

SEG2105 – Introduction to Software Engineering

Term Project – Tutron

Final Report

Group 5

Yahya Saleh, 300063517

Hasan Al-Yabroudi, 300256049

Yichen Liu, 300191840

Maximillien Igiraneza, 300198747

Prof. Wassim El Ahmar

Date: July 17, 2023

Introduction

The objective of this project was to design and implement a functional Android application called 'Tutron' while applying the technical skills and design principles we have learned throughout this course. The purpose of this application is to create a platform that bridges the gap between students and tutors in Ottawa, providing a seamless and integrated environment for the exchange of educational services.

Tutron is a solution for students seeking one-on-one tutoring services and tutors who offer their teaching skills and set their own availability and pricing. The application supports three types of users: Tutor, Student, and Administrator. Each type of user interacts with the application differently, which the software accommodates with a variety of features and functionalities.

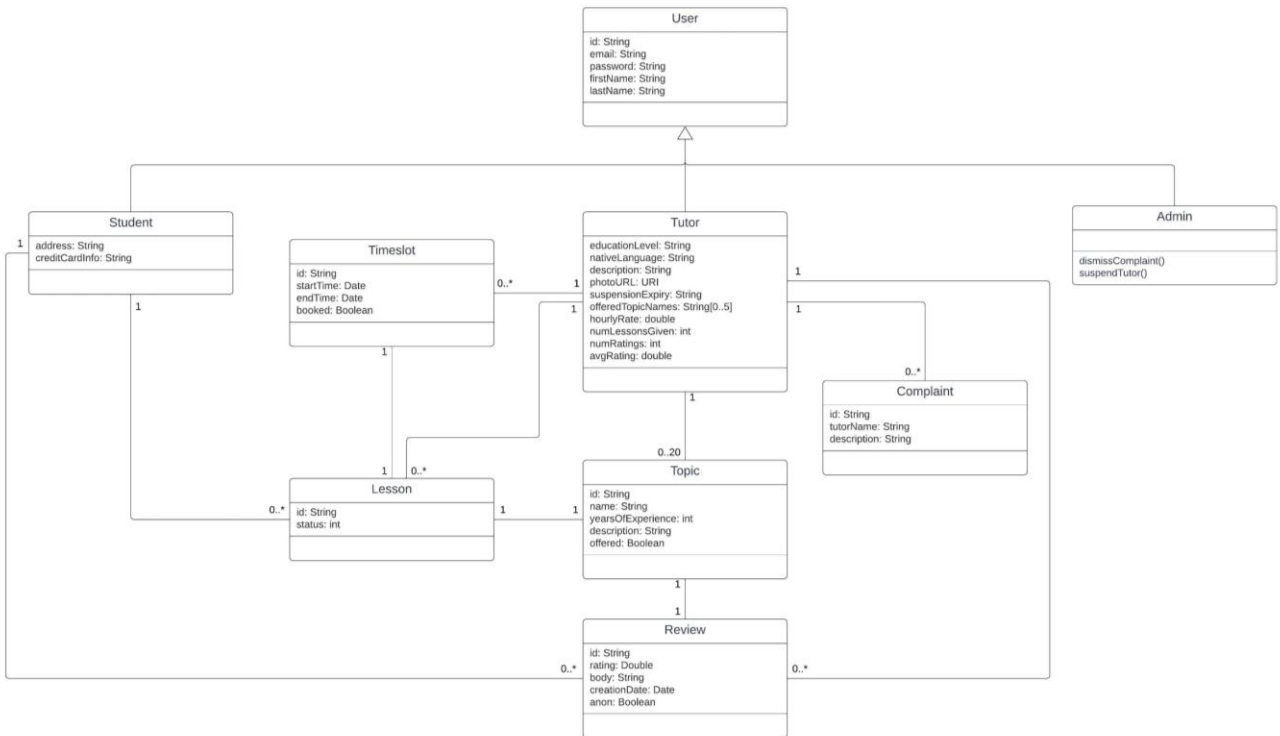
The project involved comprehensive tasks including designing, coding, testing, and debugging an application that adheres to a given set of requirements. It required us to work collaboratively, manage version control using GitHub, and ensure all features were integrated into a single application. The team focused on providing an intuitive user interface and user-specific functionalities to facilitate a robust user experience.

For Tutors, we created features that allow the user to offer their services by setting their availability and pricing. For Students, we implemented functionality to search for tutors and book sessions with the chosen tutors. In addition to these, we also integrated features to manage potential issues or disputes between students and tutors by introducing an Administrator user who can action complaints.

This report presents the final UML class diagram for the application, a summary of each team member's contributions, a series of screenshots of the application, and a discussion of the lessons we learned through our journey creating Tutron.

UML Class Diagram

This section presents the final UML class diagram for the Tutron application.



Team Member Contribution Table

For each deliverable, one team member lead the design of the required feature set, while the remaining team members adopted supporting roles. The design leader for each deliverable was responsible for drawing the UML class diagram and formulating a technical road map for how the feature set could be achieved (e.g., finding the necessary Firebase and Firestore API calls required to facilitate user authentication and document creation for deliverable 1). The supporting team members were responsible for suggesting improvements to and or editing the UML diagram, performing code-reviews, suggesting refactors, and performing usability testing at each stage of the application's development.

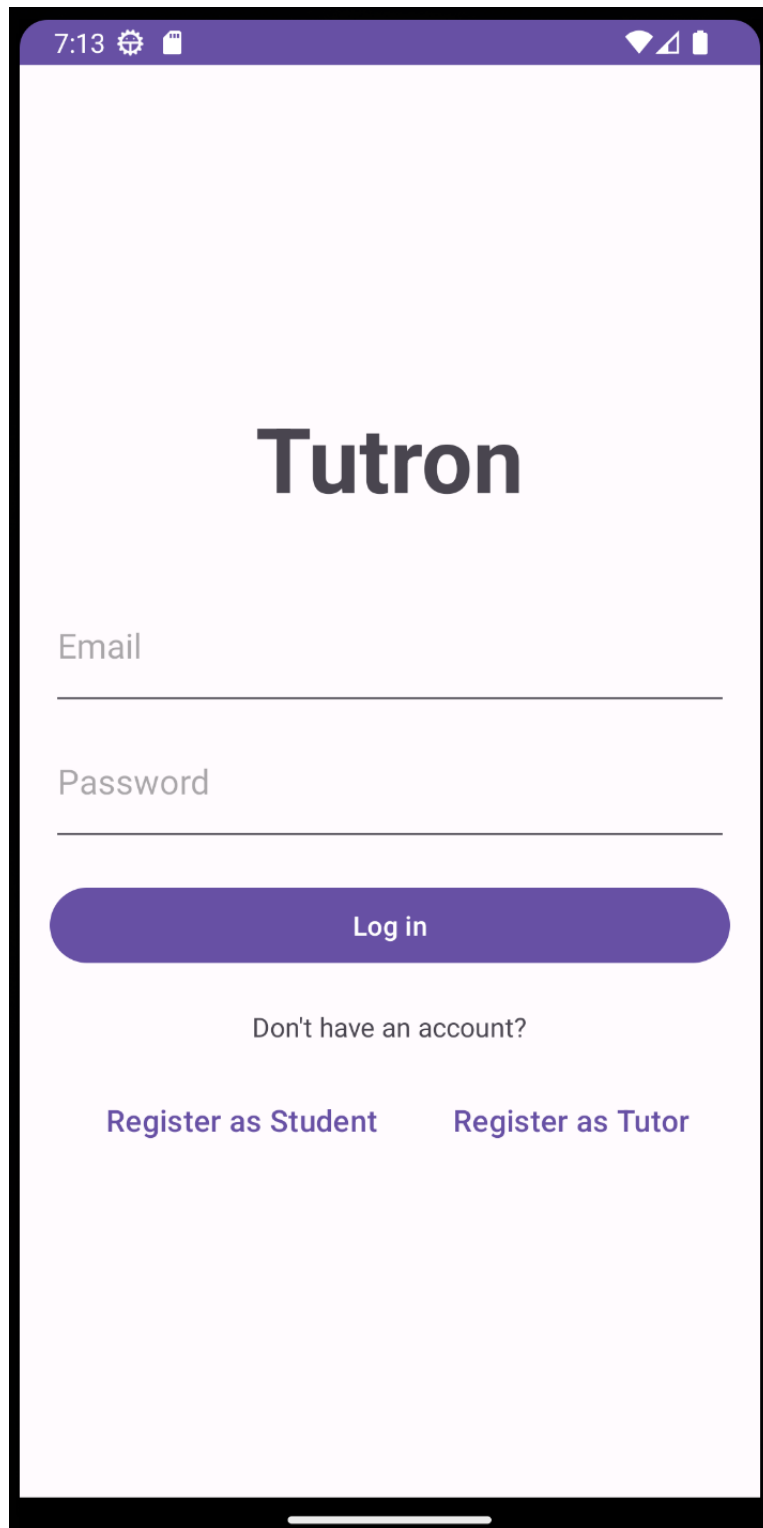
The following table summarizes the contributions of each team member for each of the deliverables. "Support" is used to denote a supporting role as described above.

