



**CSC 431**

**TutorMe**

## **Software Requirements Specification (SRS)**

**Team 02**

Rayan N Excellent	Developer
Dido Franceschi	Developer
Pavan Gudoor	Developer
Christopher Wu	Developer



# Version History

Version	Date	Author(s)	Change Comments
1.0	2/20/22	Rayan Excellent Dido Franceschi Pavan Gudoor Christopher Wu	First draft

# Table of Contents

CSC 431

TutorMe

<b>Software Requirements Specification (SRS)</b>	<b>1</b>
<b>Version History</b>	<b>3</b>
<b>Table of Contents</b>	<b>4</b>
<b>Table of Tables</b>	<b>6</b>
<b>Table of Figures</b>	<b>7</b>
System Requirements	8
1.1 Functional Requirements	8
1.1.1 Sign-up for account	8
1.1.2 Log into Existing Account	8
1.1.3 Reset Password	8
1.1.4 Toggle Sidebar	9
1.1.5 Manage Account Settings	9
1.1.6 Display Nearby Tutors Map	9
1.1.7 Inspect Tutor Profile	10
1.1.8 Book Session with Tutor	10
1.1.9 Track Current Tutor ETA	10
1.1.10 View Upcoming Sessions	10
1.1.11 View Previous Sessions	11
1.1.12 View Session Notes	11
1.1.13 Pay For Session	11
1.2 Non-Functional Requirements	12
1.2.1 Report Issues & Concerns	12
1.2.2 Sort Available Tutor List by Criteria	12
1.2.3 Toggle between Map and List View	12
1.2.4 Keep user logged in	12
System Constraints	13
2.1 Tool Constraints	13
2.1.1 Mobile Application Framework Constraint	13
2.2 Language Constraints	13
2.2.1 React Native (Front-End), Spring Boot (Back-end)	13
2.3 Platform Constraints	13

2.3.1 Mobile Platform	13
2.4 Hardware Constraints	13
2.4.1 Mobile Phone, Tablet	13
2.5 Network Constraints	14
2.5.1 Database Accessing	14
2.5.2 Google Maps API	14
Requirements Modeling	15
3.1.1 Class Diagram	15
3.1.2 User and System	15
Evolutionary Requirements	16
4.1 Functional Requirements	16
4.1.1 Network Expansion	16
4.2 Non-Functional Requirements	16
4.2.1 Create Mentorship program	16

# Table of Tables

<b>1. System Requirements</b>	<b>8</b>
<b>Functional Requirements</b>	<b>8</b>
Sign-up for account	8
Log into Existing Account	8
Reset Password	8
Toggle Sidebar	9
Manage Account Settings	9
Display Nearby Tutors Map	9
Inspect Tutor Profile	10
Book Session with Tutor	10
Track Current Tutor ETA	10
View Upcoming Sessions	10
View Previous Sessions	11
View Session Notes	11
Pay For Session	11
<b>Non-Functional Requirements</b>	<b>12</b>
Report Issues & Concerns	12
Sort Available Tutor List by Criteria	12
Toggle between Map and List View	12
Keep user logged in	12
<b>2. System Constraints</b>	<b>13</b>
Tool Constraints	13
Mobile Application Framework Constraint	13
Language Constraints	13
React Native (Front-End), Spring Boot (Back-end)	13
Platform Constraints	13
Mobile Platform	13
<b>Hardware Constraints</b>	<b>13</b>
Mobile Phone, Tablet	13
Network Constraints	14
Database Accessing	14
Google Maps API	14
<b>3. Requirements Modeling</b>	<b>15</b>

Class Diagram	15
User and System	15
<b>4. Evolutionary Requirements</b>	16
Functional Requirements	16
Network Expansion	16
Non-Functional Requirements	16
Create Mentorship program	16

# Table of Figures

## Requirements Modeling

3.1.1 Class Diagram	10
3.1.2 User and System	10



# 1. System Requirements

## 1.1 Functional Requirements

### 1.1.1 Sign-up for account

Title	Sign up
Description	Page to create new user account
Priority	0
Precondition(s)	Application must be installed and running
Basic Flow	User clicks create account (options to create app using, facebook, gmail, apple id) User enter details
Postconditions(s)	Account is created, email is sent to account email, user is logged into their account and gain access to app functions
Use Case Diagram	<Link or number, if present>

### 1.1.2 Log into Existing Account

Title	Log into Existing Account
Description	Existing signed out users must sign in to use application
Priority	0
Precondition(s)	User must have already created an account
Basic Flow	User enters account email User enters corresponding password with email User presses sign out
Postconditions(s)	User is authenticated, screen changes to main page with map view of nearby tutors
Use Case Diagram	<Link or number, if present>

