Computer Science Department
CSCI 409 Senior Project II – Final Project Report Spring 2024

Title of the project:	"Online Tutoring Marketplace"
Team Members:	Aruay Amangeldi, Bibigul Konkayeva, Dilnas Yerkemblan, Zhassulan Abdrakhmanov
Project Advisor/Co-Advisors	Askar Boranbayev

Executive Summary (10%)

The goal of this project was to develop an online marketplace to connect tutors and students. This addressed issues of inconvenience and high time consumption of the traditional methods. Our delivered solution is a web and mobile application, implemented on React, Java, and Flutter for an effective student-tutor match. The platform functionality also enables online class scheduling and user-wise communication. The developed solution fulfills the objectives of creating an affordable and convenient tutoring platform.

Introduction (10%)

Problem description and motivation:

The motivation behind our project is to address the inefficiency of the current methods of finding qualified tutors in Kazakhstan. Oftentimes, it is time-consuming and sometimes even costly. There is also a limited number of tutor websites currently operating in Kazakhstan, and, even then, most of them have irrelevant data. To handle these issues, this project intends to build an online tutoring marketplace that provides a seamless, efficient, and engaging solution for matching students with tutors based on the personalized criteria. The platform services are also offered to educational institutions in order to help them improve communication with their students and promote themselves to a larger audience.

Project description:

The focus of our online tutoring marketplace is on usability. The platform serves both tutors and students. Those in need of tutoring services can create a student profile, specify their subject, location, and pricing preferences, and then receive personalized tutor recommendations. When a tutor is chosen, the platform offers a messaging chatbox for communication, video classes, and the ability to share educational materials. Tutors can create their profiles, list their schedules and choose students. In order to increase engagement among student, tutors can also recommend certain study materials and post them on their weed, or host video conferencing webinars and classes.

Report organization:

This report is organized into the following sections to provide a cohesive and systematic project presentation:

1. *Background and related work:* includes approaches others have used to solve a similar problem
2. *Project approach:* includes a detailed description of our solution, high-level software architecture diagrams, workflow diagrams, roles of each team member, tools used, and use case diagrams.

3. *Project execution:* includes design decisions, changes to the project, and encountered problems over the course of the last two semesters.
4. *Evaluation:* an analysis of our project evaluation through user feedback using a questionnaire.
5. *Conclusion and possible future work*
6. *References*
7. *Appendix:* includes the questions and summary of answers from the questionnaire.

Background and Related Work (15%)

For research purposes, we analyzed similar popular online marketplaces.
LRNKEY [1]. Learnkey is an online tutoring platform. The platform has tutors in more than 25 fields. Methodology is simple as ours: student chooses tutor from given list, books a lesson. Also, similar to our project, it provides a way to sign up for both students and tutors. First lesson for a teacher is free for students. Lessons are only one on one through online video calls provided by the platform, in contrast to us. Students can assess and leave comments about tutors. Tutors also should provide schedules and preview videos.

Preply [2]. Preply is an online language tutoring platform. The platform has nearly 24 language sections. The methodology is as previous one: student chooses tutor from the list, and books a lesson. Similar to LRNKEY, it provides a registration for both students and teachers. In contrast to LRNKEY and our project, the service is more global in a way that it provides a variety of languages that the website works with. Tutors there also provide schedules and preview videos. Tutors are divided into categories such as simple tutors, “super tutors”, and “professional tutors”. Similarly to LRNKEY, students can assess and leave comments about tutors.

Skooli [3]. Skooli is a tutoring platform. It emphasizes its focus on student safety by including rigorous tutor background checks and data privacy for students, which is also a key piece in our project. It also provides 24/7 online help for students, which is quite difficult to implement. It requires its teachers to have a license. In contrast to previous platforms, it provides its services for organizations and companies, schools and districts, and individual students. It uses an hourly payment system, which is not the system that we will use.

