CSC 431

Medicollab

Software Requirements Specification (SRS)

Team #6

Arindham Aneja	Functional Requirements
Yiran Sheng	Non-Functional Requirements
Peiyu Jiang	System Constraints

Version History

Version	Date	Author(s)	Change Comments
1	Feb 26, 2021	Arindham Aneja, Yiran Sheng, Peiyu Jiang	First Draft
2	Mar 7, 2021	Arindham Aneja, Yiran Sheng, Peiyu Jiang	Second Draft
3	May 5, 2021	Arindham Aneja, Yiran Sheng, Peiyu Jiang	Final Draft

Table of Contents

1. System Requirements

1.1 Functional Requirements

1.1.1 Sign up

Title	Sign up
Description	Professor and student will create their account in the application by using sign up function.
Priority	0
Precondition(s)	1. Professor or Student has installed **"Medicollab"** in his/her Iphone or android device. 2. Professor and student must have internet connection.
Basic Flow	**1.** Professor or student will open the application. **2.** Professor and student will select the "Sign Up" option. **3.** Professor or student will enter their "Your school" **4.** Professor or student will enter theor "Email Address". **5.** Professor or student will enter their "Password". **6.** Professor or student will select their "Account type". **7.** Professor or student will press Sign up button.
Post conditions(s)	Professor or student will be directed to the Sign in page.
Use Case Diagram	3.1

1.1.2 Sign in

Title	Sign In
Description	Professor and student will provide right credentials in order to login and use the application.
Priority	0
Precondition(s)	1. User has successfully created the account.
Basic Flow	**1.** Professor or student will open the application. **2.** Login screen will be displayed to student/ professor. **3.** Professor or student will enter his/her Email address. **4.** Professor or student will enter his/her password. **5.** Professor or student will press the Sign In button. **6.** If password forgotten, click link and it will send the option to reset the password to the email filled in the login form.
Post conditions (s)	The dashboard screen will be displayed to the user.
Use Case Diagram	3.1

1.1.3 Edit profile

Title	Edit profile
Description	Professor or student will be able to update his/her profile information.
Priority	2
Precondition(s)	Professor or students has logged in successfully.
Basic Flow	1. Professor or student will select the “Edit Profile” option on the dashboard. 2. Professor or student will update the information. 3. Professor or student will select the update button
Post conditions(s)	The profile will be updated successfully.
Use Case Diagram	3.1

1.1.4 See the list of professors

Title	See the list of professors
Description	Student will see the list of all available professor that have been added in the application.
Priority	2
Precondition(s)	Student has logged in successfully.
Basic Flow	1. Student will click on ‘See full list’ button. 2. System will show the list of all available professors added in the application.
Post conditions(s)	The list will be displayed to the student.
Use Case Diagram	3.1

1.1.5 Send a request to a professor

Title	Send a request to a professor
Description	Student will send the request to the professor to add him/her in the list.
Priority	0
Precondition(s)	Student has seen the list of professors.
Basic Flow	1. Student will enter name of Professor in search bar. 2. Medical student will select “Search” option. 3. Application will show list of respective professors. 4. Student will open the profile of professor 5. Student will vaguely write his incentive to take the class. 6. Student will select the send request option.
Post conditions(s)	Request will be sent to the professor.
Use Case Diagram	3.1

1.1.6 Accept or reject the student request

Title	Accept or reject the student request
Description	The professor will see the list of all requests and click on 'Accept' or 'Reject' button to accept or reject the request.
Priority	0
Precondition(s)	Professor is logged in.
Basic Flow	**1.** Professor will select the Request icon on dashboard. **2.** Application will show the list of student's request. **3.** Professor will select a student to see the student's profile. **4.** Professor will select the "Accept request" or 'Reject request' options.
Post conditions(s)	A notification regarding request decision will be sent to the student.
Use Case Diagram	3.1

1.1.7 View all students

Title	View all students
Description	The professor will see the list of all students that are added in his/her students list.
Priority	0
Precondition(s)	Professor is logged in.
Basic Flow	**1.** Professor will select "Students" option from the dashboard screen. **2.** Professor will swipe all the way right on his screen to "View all students". **3.** Application will show the list of all students to the professor.
Post conditions(s)	The students list will be displayed to the professor.
Use Case Diagram	3.1

1.1.8 View assignments status

Title	View assignments status
Description	Students will see the list of all the assignments status which professor has assigned to him/her.
Priority	0
Precondition(s)	Student is logged into the application successfully.
Basic Flow	**1.** Student will select "My Tasks" option from the dashboard screen. **2.** Application will show the list of all assignments and their status to the student.
Post conditions(s)	The students will see the list of all assignments.
Use Case Diagram	3.1

1.1.9 View pending Assigments

Title	View pending assignments
Description	Students will see the list of all the pending assignments that the professor has assigned to him/her.
Priority	0
Precondition(s)	Student is logged into the application successfully.
Basic Flow	**3.** Student will swipe right from “My Tasks” page to view it. **4.** Application will show the list of all the pending assignments to the student.
Post conditions(s)	The students will see the list of all pending assignments.
Use Case Diagram	3.1