### 1.1.3 Reset Password

Title	Reset Password
Description	Forgotten password can be reset using this feature
Priority	2
Precondition(s)	User must have an account
Basic Flow	User has forgotten their login password User clicks reset password button User prompted to type in their email User is given a temporary key to reset password User must enter key and reset password in-app
Postconditions(s)	User password is changed. User is taken again to login page with a feedback notification "Password has been reset"
Use Case Diagram	<Link or number, if present>

### 1.1.4 Toggle Sidebar

Title	Toggle Sidebar
Description	Sidebar used a navigation bar to switch between application pages
Priority	0
Precondition(s)	User must be signed in. Navigation toggle button will remain available on toolbar
Basic Flow	User taps the toggle button on the upper toolbar. Button icon changes to open sidebar icon
Postconditions(s)	Navigation Sidebar opens, User can switch to Settings, Tutors, My Sessions, Book Solutions, Wallet, & Profile pages
Use Case Diagram	<Link or number, if present>

### 1.1.5 Manage Account Settings

Title	Manage Account Setting
Description	User can inspect and edit personal information, notification, preferences
Priority	4
Precondition(s)	User must log in, User has clicked account picture/settings button on sidebar
Basic Flow	User is shown the Account Settings page, Profile picture is shown with Name, School, Address User taps which detail she would like to edit (Name cannot be changed) Notifications settings button clickable
Postconditions(s)	Settings are changed and saved to new conditions
Use Case Diagram	<Link or number, if present>

### 1.1.6 Display Nearby Tutors Map

Title	Nearby Tutors Map
Description	Displays Map/List View of near by tutors
Priority	0
Precondition(s)	User must be signed in. This will serve as the home screen.
Basic Flow	User logs in Tutors Map Page set as home page upon login
Postconditions(s)	A list of pictures/profiles of tutors with their ratings and expertise shows up, User can click profiles to inspect/book tutor and proceed to payment
Use Case Diagram	<Link or number, if present>

### 1.1.7 Inspect Tutor Profile

Title	Inspect Tutor Profile
Description	Displays information about tutors that have been selected
Priority	0
Precondition(s)	Click on profile of tutor
Basic Flow	User must click profile of tutor on Nearby Tutors Map, View Upcoming Sessions, View Previous Sessions, etc.
Postconditions(s)	Displays Full Name, Educational Background, Subject Expertise, Rating
Use Case Diagram	<Link or number, if present>

### 1.1.8 Book Session with Tutor

Title	Book Session With Tutor
Description	User is directed to book session page to insert details regarding session
Priority	0
Precondition(s)	User logs in, User switched to book sessions tab via navigation bar or clicking TutorMe button on home page
Basic Flow	Users are prompted to enter subject, topic, grade, date, time, and location for the session. User can select multiple subjects, (Price will reflect) User enters the information required. User must also enter session description (whether it is an assignment, pictures, etc.) List of available tutors is propagated and constantly refreshed.
Postconditions(s)	Session slip is sent to nearby available tutors for request to join the search pool. Session is confirmed with selected tutor upon fee and profile inspection
Use Case Diagram	<Link or number, if present>

### 1.1.9 Track Current Tutor ETA

Title	Track Current Tutor ETA
Description	Transit for Tutor of imminent session displayed on screen prior to arrival confirm
Priority	3
Precondition(s)	Session must have been booked and within an hour of reserved time
Basic Flow	Transit is displayed on homepage of Tutor to User location/meetup address
Postconditions(s)	Upon Tutor arrival, User is prompted to
Use Case Diagram	<Link or number, if present>

### 1.1.10 View Upcoming Sessions

Title	View Upcoming Sessions
-------	------------------------

Description	Lists tutors from upcoming sessions
Priority	0
Precondition(s)	User is on My Sessions
Basic Flow	User selects My Sessions, and toggles to Upcoming Sessions
Postconditions(s)	Displays details of appointment with tutor, including Full Name, Subject, Cost, Location, Time, and Duration
Use Case Diagram	<Link or number, if present>

### 1.1.11 View Previous Sessions

Title	View Previous Sessions
Description	Lists tutors from previous sessions
Priority	3
Precondition(s)	User is on My Sessions
Basic Flow	User selects My Sessions, and toggles to Past Sessions
Postconditions(s)	Displays Date, Location, Full Name, Subject, Duration, Cost
Use Case Diagram	<Link or number, if present>