Project Approach (20%)

Last semester during the initial planning stages, we created a system flow diagram and UML diagram to visualize the key components and their interaction on our platform. The system flow diagram structurizes the main processes and data flows between components on the system. It includes high-level flow charts starting from opening the web application and all the way up to conducting the lessons on the platform. User functionality can be shown to differ depending on the user type (Student, Tutor, or Admin).

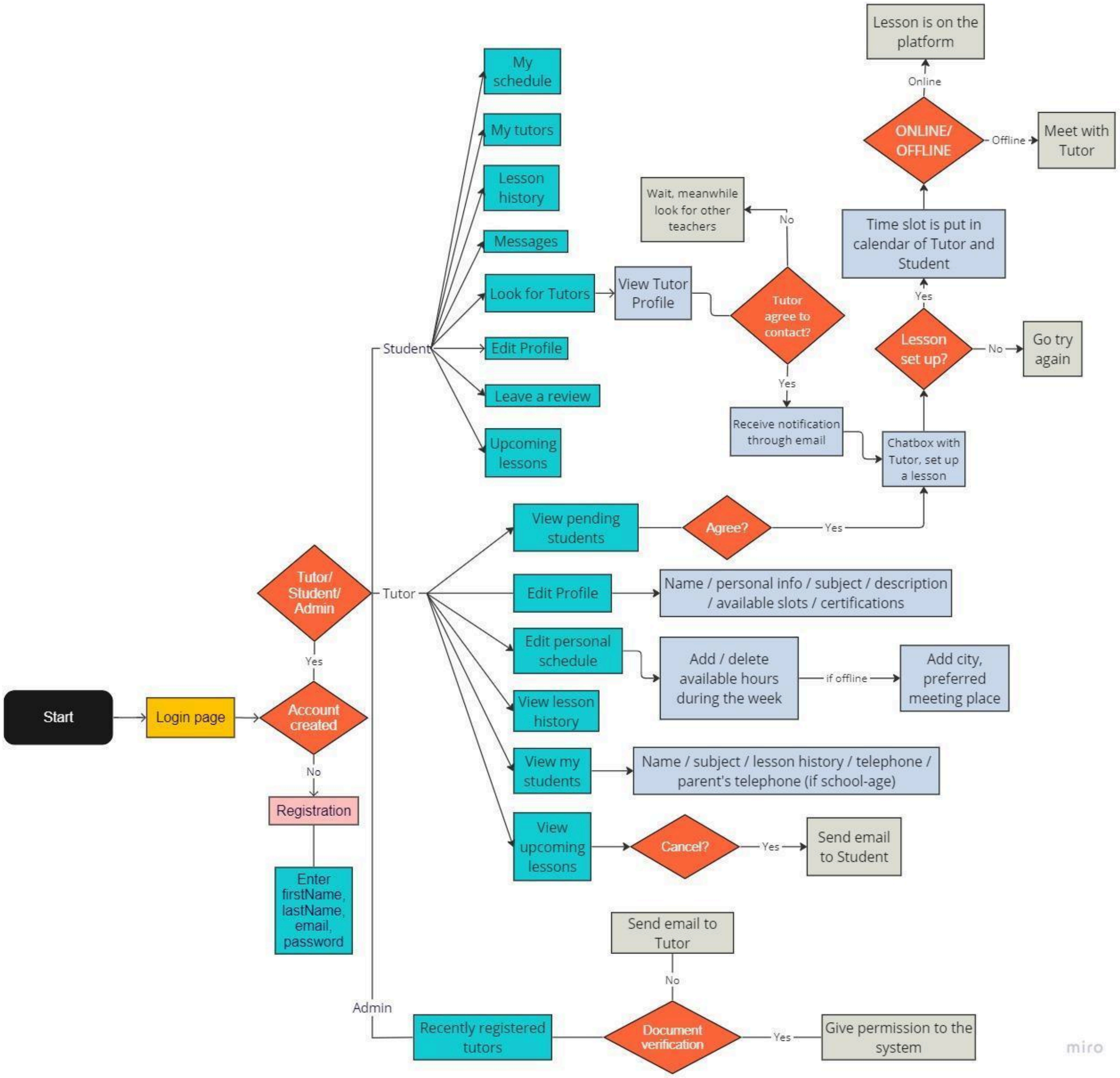

Start
Login page
Account created
Yes
No
Registration
Enter firstName, lastName, email, password
Tutor/ Student/ Admin
Student
My schedule
My tutors
Lesson history
Messages
Look for Tutors
View Tutor Profile
Tutor agree to contact?
No
Wait, meanwhile look for other teachers
Yes
Receive notification through email
Chatbox with Tutor, set up a lesson
Lesson set up?
No
Go try again
Yes
Time slot is put in calendar of Tutor and Student
ONLINE/ OFFLINE
Online
Lesson is on the platform
Offline
Meet with Tutor
Edit Profile
Leave a review
Upcoming lessons
Tutor
View pending students
Agree?
Yes
Edit Profile
Name / personal info / subject / description / available slots / certifications
Edit personal schedule
Add / delete available hours during the week
if offline
Add city, preferred meeting place
View lesson history
View my students
Name / subject / lesson history / telephone / parent's telephone (if school-age)
View upcoming lessons
Cancel?
Yes
Send email to Student
Admin
Recently registered tutors
Document verification
No
Send email to Tutor
Yes
Give permission to the system
miro

Figure 1. System flow diagram

The following is the UML diagram of our system. It describes the platform as a composition of classes, where each class, as a separate entity, interacts with other classes as defined by the relationships below.

