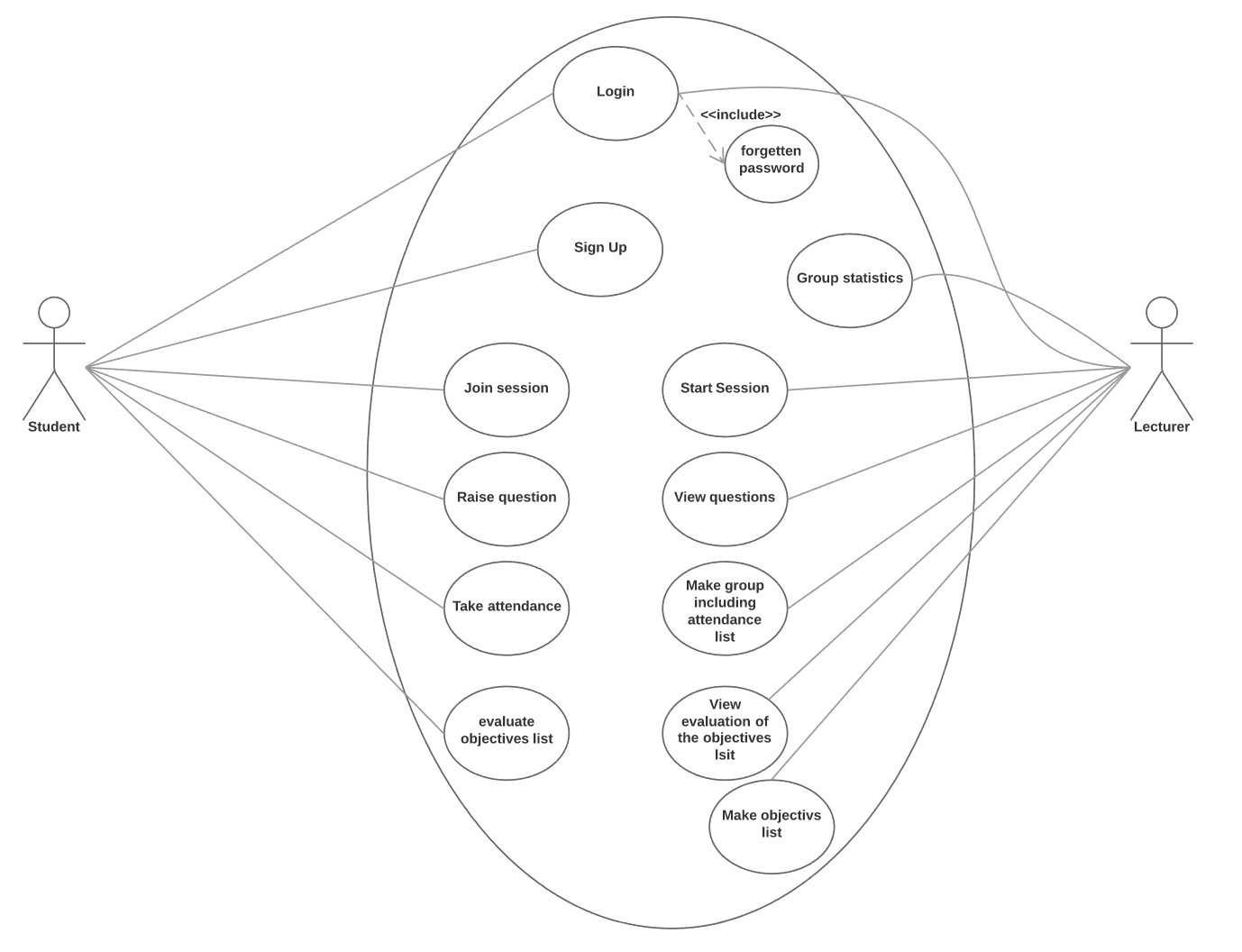
**5.1.21** **US21**

* **Title:** Take notes about students.
* **Story**: The lecturer can write notes (like tutorial bonus - mark as riotous) so hecan make use of them later.