Member Name	Deliverable 1	Deliverable 2	Deliverable 3	Deliverable 4
Yahya Saleh	Support	Support	Support	Lead design of tutor search, lesson request and management, and review/ratings systems
Hasan Al-Yabroudi	Support	Support	Lead design of tutor topic management system	Support
Yichen Liu	Support	Lead design of complaint and tutor suspension system	Support	Support
Maximillien Igiraneza	Lead design of authentication and user registration system	Support	Support	Support

Screenshots of Application

The following section contains screenshots of each component of the application. Dummy data has been used to illustrate how the app may look to real users.

Authentication

A mobile application interface for authentication. The screen has a light purple background. At the top, there is a status bar with the time 7:13, a gear icon, a battery icon, and a signal strength icon. The main heading "Tutron" is centered in a large, bold, dark grey font. Below the heading are two input fields: "Email" and "Password", each with a horizontal line underneath. A purple rounded rectangular button with the text "Log in" is centered below the input fields. Below the button is the text "Don't have an account?". At the bottom, there are two links: "Register as Student" and "Register as Tutor", both in a purple font. The entire screen is framed by a thick black border.

7:13

Tutron

Email

Password

Log in

Don't have an account?

[Register as Student](#) [Register as Tutor](#)

Student Registration

7:14

Student Registration

Fill in the following information. Scroll down to submit!

Personal Info

First Name

Last Name

Email

Password

Payment Info

Address

City

Province

Postal Code

Credit Card Number

Expiry

CVV

Submit

Tutor Registration

7:15

Tutor Registration

Fill in the following information. Scroll down to submit!

Personal Info

First Name

Last Name

Email

Password

Tutoring Details

Native language:

☐ English

☐ French

Education level:

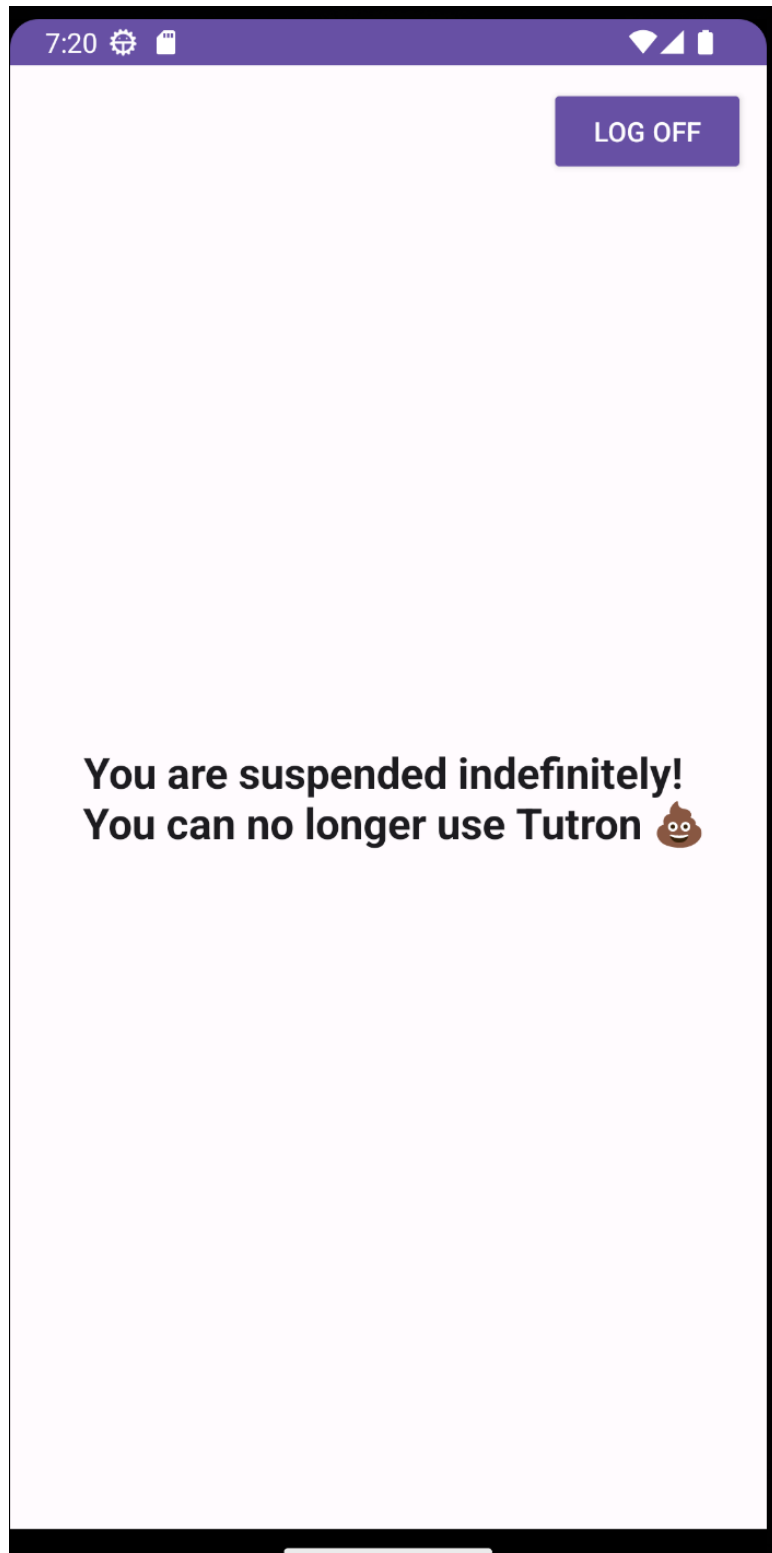
☐ High School Diploma

☐ Bachelor's Degree

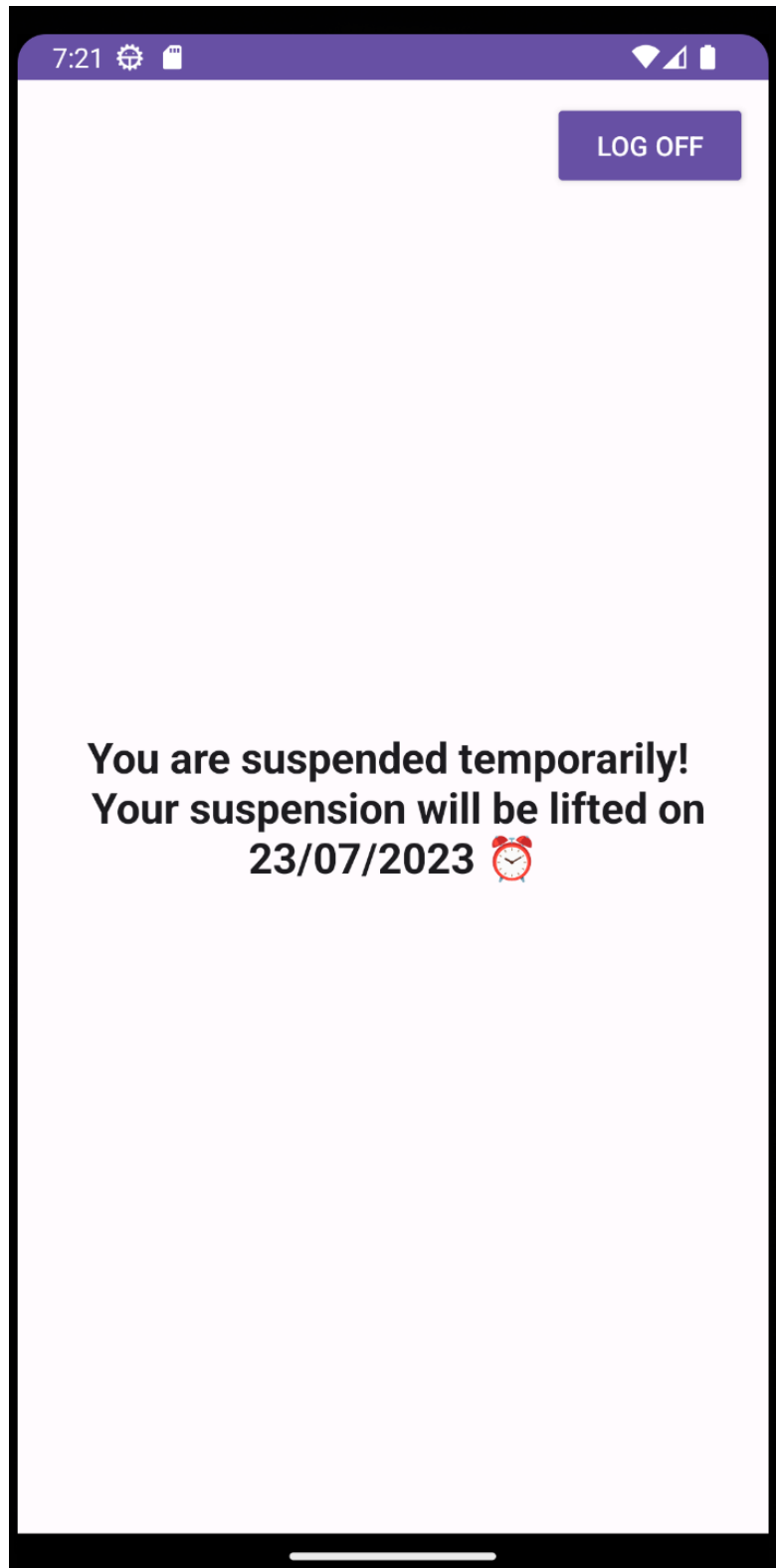
☐ Master's Degree

☐ PhD

Tutor Home (Indefinitely Suspended)




Tutor Home (Temporarily Suspended)



Tutor Home (Non-Suspended)

7:16

LOG OFF



Isaac Newton

★ 0.0 · 2 lessons given

🎓 High School Diploma 🗣️ English

\$ 20.0 per hour

I'm a mathematician and physicist. I developed calculus and invented classical mechanics.