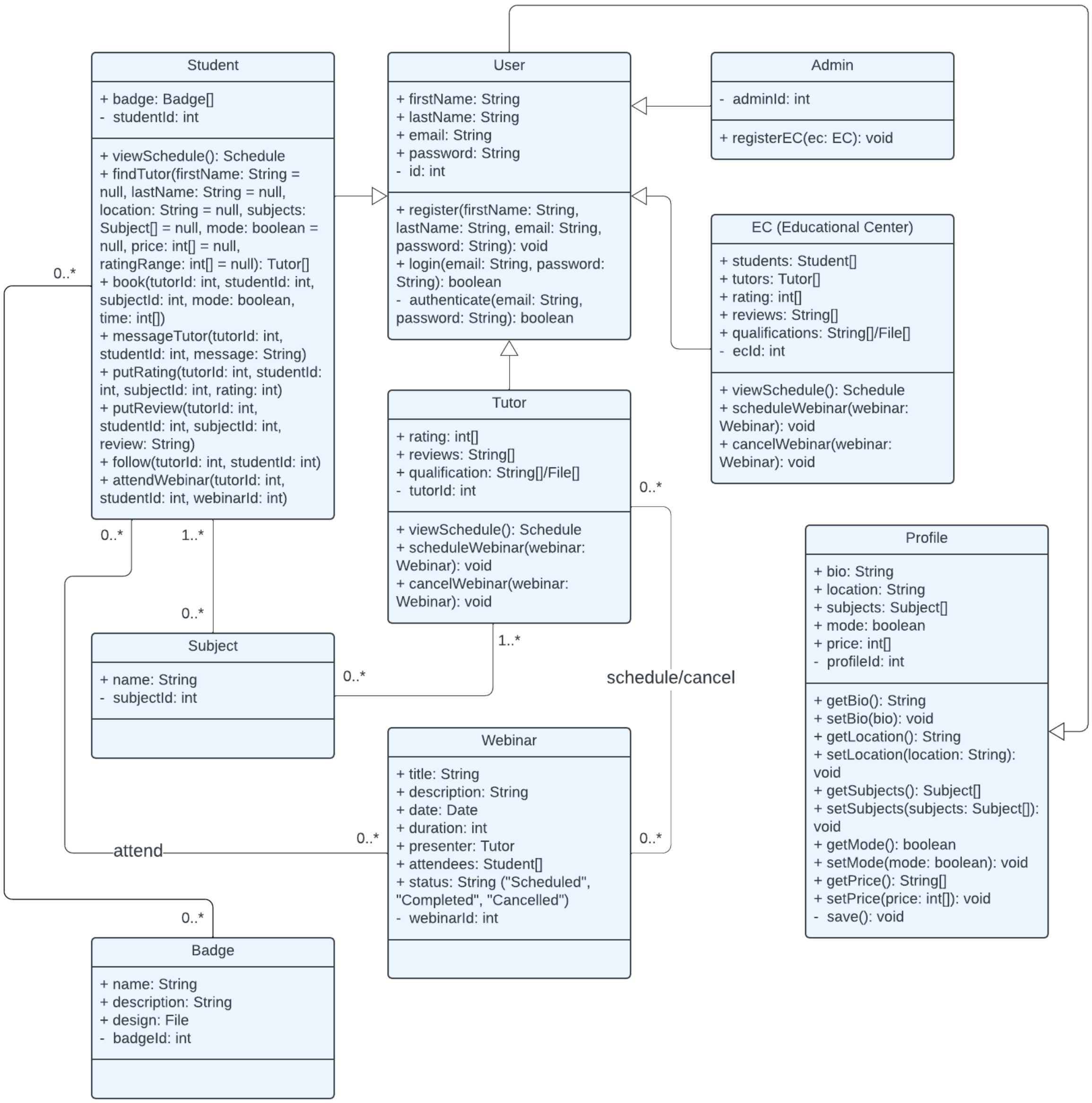

Student
+ badge: Badge[]
- studentId: int
+ viewSchedule(): Schedule
+ findTutor(firstName: String = null, lastName: String = null, location: String = null, subjects: Subject[] = null, mode: boolean = null, price: int[] = null, ratingRange: int[] = null): Tutor[]
+ book(tutorId: int, studentId: int, subjectId: int, mode: boolean, time: int[])
+ messageTutor(tutorId: int, studentId: int, message: String)
+ putRating(tutorId: int, studentId: int, subjectId: int, rating: int)
+ putReview(tutorId: int, studentId: int, subjectId: int, review: String)
+ follow(tutorId: int, studentId: int)
+ attendWebinar(tutorId: int, studentId: int, webinarId: int)
User
+ firstName: String
+ lastName: String
+ email: String
+ password: String
- id: int
+ register(firstName: String, lastName: String, email: String, password: String): void
+ login(email: String, password: String): boolean
- authenticate(email: String, password: String): boolean
Admin
- adminId: int
+ registerEC(ec: EC): void
EC (Educational Center)
+ students: Student[]
+ tutors: Tutor[]
+ rating: int[]
+ reviews: String[]
+ qualifications: String[]/File[]
- ecId: int
+ viewSchedule(): Schedule
+ scheduleWebinar(webinar: Webinar): void
+ cancelWebinar(webinar: Webinar): void
Tutor
+ rating: int[]
+ reviews: String[]