1.1.10 View completed assigments

Title	View completed assignments
Description	Students will see the list of all the completed assignments that the professor had assigned to him/her.
Priority	0
Precondition(s)	Student is logged into the application successfully.
Basic Flow	**5.** Student will swipe right from “Pending assingments” page to view it. **6.** Application will show the list of all the completed assignments to the student.
Post conditions(s)	The students will see the list of all completed assignments.
Use Case Diagram	3.1

1.1.11 Set class schedule

Title	Set class schedule
Description	Professor will set the schedule of a specific class and notification will be sent to all the students.
Priority	3
Precondition(s)	Professor is logged into the application successfully.
Basic Flow	**1.** Professor will select “My Schedule” option from the dashboard screen. **2.** Application will display the list of all classes to the professor. **3.** Professor will select a specific class. **4.** Professor will set the class time or schedule for the upcoming class.

	5. Application will send the notification of updated schedule to all the students of that particular class.
Post conditions(s)	The class schedule will be updated successfully.
Use Case Diagram	3.1

1.1.12 See class schedule

Title	See classes schedule
Description	Student will see the schedule of a specific class or all classes.
Priority	3
Precondition(s)	Student is logged into the application successfully.
Basic Flow	**1.** Student will select "My schedule" option from the dashboard screen. **2.** Application will display the list of all classes to the student along with the class time. **3.** Student will select a specific class. **4.** Application will show the class time and venue to the student.
Post conditions(s)	The class schedule will be displayed to the student.
Use Case Diagram	3.1

1.1.13 Logout

Title	Logout
Description	The user will click on the logout button to stop using the application.
Priority	0
Precondition(s)	The user is logged into the application successfully.
Basic Flow	**1.** The user will select the "Logout" option from the dashboard screen. **2.** The application will destroy the user session. **3.** The application will display login screen to the user.
Post conditions(s)	The user will be logged out from the application.
Use Case Diagram	3.1

1.2 Non-Functional Requirements

1.2.1 Usability

Identifier	NFR-1
Title	Usability of application
Description	The student and professor will have the experience to use Iphone and android applications and can understand how the this application works and use it with ease.
Priority	0
Applicable FR(s)	It will be applicable to all the functional requirements.

1.2.2 Reliability

Identifier	NFR-2
Title	Reliability
Description	The system will be available online to users at all times and will function effectively and will be down no longer than 30 minutes each month.
Priority	2
Applicable FR(s)	It will be applicable to all the functional requirements.

1.2.3 Capacity

Identifier	NFR-3
Title	Capacity
Description	Up to 1000 applicants can send requests to a professor in a day. Up to a 1,000,000 requests can be stored into the databse
Priority	2
Applicable FR(s)	Send a request to a professor (1.1.5)

1.2.4 Data Integrity & Security

Identifier	NFR-4
Title	Data Integrity & Security
Description	**1.** The system shall maintain data integrity by keeping backups of all updates to the database. **2.** The user credentials are stored under encryption to ensure safekeeping.
Priority	0
Applicable FR	It will be applicable to all the functional requirements.

System Constraints

2.1 Tool Constraints

2.1.1 Integrated development environment

Title	Integrated development environment
Description	We will use Android Studio IDE and Xcode for the development of this project. As of preparing this application, the latest stable version of Android Studio is 4.1.2 and Of Xcode is Xcode 12.
Priority	0

2.2 Language Constraints

2.2.1 Backend Development Language

Title	Backend Development Language
Description	Medicollab application will be developed in JAVA language. The JAVA language will be used to develop the backend functionalities of application.
Priority	2

2.2.2 Frontend Development Language

Title	Front Development Language
Description	Medicollab application’s frontend will be developed in XML.
Priority	2

2.2.3 Data Storage And Access

Title	Data Storage And Access
Description	Medicollab application’s data storage and access will be done in SQL.
Priority	2

2.3 Platform Constraints

2.3.1 Android platform

Title	Android platform
Description	Medical student's collaboration platform application will be an android based application and will run on all Android platform i.e. Mobile, Tablets, etc
Priority	0

2.3.2 IOS platform

Title	IOS platform
Description	Medical student's collaboration platform application will be an IOS based application and will run on all IOS platform i.e. Mobile, Tablets, etc
Priority	0

2.4 Hardware Constraints

2.4.1 Hardware supportability

Title	Hardware supportability
Description	Medicollab pplication will run on all android devices having android OS 6.0 Marshmallow and above and all iphone applications having IOS 14 and above.
Priority	3

2.5 Deployment Constraints

2.5.1 Deploy application

Title	Deploy application
Description	Medicollab application will be deployed on the Google play store and App Store. Student and professors will be able to easily download the application from there.
Priority	0

2.6 Budget & Schedule Constraints

2.6.1 Budget requirement

Title	Budget requirement
Description	The budget requirement for Medicollab application will be $500.
Priority	2