MANAGE LESSONS

MANAGE TOPICS

20.0

UPDATE HOURLY RATE

Update Your Availability

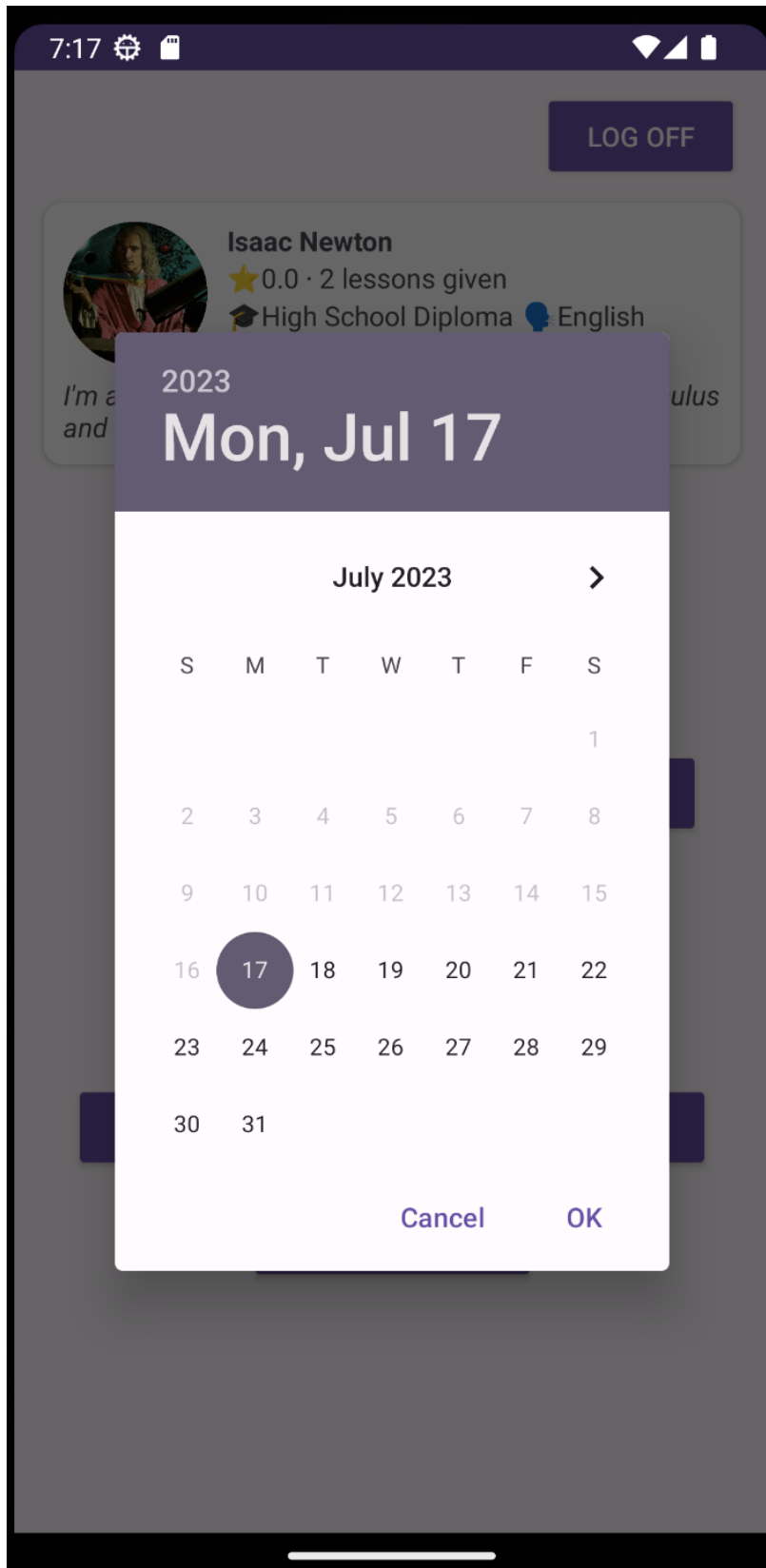
SET DATE

SET START TIME

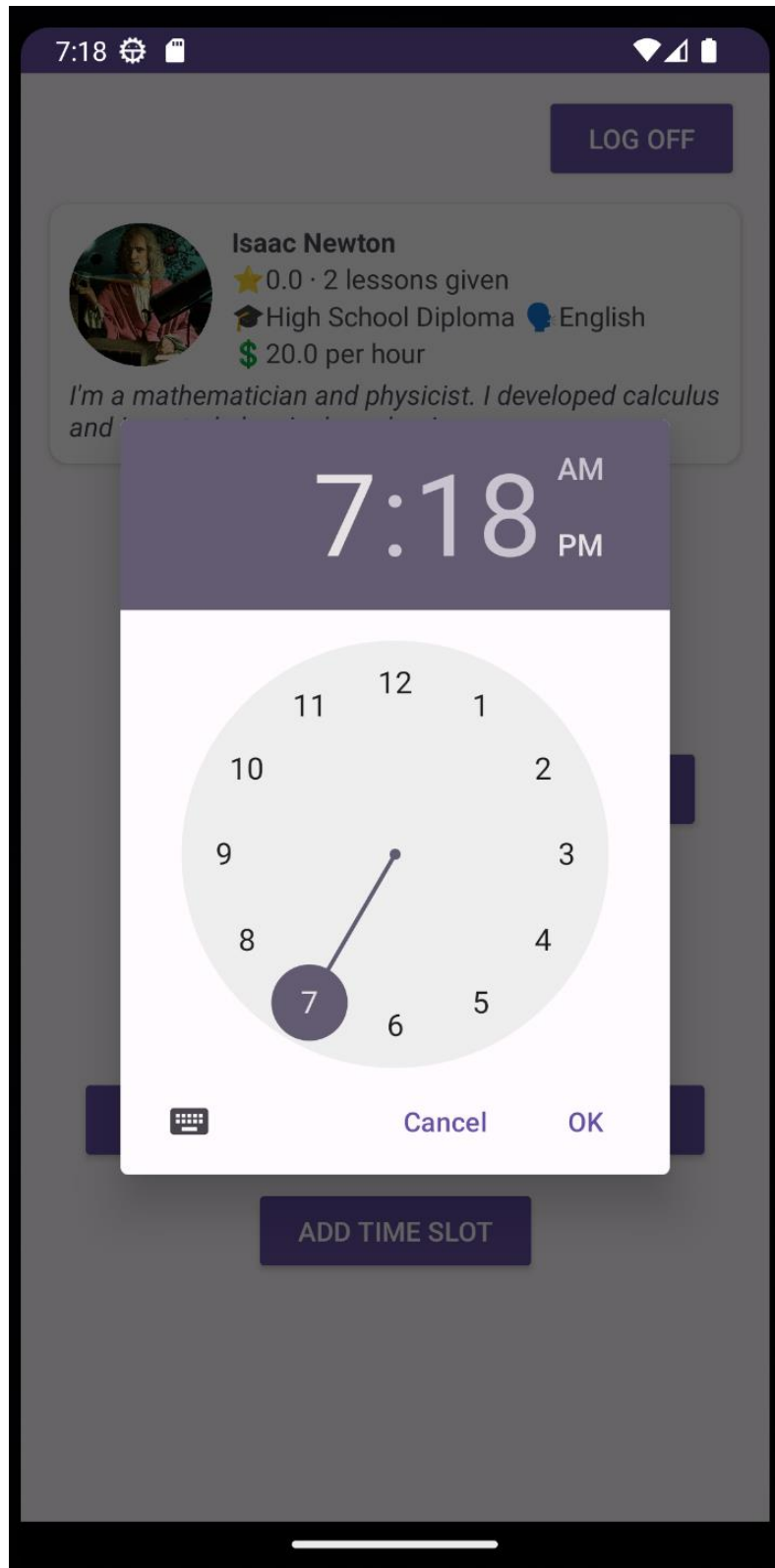
SET END TIME

ADD TIME SLOT

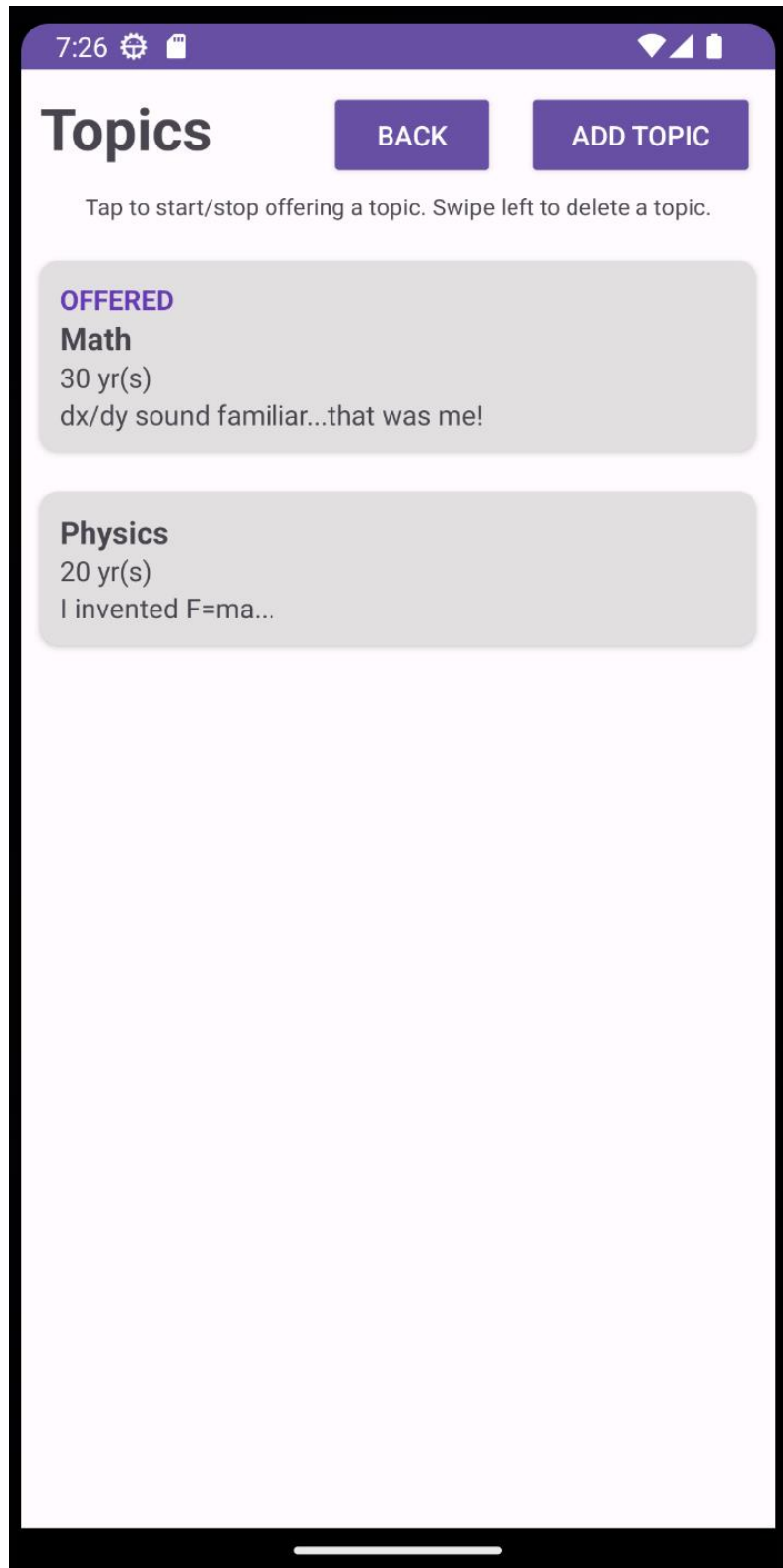
Tutor Home (Date Picker)



Tutor Home (Time Picker)



Tutor Topic Manager



Tutor Topic Adder

The image shows a mobile application interface for adding a topic to a profile. At the top, there is a purple status bar with the time 7:27, a settings gear icon, a memory card icon, and standard Android connectivity icons (Wi-Fi, signal strength, battery). Below the status bar is a white header area containing a purple 'BACK' button in the top right corner. The main content area has a light pink background and features the heading 'Add a topic to your profile!' in bold black text. Below the heading are three input fields: 'Topic name', 'Years of experience', and 'Describe your experience', each with a light gray placeholder text and a horizontal underline. At the bottom of the form is a purple 'ADD TOPIC' button. The entire app interface is framed by a black border, and a white horizontal line at the very bottom represents the mobile home indicator bar.

7:27

BACK

Add a topic to your profile!