+ qualification: String[]/File[]
- tutorId: int
+ viewSchedule(): Schedule
+ scheduleWebinar(webinar: Webinar): void
+ cancelWebinar(webinar: Webinar): void
Profile
+ bio: String
+ location: String
+ subjects: Subject[]
+ mode: boolean
+ price: int[]
- profileId: int
+ getBio(): String
+ setBio(bio): void
+ getLocation(): String
+ setLocation(location: String): void
+ getSubjects(): Subject[]
+ setSubjects(subjects: Subject[]): void
+ getMode(): boolean
+ setMode(mode: boolean): void
+ getPrice(): String[]
+ setPrice(price: int[]): void
- save(): void
Subject
+ name: String
- subjectId: int
Webinar
+ title: String
+ description: String
+ date: Date
+ duration: int
+ presenter: Tutor
+ attendees: Student[]
+ status: String ("Scheduled", "Completed", "Cancelled")
- webinarId: int
Badge
+ name: String
+ description: String
+ design: File
- badgeId: int
0..*
0..*
1..*
0..*
0..*
1..*
0..*
schedule/cancel
attend
0..*
0..*
0..*

Figure 2. UML diagram for the system

Then, we created use case and sequence diagrams alongside page mockups for each main functional requirement. Specifically, use case diagrams were used to capture the primary flow of user interaction at a higher level of abstraction, while sequence diagrams were created to further understand the technical interactions between the system components.