2.6.2 Time requirement

Title	Time requirement
Description	The Medicollab application should be completed by May 15th, 2021.
Priority	2

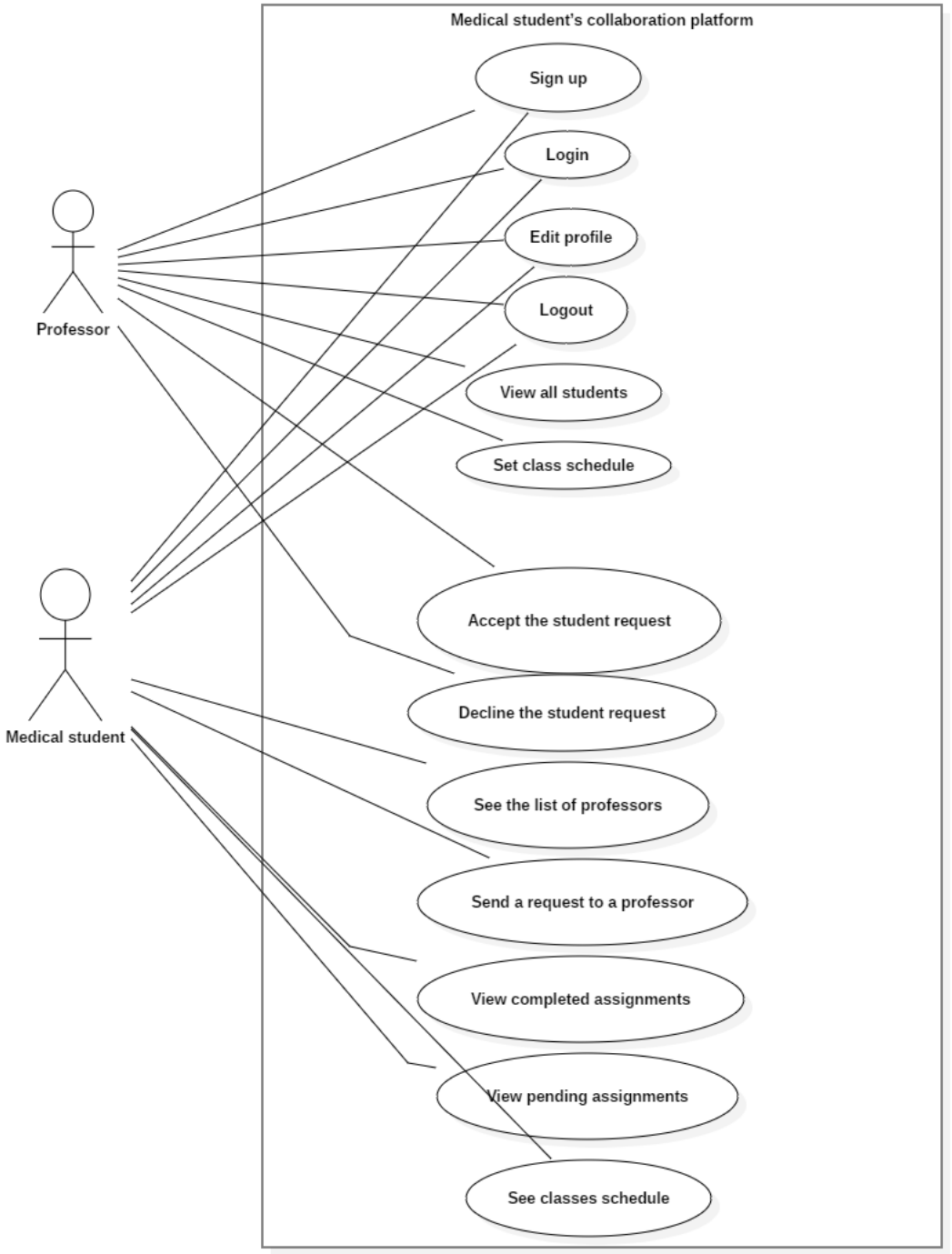
2.7 Miscellaneous Constraints

2.7.1 Application language

Title	Application language
Description	English is the universal language nowadays, therefore, the application language will be English.
Priority	0

3 Requirements Modeling

3.1 Medicollab Diagram



4 Evolutionary Requirements

4.1 Functional Requirements

4.1.1 Model View controller (MVC) architecture

Title	Model View controller (MVC) architecture
Description	The application will be developed using the Model View Controller (MVC) architecture. By using MVC architecture, we will be able to add the new features in future easily.
Priority	0
Precondition(s)	The application will be upgraded when there will need of adding new features.
Post conditions(s)	The new features will be added successfully.
Use Case Diagram	N/A

4.2 Non-Functional Requirements

4.2.1 Code standards

Title	Code standards
Description	Proper code standard will be used to develop the application. The comments will be added in the code to understand the functionality in future.
Priority	0
Applicable FR(s)	This applies to all the functional requirements.