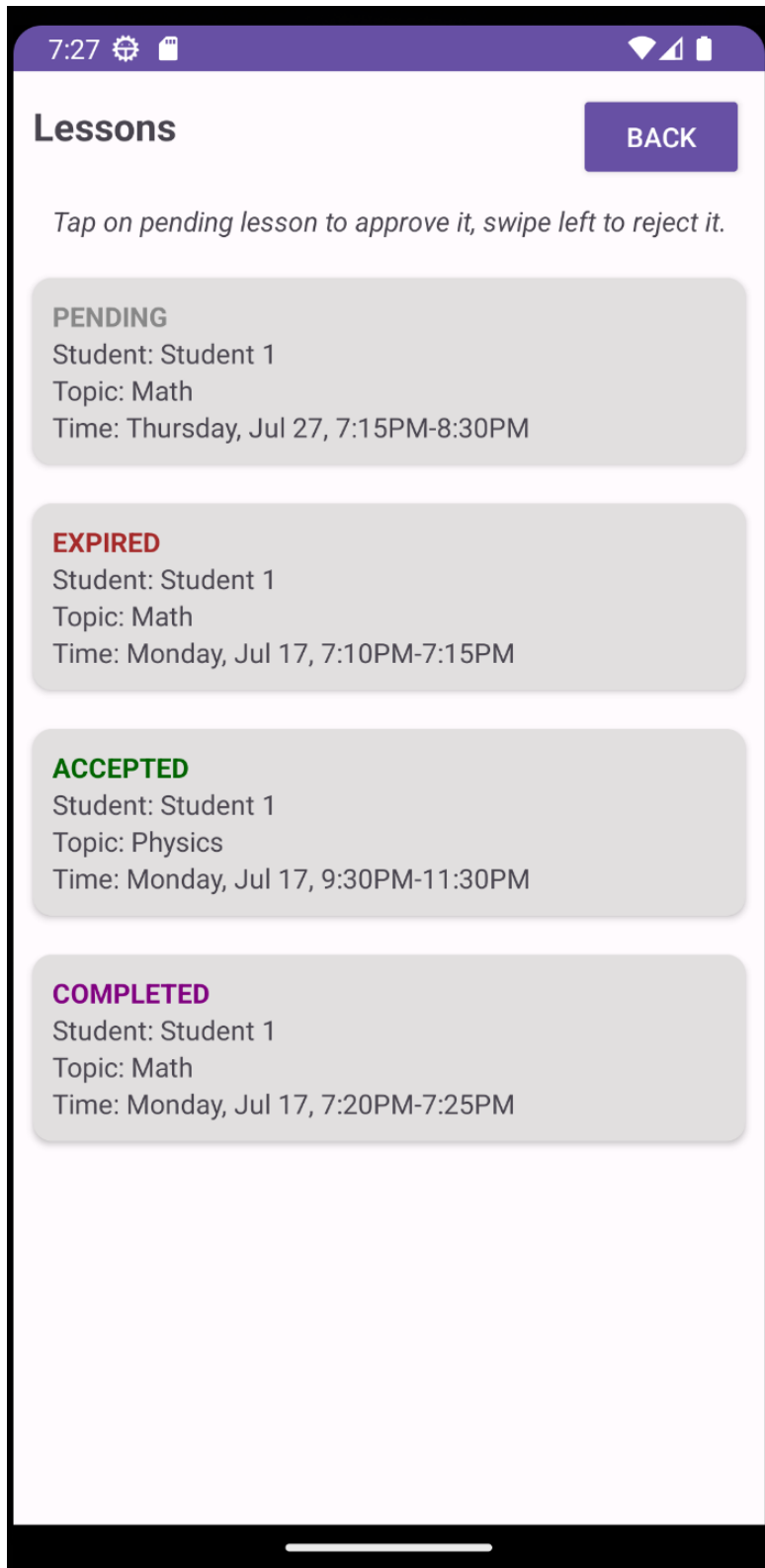
Topic name

Years of experience

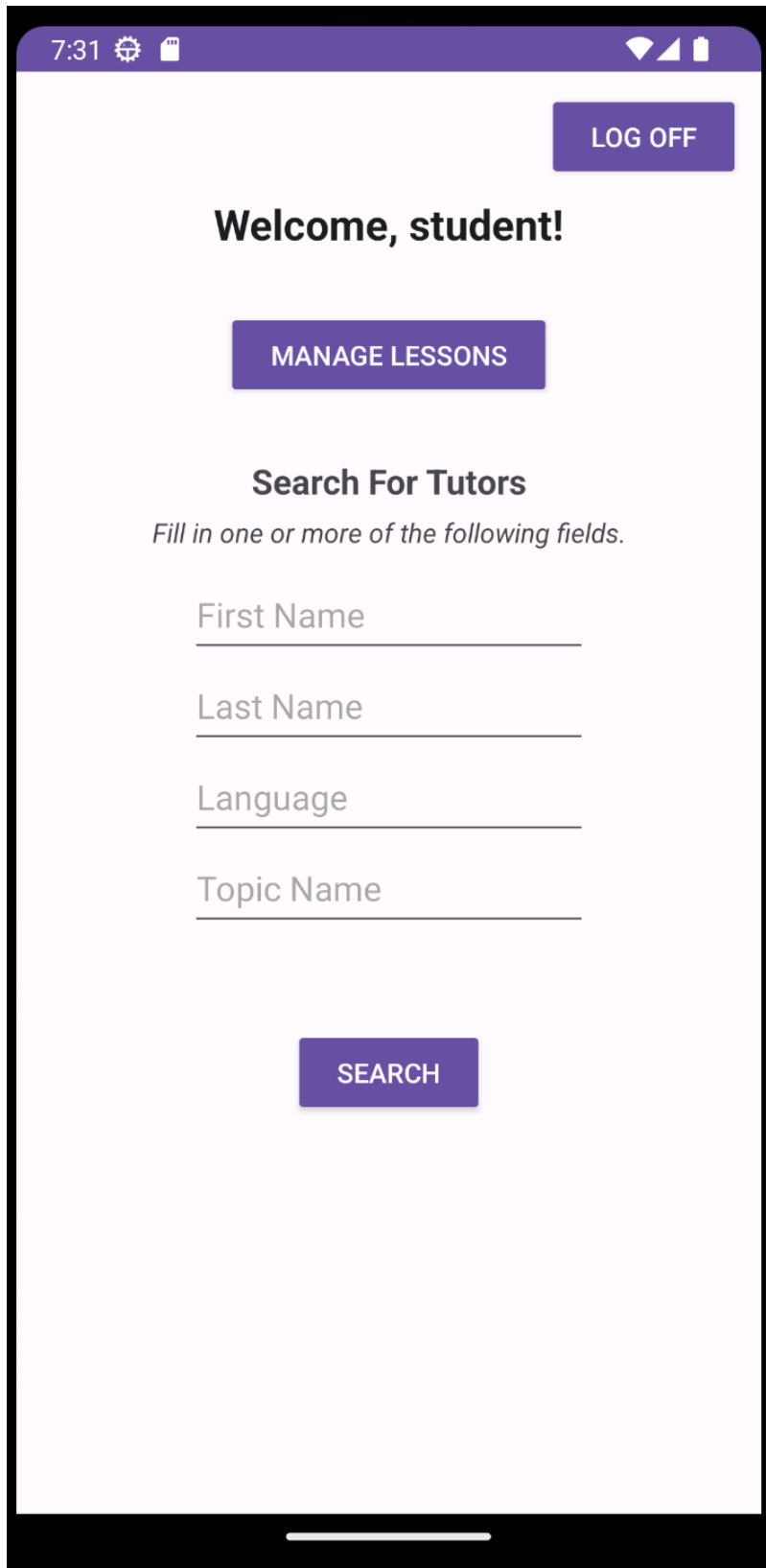
Describe your experience

ADD TOPIC

Tutor Lesson & Purchase Request Manager



Student Home

A mobile application interface for a student home page. The interface is displayed on a black background representing a phone. At the top is a purple status bar with the time 7:31, a gear icon, a document icon, a Wi-Fi signal icon, and a battery level icon. The main content area has a light purple background. In the top right corner, there is a purple button with the text "LOG OFF". Below this, the text "Welcome, student!" is centered. Underneath the welcome message is a purple button with the text "MANAGE LESSONS". Further down, the text "Search For Tutors" is centered, followed by the instruction "Fill in one or more of the following fields." in a smaller, italicized font. Below the instruction are four text input fields, each with a label above it: "First Name", "Last Name", "Language", and "Topic Name". At the bottom of the search section is a purple button with the text "SEARCH".

7:31

LOG OFF

Welcome, student!

MANAGE LESSONS

Search For Tutors

Fill in one or more of the following fields.