1. User registration, profile creation

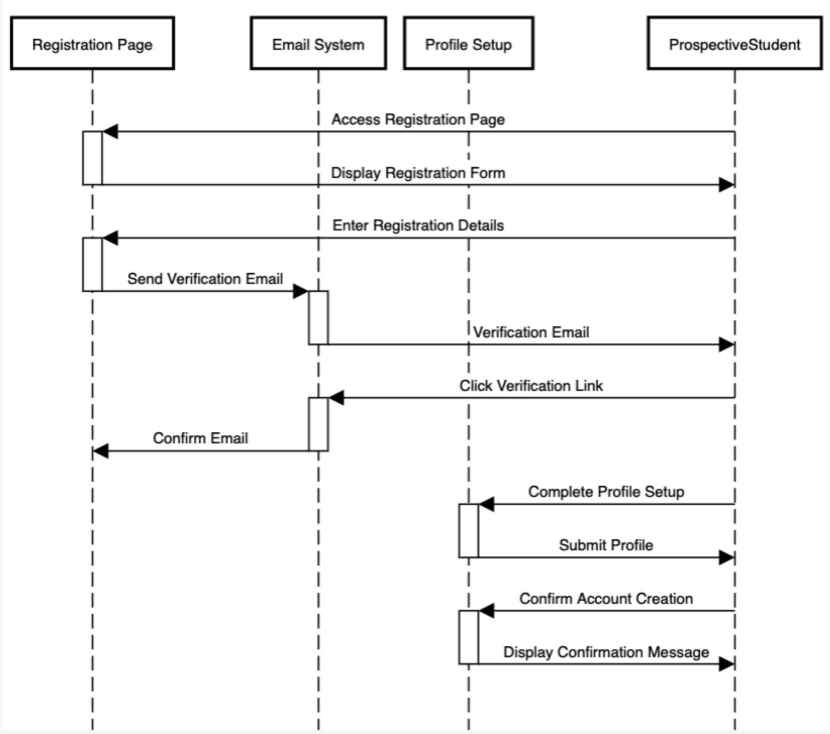
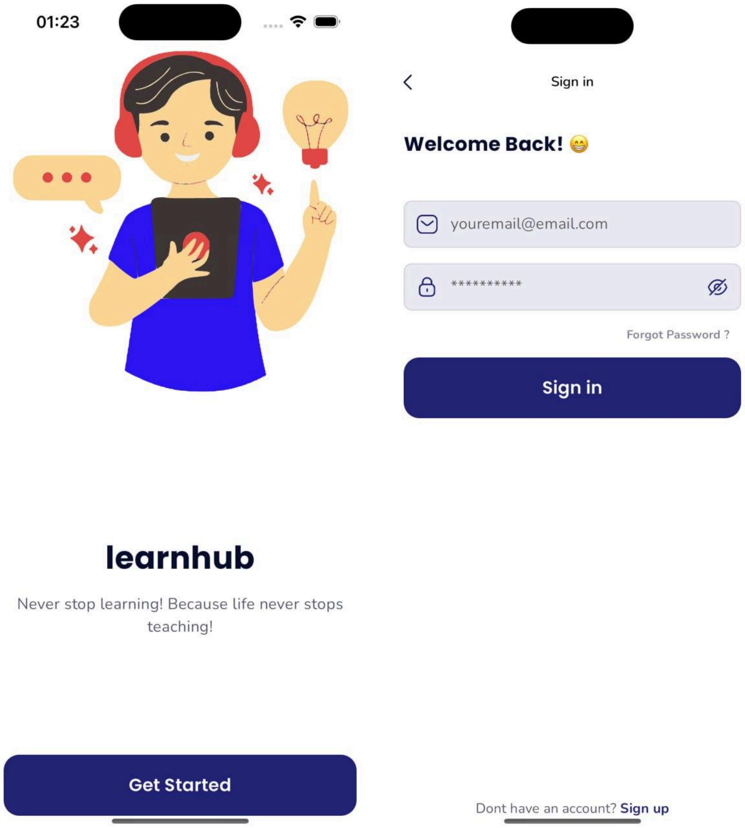