**5.1.22** **US22**

* **Title:** Give attendance grade of each student.
* **Story**: The lecturer can let the app evaluate the attendance grade according togiven equation given by the user. (it should be function in percentage of each student) like: [percentage] \* 1 + 1

**5.2 System Requirements (Use Cases)**

5.2.0 UML Diagram

SRS cmp203 Version: 1.0

5.2.1 The user can sign up to the system

**Name: Sign up**

**Identifier** UC0

**Preconditions:**

**Basic Course**:

1. The system has two edit boxes
2. The user enter his email address and password
3. The user click button sign up
4. The system authenticate the user and sign him up

**Alternate Course A: Description of the alternate course**

**Condition:** Email or password aren’t valid, or the email already exist

A.6 The system show the user the error message with some details

**Post conditions** (List the state(s) the system can be in when this use case ends)

1. The user should be logged in and be authenticated

**Map to:**

5.2.2 The User can log in

**Name: Log In**

**Identifier** UC1

**Preconditions:**

1. A login page is displayed.
2. The database holds all users information. **Basic Course**:
3. The user enter his email address and password in the displayed fields.
4. System check whether that email is in the database.
5. if the email and password matches, system logs the user in.

**Alternate Course A: User can’t log in**

**Condition:** Email is not in the database or emailand password don’t match

A.4 The system should clear the username and password textboxes and display error message with some details

**Post conditions**

1. The user is signed in and main activity loads

**Map to:**

SRS cmp203 Version: 1.0

5.2.3 Change the password by email

**Name:** Forgotten password

**Identifier** UC2

**Preconditions:**

**1-** **The user should be signed up before.**

**2- The system should be in the log in screen**

**Basic Course**:

1- The user click on “forgot password ?”.

2- The system shall show him a new edit box to enter his email address signed up with.

3- After user input it, the system should send to his email a link to change his password.

**Alternate Course A: The email is not in the system database**

**Condition:** The user enter an email address which not signed up with it before

A.7 The system wouldn’t send any emails and should not tell the user for security reasons.

5.2.4 The User edits his profile

**Name:** Edit Profile

**Identifier** UC3

**Trigger** Student clicks on edit button from profile fragment.

**Preconditions:**

1- User Logged in

**Basic Course**:

1- The user edits the data he want to change.

2- The users clicks on the save button.

**Alternate Course A: Description of the alternate course**

**Condition:** User left any empty textbox

A.8 The system should print an error message

**Post conditions** (List the state(s) the system can be in when this use case ends)

1- profile changed in the system database and in the user textboxes

**Map to:**

SRS cmp203 Version: 1.0

5.2.5 The User can Create Group

5.2.6 The User deletes a group

5.2.7 The User asks a question

**Name: Ask question**

**Identifier** UC6

**Trigger** Student write a question and clicked send button in the question fragment

**Preconditions:**

**1-** **User Logged in**

**2-** **Should be already in an active session.**

**Basic Course**:

1- The user types the question in the question textbox

2- The users clicks on the send button

**Alternate Course A:**

**Condition:** User didn’t type a question

1- The system should print an error message

**Post conditions**

1- Question has been sent to the system database and lecturer see it at the real time

**Map to:**

5.2.8 The User starts a session

**Name: Start Session**

**Identifier** UC7

**Trigger** User has clicked on start session button in main activity

**Preconditions:**

**1-** **User Logged in**

**2-** **User created at least one group**

**Basic Course**:

1- The user chooses a group from the groups he created

2- The users adds objectives to the session

**Alternate Course A: No Alternate course**

**Post conditions**

1- The session has started

**Map to:**

SRS cmp203 Version: 1.0

5.2.9 The student take his attendance

**Name: take attendance**

**Identifier** UC8

**Trigger Time of attendance have come and the student is already connected to the session.**

**Preconditions:**

1. Logged in.
2. Should be already in an active session.
3. Attendance time of the session had come**.**

**Basic Course**:

1. The student provide the secret of this session.
2. The system show him all student names if the secret is correct.
3. The system should make the device in airplane mode.
4. The user choose his name and mark it, he can mark only one name.
5. After attendance time is ended, the app make device online again and sends the data to the database.