First Name

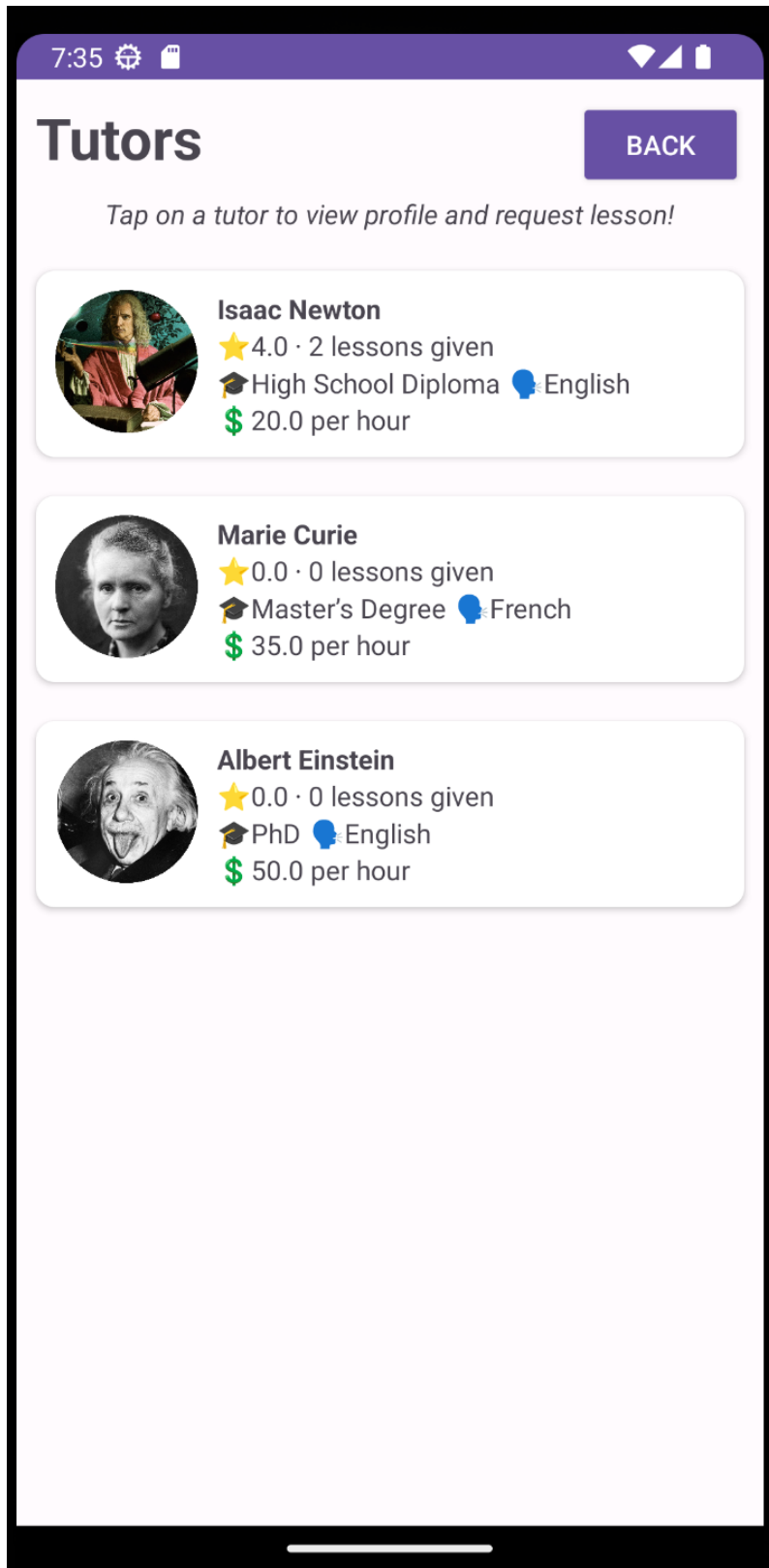
Last Name

Language

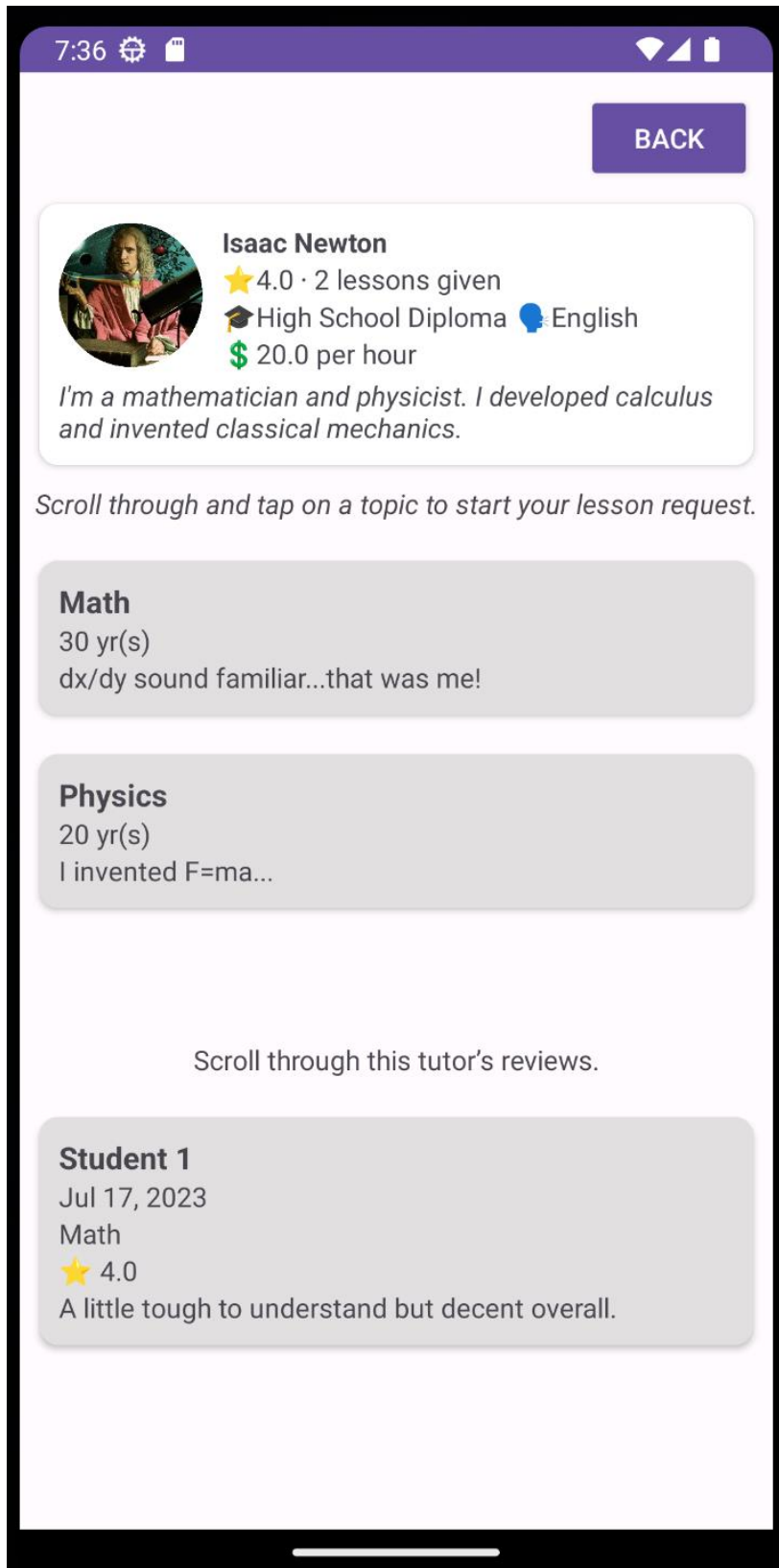
Topic Name

SEARCH

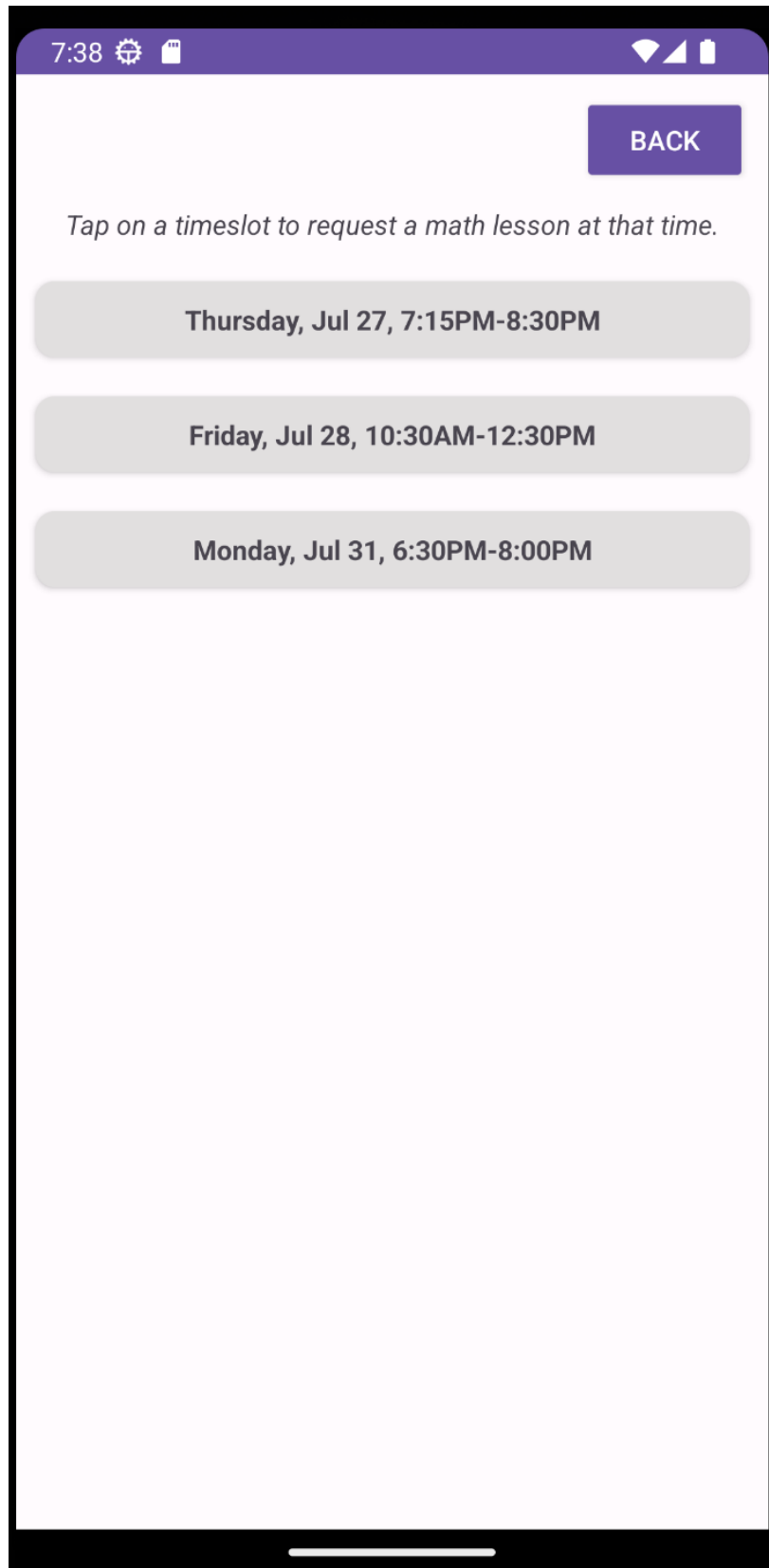
Tutor Search Results



Tutor Profile (*as seen by student*)



Lesson Request



A mobile app interface for requesting a math lesson. The screen has a purple header bar with the time 7:38, a gear icon, a document icon, and status icons for Wi-Fi, signal, and battery. A purple 'BACK' button is in the top right. Below the header, a light purple background contains