Figure 3. Sequence diagram for student registration and profile creation



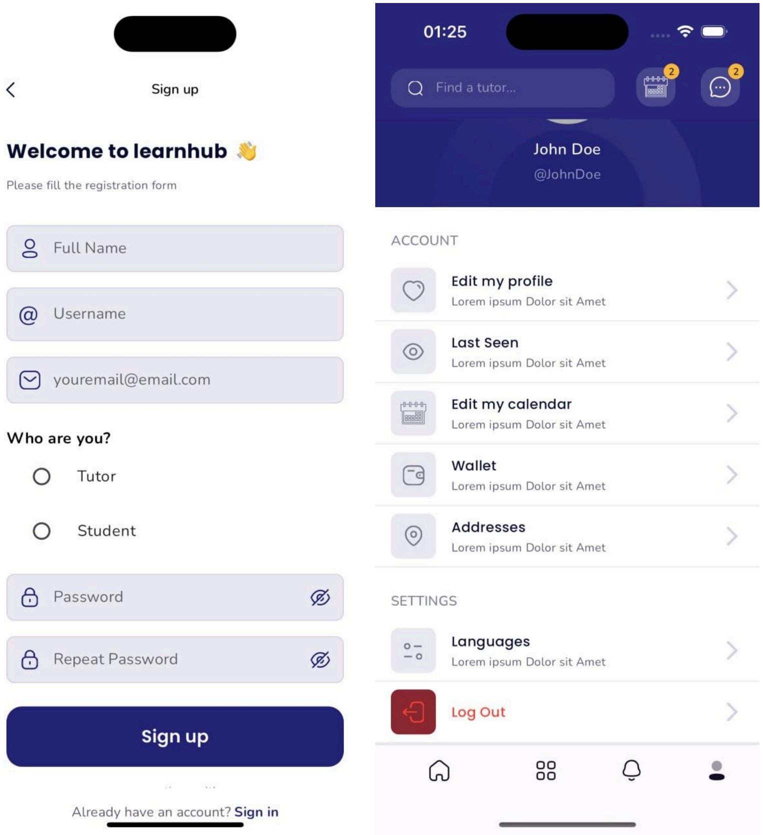
learnhub

Never stop learning! Because life never stops teaching!

Welcome Back! 😁

youremail@email.com

****



Sign up

Welcome to learnhub 👋

Please fill the registration form

Full Name

Username

youremail@email.com

Who are you?

○ Tutor

○ Student

Password

Repeat Password

Sign up

Already have an account? Sign in

01:25

Find a tutor...

John Doe

@JohnDoe

ACCOUNT

Edit my profile
Lorem ipsum Dolor sit Amet

Last Seen
Lorem ipsum Dolor sit Amet

Edit my calendar
Lorem ipsum Dolor sit Amet

Wallet
Lorem ipsum Dolor sit Amet

Addresses
Lorem ipsum Dolor sit Amet

SETTINGS

Languages
Lorem ipsum Dolor sit Amet

Log Out

Figure 4. Mobile app screenshots for student registration and profile creation

2. Tutor search and filtering

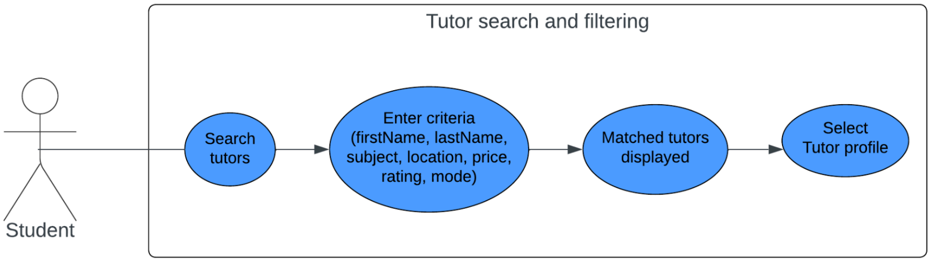

Tutor search and filtering
Student
Search tutors
Enter criteria (firstName, lastName, subject, location, price, rating, mode)
Matched tutors displayed
Select Tutor profile