**Alternate Course A: The user turn off airplane mode**

**Condition:** The user force the device to turn off the airplane mode or turn on the wifi

1- The system should end his connection to the session, and he can’t reconnect until attendance time is ended.

**Post conditions**

**Map to:**

SRS cmp203 Version: 1.0

5.2.10 The User asks a question

**Name: Ask question**

**Identifier** UC9

**Trigger** Student write a question and clicked send button in the question fragment

**Preconditions:**

**1-** **User Logged in**

**2-** **Should be already in an active session.**

**Basic Course**:

1- The user types the question in the question textbox

2- The users clicks on the send button

**Alternate Course A:**

**Condition:** User didn’t type a question

1- The system should print an error message

**Post conditions**

1- question has been sent to the lecturer

**Map to:**

5.2.11 The User rates objectives

**Name: Rate objectives**

**Identifier** UC10

**Trigger Session has ended**

**Preconditions:**

**1-** **User Logged in**

**2-** **Should be already in an active session.**

**Basic Course**:

1- The user is prompted to rate session objective.

2- If the user agreed to rate objectives he will see objectives in list view.

3- He will rate each objective by a scale from one to five.

4- User clicks on send button.

**Post conditions**

1- objectives has been rated

**Map to:**

SRS cmp203 Version: 1.0

5.2.12 The lecturer can end the session and the student can disconnect at any time Name: End session

Identifier UC11

Trigger User click on disconnect button

Preconditions:

1- User Logged in.

2- Should be already in an active session.

Basic Course:

1- The user click in disconnect button

2- The system shall show him a alert dialog

3- If he confirms, the user disconnects and if he is the owner of the session the session became inactive

Post conditions:

Map to:

5.2.13 The lecturer can view the question

**Name: View Question**

**Identifier:** UC12

**Trigger:** The Group owner select view Question option to see asked questions.

**Preconditions:**

**1-** **User Logged in**

**2-** **Should be already in an active session.**

**3-** **Must be the session owner.**

**Basic Course**:

1- The Group owner press on the button to see asked question.

**Alternate Course A: No questions exist**

**Condition:** There is no questions asked

1- The system should tell the lecturer that there is no questions

**Post conditions:**

The session owner see the full list of question.

**Map to:**

SRS cmp203 Version: 1.0

5.2.14 The User joins a session

**Name: Join Session**

**Identifier** UC13

**Trigger** Student clicks on join session button in sessions fragment

**Preconditions:**

**1-** **User Logged in**

**2- User must have session id**

**Basic Course**:

1- The user types the session id

2- The users clicks on the connect button

**Alternate Course A:**

**Condition:** User didn’t type a session id or wrong id session1- The system should print an error message

**Post conditions** (List the state(s) the system can be in when this use case ends)

1. User joins the session

**Map to:**

5.2.15 The owner can see the percentage of objectives list

**Name: View Session Objectives List.**

**Identifier** UC14

**Trigger** When the Session owner press on a Button to see the result of session evaluation.

**Preconditions:**

**1-** **User Logged in**

**2-** **Must be the owner of the Session.**

**Basic Course**:

1- Session owner press on a button to see the result of evaluation of the Session.

**Alternate Course A:**

**Condition:** Session is not evaluated yet.

1- The system should print a warning message.

SRS cmp203 Version: 1.0

5.2.16 Make excel file of specific group

**Name: Make Excel file**

**Identifier** UC15

**Trigger: press on option to export the group into excel file**

**Preconditions:**

**1-User Logged in**

**2-User must be the group owner.**

**Basic Course**:

1-After making a group with list of names.