the instruction 'Tap on a timeslot to request a math lesson at that time.' in italics. Three light gray rounded rectangular buttons are listed vertically, each containing a date and time range. The first button is 'Thursday, Jul 27, 7:15PM-8:30PM', the second is 'Friday, Jul 28, 10:30AM-12:30PM', and the third is 'Monday, Jul 31, 6:30PM-8:00PM'. A white horizontal line is visible at the bottom of the screen.

7:38 ⚙️ 📄 📶 📶 🔋

BACK

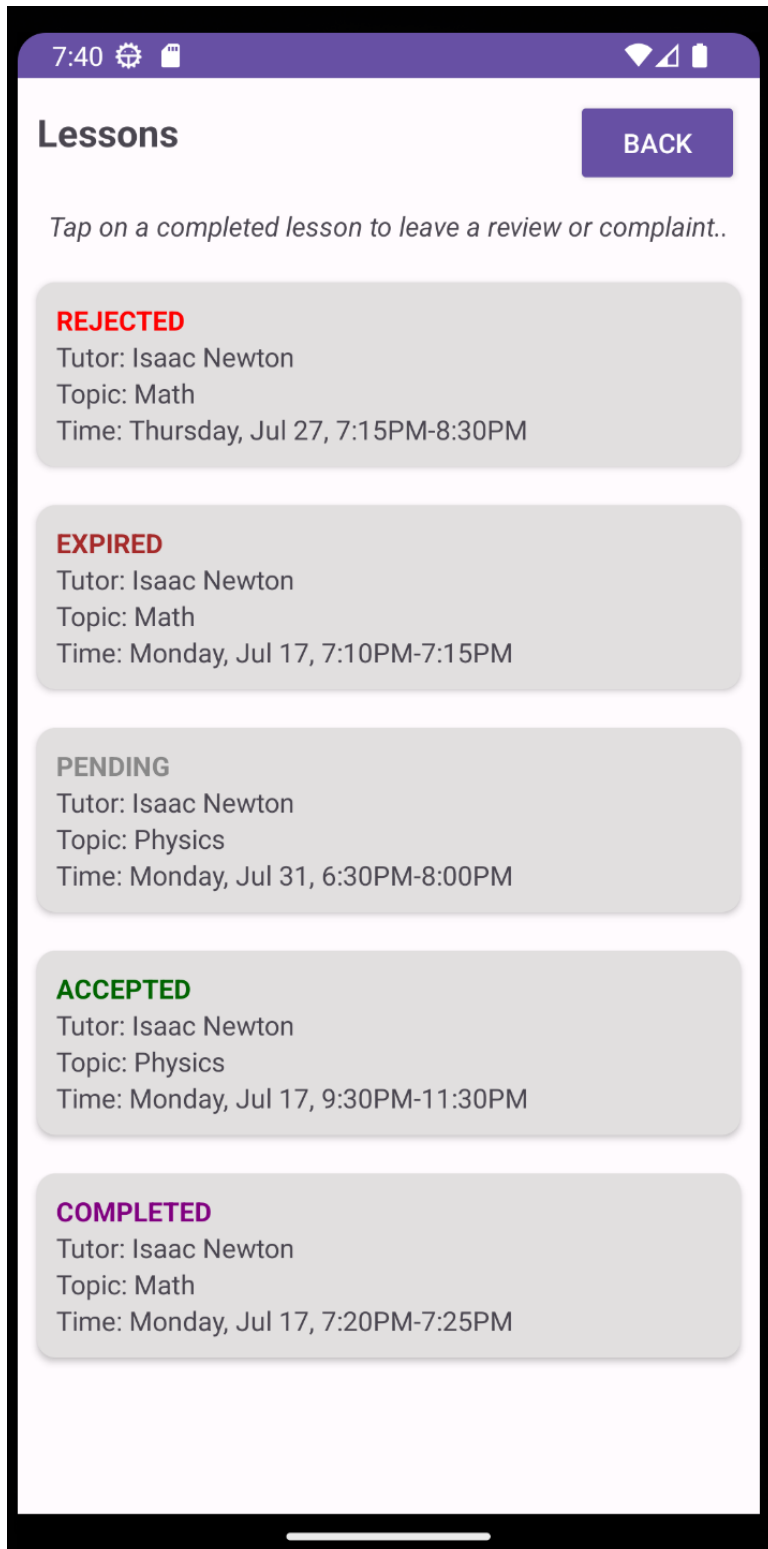
Tap on a timeslot to request a math lesson at that time.