Figure 5. Use case diagram for tutor search and filtering

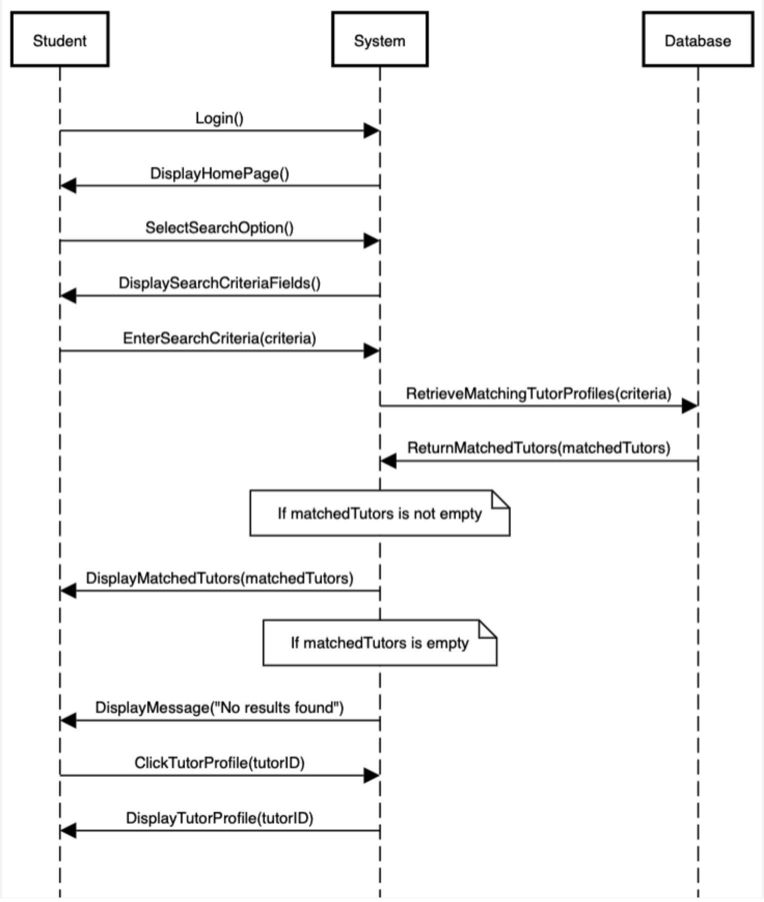
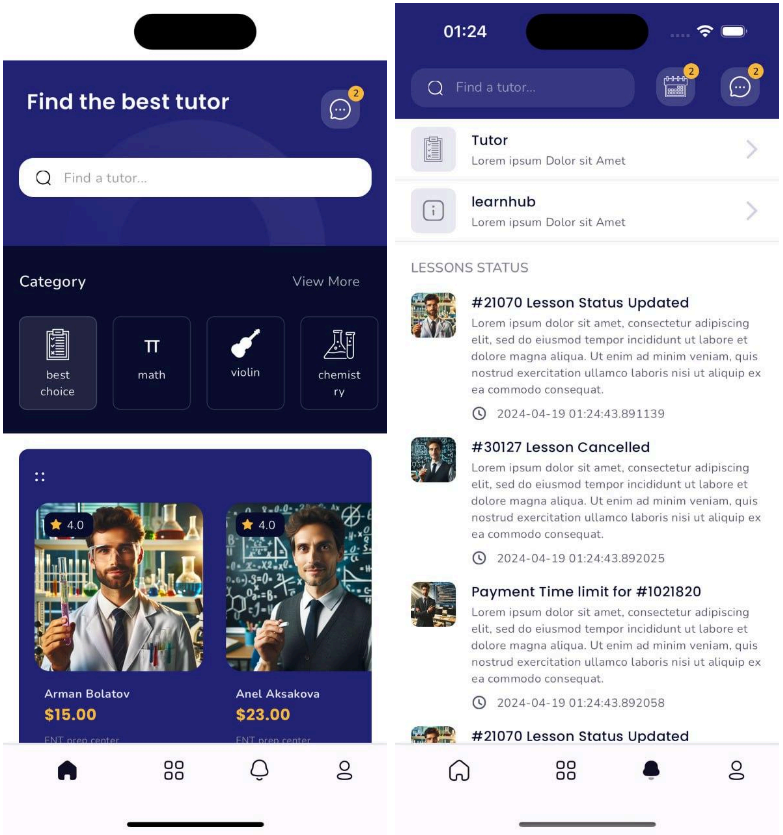