### 1.1.12 View Session Notes

Title	<Insert title>
Description	<A one or two sentence description>
Priority	<Priority from 0 (highest) – 5 (lowest)>
Precondition(s)	<What needs to happen before>
Basic Flow	<Brief, Sufficiently descriptive details of interaction step by step>
Postconditions(s)	<What happens as a result>
Use Case Diagram	<Link or number, if present>

### 1.1.13 Pay For Session

Title	Pay For Session
Description	Payment for upcoming tutoring session
Priority	0
Precondition(s)	Desired tutor must be selected for tutoring to happen
Basic Flow	Select tutor from Tutors Map Page
Postconditions(s)	Confirms payment details of time, location, tutor details, and cost
Use Case Diagram	<Link or number, if present>

## 1.2 Non-Functional Requirements

### 1.2.1 Report Issues & Concerns

Title	Report Issues and Concerns
Description	This functionality ensures that any errors in the app can be resolved through reporting issues or concerns via a survey of questions.
Priority	3
Applicable FR(s)	<Which functional requirement(s) is this applicable to?>

### 1.2.2 Sort Available Tutor List by Criteria

Title	Sort Available Tutor by Criteria
Description	A user needs to be able to sort through a desired criteria of Tutors, be it sorted by closest distance, highest rating, specific class, etc.
Priority	0
Applicable FR(s)	<Which functional requirement(s) is this applicable to?>

### 1.2.3 Toggle between Map and List View

Title	Toggle Between Map and List View
Description	Users can have the option to select how to view the list of Tutors, be it by spatial location or a sorted list.
Priority	2
Applicable FR(s)	<Which functional requirement(s) is this applicable to?>

### 1.2.4 Keep user logged in

Title	Keep user logged in
Description	After registering an account, the app should not prompt the user to make a new account. The app should register the user as either a student or Tutor.
Priority	0
Applicable FR(s)	<Which functional requirement(s) is this applicable to?>

## 2. System Constraints

### 2.1 Tool Constraints

#### 2.1.1 Mobile Application Framework Constraint

Title	Mobile Application Framework Constraint
Description	Our application serves quick access to tutors nearby, whereby a mobile app would be best to serve that purpose, due to requiring location and transactions
Priority	0

### 2.2 Language Constraints

#### 2.2.1 React Native (Front-End), Spring Boot (Back-end)

Title	React Native (Front-end), Spring Boot (Back-end)
Description	React Native supports many operating systems, while Spring Boot enables fast and easy development
Priority	0

### 2.3 Platform Constraints

#### 2.3.1 Mobile Platform

Title	Mobile Platform
Description	This application will be reserved solely for mobile purpose, enabling support for both iOS and Android platforms
Priority	0

### 2.4 Hardware Constraints

#### 2.4.1 Mobile Phone, Tablet

Title	Mobile Phone, Tablet
Description	Users can access this app via phone or tablet, provided the necessary information of the user and device can be used within the app (phone number, email, geolocation, etc)

Priority	0
----------	---

## 2.5 Network Constraints

### 2.5.1 Database Accessing

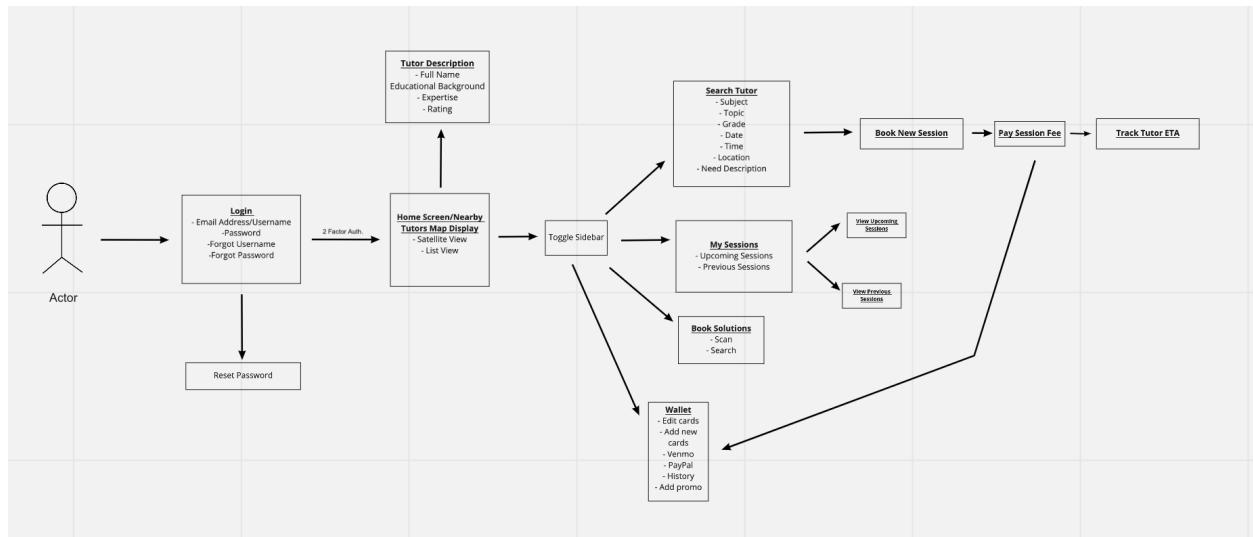
Title	Database Accessing
Description	We will retrieve information such as getting user information, updating user state, reading information from Google Maps API, and deleting operations.
Priority	0