Thursday, Jul 27, 7:15PM-8:30PM

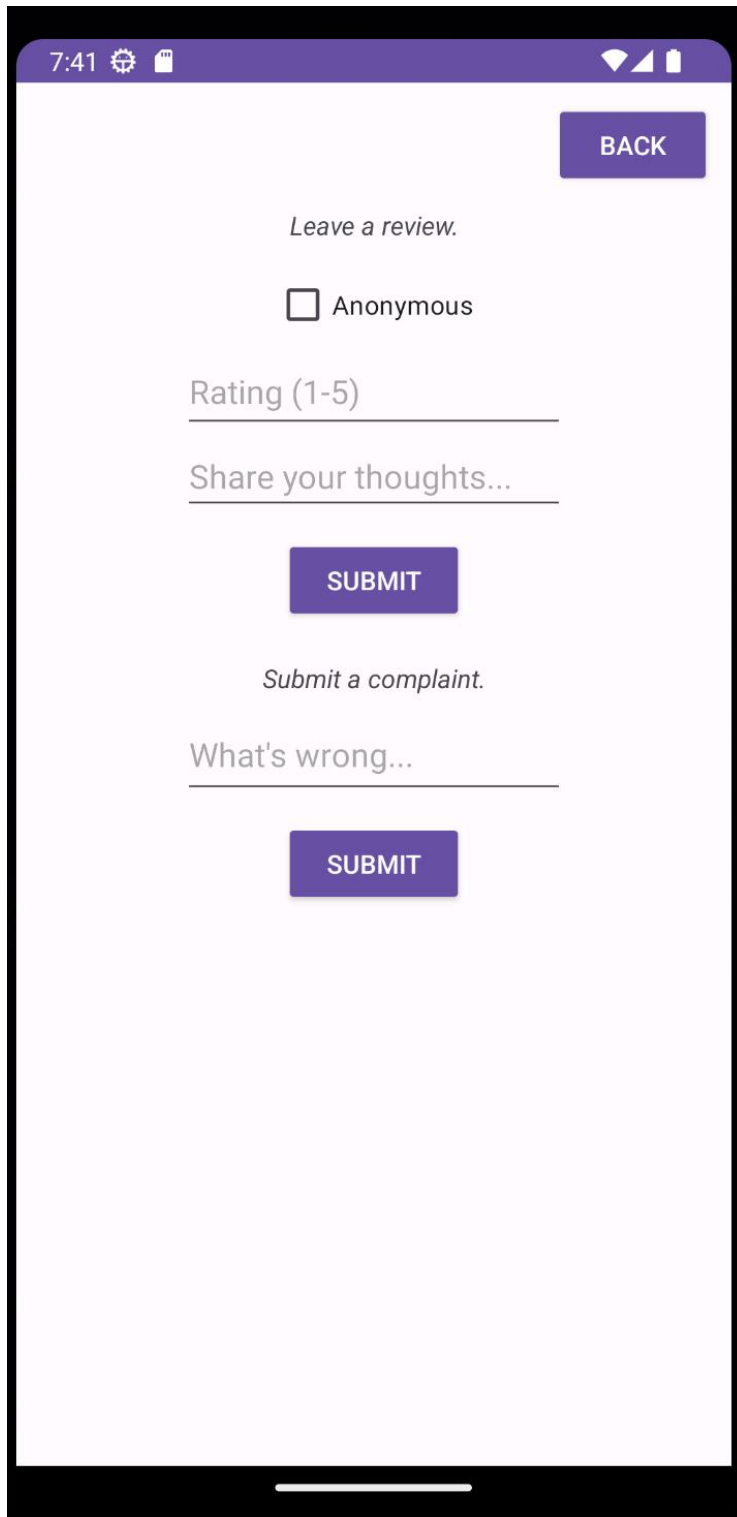
Friday, Jul 28, 10:30AM-12:30PM

Monday, Jul 31, 6:30PM-8:00PM

Student Lesson Manager



Review and Complaint Submission



The image shows a mobile application interface for submitting reviews and complaints. The app has a purple header bar with the time 7:41, a settings gear icon, a document icon, and status icons for Wi-Fi, signal, and battery. A 'BACK' button is in the top right corner. The main content area is light purple and contains two sections. The first section, 'Leave a review.', includes an 'Anonymous' checkbox, a 'Rating (1-5)' input field, and a 'Share your thoughts...' text area. A 'SUBMIT' button follows. The second section, 'Submit a complaint.', includes a 'What's wrong...' text area and another 'SUBMIT' button. The app is framed by a black border with a white home indicator bar at the bottom.

7:41 ⚙️ 📄 📶 🔋

BACK

Leave a review.

☐ Anonymous

Rating (1-5) _____

Share your thoughts... _____

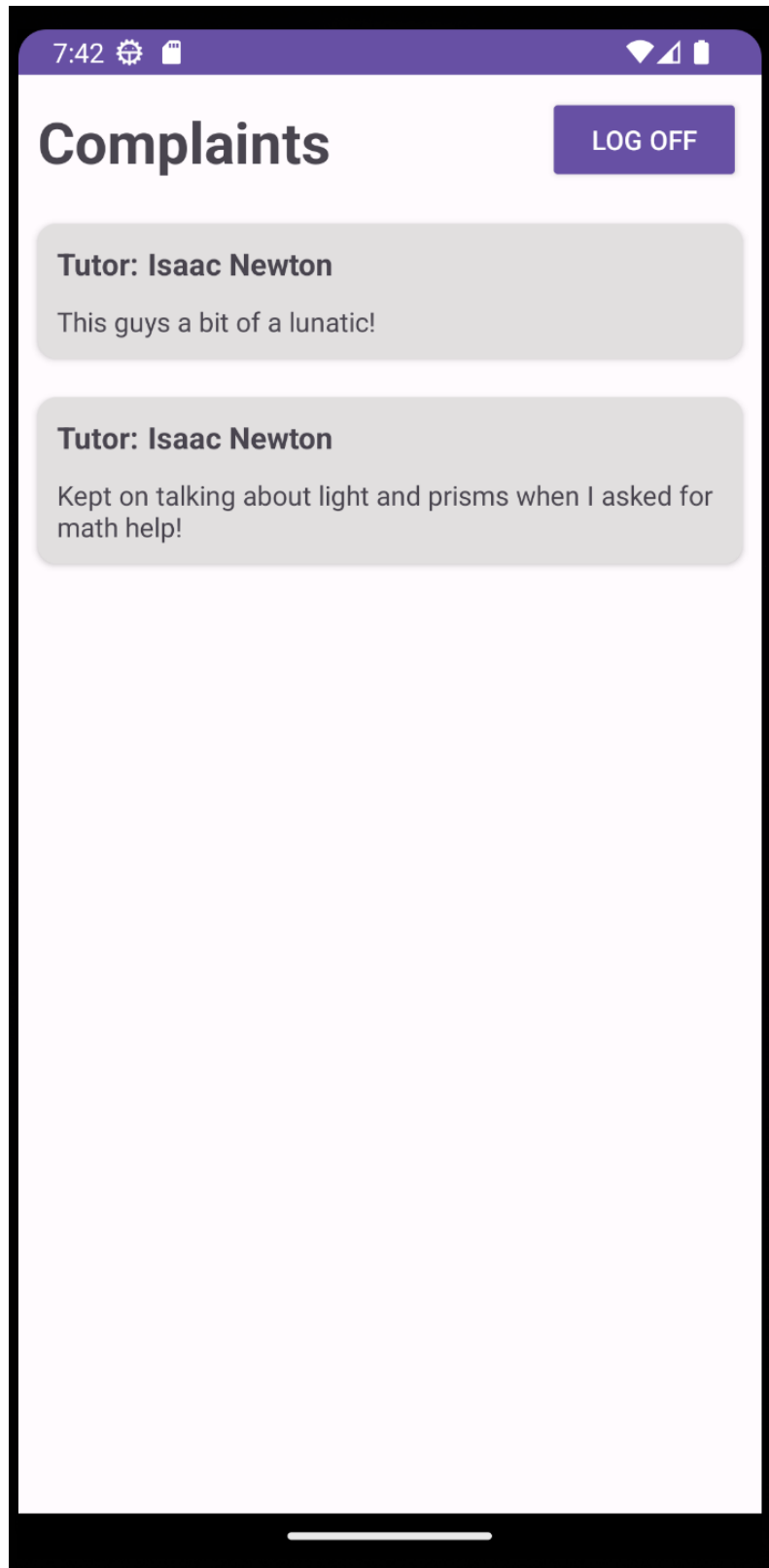
SUBMIT

Submit a complaint.



What's wrong... _____




SUBMIT

Administrator Complaint List



Actioning Complaint

7:43  

BACK

Tutor: Isaac Newton

Kept on talking about light and prisms when I asked for math help!

DISMISS

SUSPEND TUTOR

Lessons Learned

This project was an intensive hands-on experience that has given us deep insight into the many facets of software engineering. We learned about the complexity of building a complete application and the importance of careful planning, robust design, and meticulous testing.

One of our primary learning experiences was realizing how critical reusability and modularity is in software design. In the early stages of the project, we noticed redundancy in our codebase due to a naïve approach towards database handling and user interface (UI) component creation. As we progressed, we discovered and developed more efficient methods to handle these tasks, which significantly reduced code duplication and expedited the development process.

For instance, we implemented a database handler class that automated document retrieval, set functions, and compound queries. Additionally, we designed a generic RecyclerView adapter used across the application, which enabled us to avoid creating multiple similar adapters for different views. These reusable modules not only streamlined our development but also improved code readability and maintainability, underlining the importance of principles such as DRY (Don't Repeat Yourself) in software development.

Our experience with design patterns, learned formally in class, became very tangible during this project. The 'Lesson' class, an association class, facilitated lesson purchase requests and provided a route for students to submit reviews and complaints. We also utilized a Singleton pattern in our 'DataManager' utility class, which tracked the current tutor or student instance throughout the application's lifecycle, eliminating the need for intent passing between activities. These patterns helped us design a robust architecture for our application and eased the implementation of complex functionalities.

Beyond the technical skills, this project underscored the significance of collaboration and team dynamics in software engineering. Working as a team on a large-scale project introduced us to challenges related to integrating diverse ideas and pieces of code effectively. We learned the value of structuring the development process and establishing coding standards to ensure smooth collaboration and conflict avoidance.

In sum, the project served as an intensive learning journey, honing our software engineering skills and teaching us valuable lessons in collaborative project development, design principles, and the importance of code reusability. If given more time, we would have liked to leverage the tools and approaches we developed and undertake a significant refactor of our

codebase. This would further reduce the code size and enhance its maintainability, leaving us better prepared for any future development challenges.