2-The user owner can export this names to Excel button by choosing this option.

**Post conditions:**

1- Excel file have been generated.

**Map to:**

5.2.17 The User takes attendance manually

**Name: Take attendance manually**

**Identifier** UC16

**Trigger** Time of attendance has come

**Preconditions:**

**1-User Logged in**

**2-Should be already in an active session.**

**3- User should be the session admin**

**Basic Course**:

1-The lecturer marks the student manually

2-The users clicks on the confirm button

**Alternate Course A: Description of the alternate course**

**Condition:** User is not admin

A.9 The system should print message to let him know that he is not admin

**Post conditions** (List the state(s) the system can be in when this use case ends)

1. The student/s have been added to the attendance sheet

**Map to:** (List the Identifiers of the user stories that this use case is addressing)

SRS cmp203 Version: 1.0

5.2.18 The Session owner gives bonus to student

**Name: Give bonus**

**Identifier** UC17

**Trigger** Session owner opens the group list

**Preconditions:**

**1-User Logged in**

**2-Should be already in an active session.**

**Basic Course**:

1-The user types the number of marks to be given to the appropriate student from group

sheet

2-The users clicks on the save button

**Post conditions** (List the state(s) the system can be in when this use case ends)

1. Bonus given to certain student/s

**Map to:** (List the Identifiers of the user stories that this use case is addressing)

5.2.19 The owner can view the group statistics

**Name: View Group Statistics.**

**Identifier** UC18

**Trigger** The group owner press a button to see statistics about the group.

**Preconditions:**

**1-** **User Logged in**

**2-** **User must be the Group owner.**

**Basic Course**:

1- The Group owner want to see statistics about the performance of the group

2- So he will click on that option

3- This will open new interface to see the result.

**Alternate Course A:**

**Condition:** No Session exists.

1- The system should print an error message “No Session Exists”.

**Post conditions**

1- The user can see the list of most active users.

2- The user can see the list of most absent users.

3- The user can see the list of most asking frequently users.

**Map to:**

SRS cmp203 Version: 1.0

5.2.20 The group owner gives grade to users in his group

**Name: Grade users**

**Identifier** UC19

**Trigger** group owner clicks on grade button in start session fragment

**Preconditions:**

**1-** **User Logged in**

**2- User should be have any groups with students.**

**Basic Course**:

1- The session owner grades the students in the group list either one by one or by typing an equation in the equation box

2- The users clicks on the save button

**Alternate Course A: Description of the alternate course**

**Condition:** User don’t have any groups

1- The system should print an error message

**Post conditions** (List the state(s) the system can be in when this use case ends)

1. The group grades are updated

**Map to:** (List the Identifiers of the user stories that this use case is addressing)

SRS cmp203 Version: 1.0

**5.3 Non-functional Requirements**

**5.3.1** **NFR 1**

* **Title:** User Friendly interface.
* **Description:** Application Interface should be easy to understand and follow bythe user.

**5.3.2** **NFR 2**

* **Title:** Fast Response Time.
* **Description:** System should be able to respond quickly during anyProcedure, so user shouldn’t notice any latency in the

Performance.

* **Addressing: US20, US21,** and **US12.**

**5.3.3** **NFR 3**

* **Title:** Correct Information.
* **Description:** System should collect correct information especially in takingAttendance (make sure that only attendant users have been Marked)
* **Addressing: US13,US12.**

**5.3.4** **NFR 4**

* + **Title:** Security.
  + **Description:** System should be secure as possible to prevent any breach orinformation being stolen.