### 2.5.2 Google Maps API

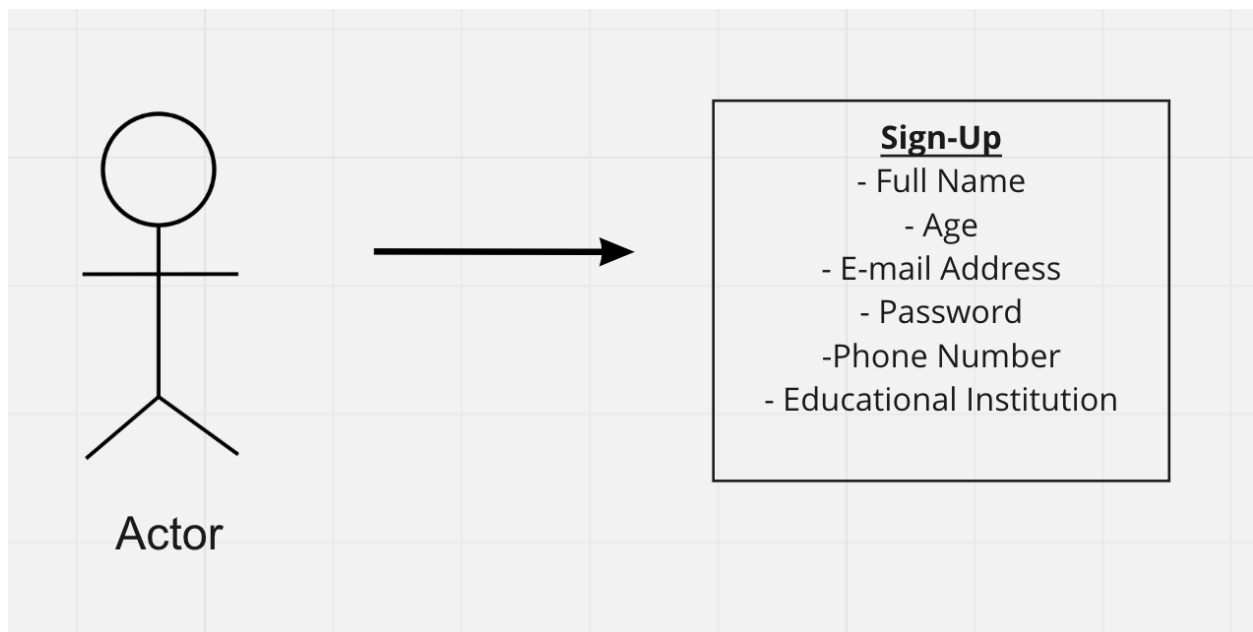
Title	Google Maps API
Description	TutorMe will use Google Maps API to enable locations throughout UM campus. Future implementations may require a wider span of area.
Priority	0

# 3. Requirements Modeling

## 3.1.1 Class Diagram



## 3.1.2 User and System





## 4. Evolutionary Requirements

### 4.1 Functional Requirements

#### 4.1.1 Network Expansion

Title	Network Expansion
Description	Users will be able to connect around the world. It is evident that users studying late at night may find it hard to connect with tutors. We will push the online narrative.
Priority	3
Precondition(s)	Users specify if they are okay with users further than 15 miles
Postconditions(s)	Information is shared with tutors around the world

### 4.2 Non-Functional Requirements

#### 4.2.1 Create Mentorship program

Title	Create Mentorship Program
Description	Instead of only tutors we expand into a program for mentors that allow members of the program to speak to mentors in their specified future fields.
Priority	1
Applicable FR(s)	Create Mentorship Program