1. **High level plan**

**Iteration 1:**

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | **User Stories** | **Estimated Time** |
| US1-US4 | | | 4days |
| US5-US7 | | | 3days |
| US8-US10 | | | 3days |
| US11-US12 | | | 6days |
|  |  | **Total Time** | 16days |
| **Iteration 2:** | | |  |
|  |  | **User Stories** | **Estimated Time** |
| US13-US15 | | | 3days |
| US16-US17 | | | 3days |
| US18 | | | 3days |
| US19 | | | 5days |
| US20-21 | | | 4days |
| US22 | | | 10days |
|  |  | **Total Time** | 28days |

1. **Supporting Information Appendix A**

SRS cmp203 Version: 1.0

**Dependability Matrix**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| U | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U |
| S0 | S0 | S0 | S0 | S0 | S0 | S0 | S0 | S0 | S1 | S1 | S1 | S1 | S1 | S1 | S1 | S1 | S1 | S1 | S2 | S2 | S2 | S2 |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| U |  | ✓ | ✓ | ✓ |  |  |  |  |  |  |  |
| S0 |  |  |  |  |  |  |  |  |  |  |  |
| 1 |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| U | ✓ |  | ✓ | ✓ |  |  |  |  |  |  |  |
| S0 |  |  |  |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| U | ✓ | ✓ |  | ✓ |  |  |  |  |  |  |  |
| S0 |  |  |  |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| U | ✓ | ✓ | ✓ |  |  |  |  |  |  |  |  |
| S0 |  |  |  |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  |  | ✓ | ✓ |  |  |  |  |  |
| S0 |  |  |  |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  | ✓ |  | ✓ |  |  |  |  |  |
| S0 |  |  |  |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  | ✓ | ✓ |  |  |  |  |  |  |
| S0 |  |  |  |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  |  |  |  |  | ✓ | ✓ |  |  |
| S0 |  |  |  |  |  |  |  |  |  |  |  |
| 8 |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  |  |  |  | ✓ |  | ✓ |  |  |
| S0 |  |  |  |  |  |  |  |  |  |  |  |
| 9 |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  |  |  |  | ✓ | ✓ |  |  |  |
| S1 |  |  |  |  |  |  |  |  |  |  |  |
| 0 |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  | ✓ | ✓ | ✓ | ✓ | ✓ |  |  |  |
| S1 |  |  |  |  |  |  |  |  |  |  |  |
| 1 |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |  |  |
| S1 |  |  |  |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  |  |  |  |  |  | ✓ | ✓ | ✓ |
| S1 |  |  |  |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  |  |  |  |  |  | ✓ | ✓ | ✓ |
| S1 |  |  |  |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| SRS | |  |  |  |  |  |  |  | cmp203 | |  |  |  |  |  |  | Version: 1.0 | | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U | U |
|  | S0 | S0 | S0 | S0 | S0 | S0 | S0 | S0 | S0 | S1 | S1 | S1 | S1 | S1 | S1 | S1 | S1 | S1 | S1 | S2 | S2 | S2 | S2 |
|  | 1 | 2 | 2 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  |  |  |  |  |  |  |  |  |  | ✓ | ✓ |  |  |  | ✓ |  |  |  |  |  |
| S1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ✓ |  |  |  |  |  |  |
| S1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | ✓ |  |  |  |  |  |  |  |
| S1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  |  |  |  |  |  |  |  |  |  | ✓ | ✓ | ✓ |  |  |  |  |  |  |  |  |
| S1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 8 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| S1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 9 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  |  | ✓ | ✓ | ✓ | ✓ |  |  |  | ✓ | ✓ |  |  |  | ✓ | ✓ |  |  |  |  |  |
| S2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 0 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  |  |  |  | ✓ | ✓ |  |  |  | ✓ | ✓ |  |  |  |  |  |  |  |  |  |  |
| S2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  |  |  |  | ✓ | ✓ |  |  |  | ✓ | ✓ |  |  |  |  |  |  |  |  |  |  |
| S2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| U |  |  |  |  |  |  | ✓ | ✓ |  |  |  | ✓ | ✓ |  |  |  |  |  |  | ✓ |  | ✓ |  |
| S2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**8-Version Control System**

Github