Student
System
Database
Login()
DisplayHomePage()
SelectSearchOption()
DisplaySearchCriteriaFields()
EnterSearchCriteria(criteria)
RetrieveMatchingTutorProfiles(criteria)
ReturnMatchedTutors(matchedTutors)
If matchedTutors is not empty
DisplayMatchedTutors(matchedTutors)
If matchedTutors is empty
DisplayMessage("No results found")
ClickTutorProfile(tutorID)
DisplayTutorProfile(tutorID)

Figure 6. Sequence diagram for tutor search and filtering



Find the best tutor

Find a tutor...

Category

View More

best choice

math

violin

chemistry

4.0

Arman Bolatov

$15.00

ENT prep center

4.0

Anel Aksakova

$23.00

ENT prep center

01:24

Find a tutor...

Tutor

Lorem ipsum Dolor sit Amet

learnhub

Lorem ipsum Dolor sit Amet

LESSONS STATUS

#21070 Lesson Status Updated

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

2024-04-19 01:24:43.891139

#30127 Lesson Cancelled

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

2024-04-19 01:24:43.892025

Payment Time limit for #1021820

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

2024-04-19 01:24:43.892058

#21070 Lesson Status Updated

Figure 7. Mobile app screenshots for tutor search and lesson status

3. Booking a class

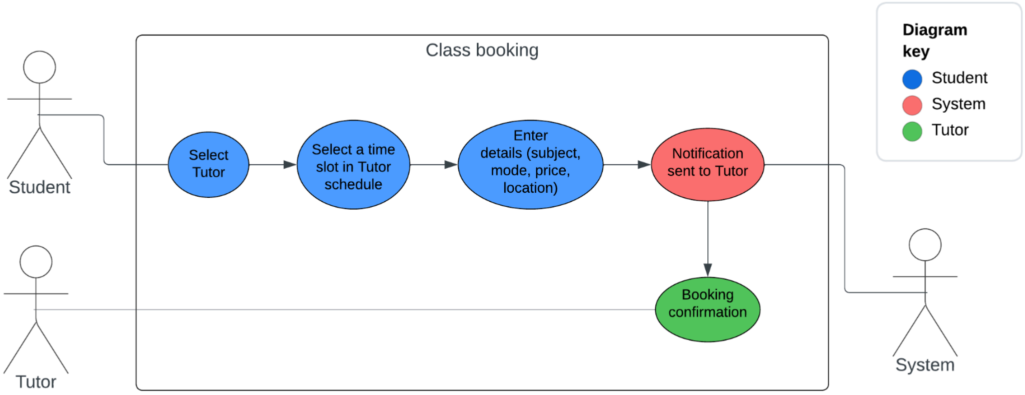

Class booking
Student
Tutor
System
Select Tutor
Select a time slot in Tutor schedule
Enter details (subject, mode, price, location)
Notification sent to Tutor
Booking confirmation
Diagram key
Student
System
Tutor

Figure 8. Use case diagram for booking a class

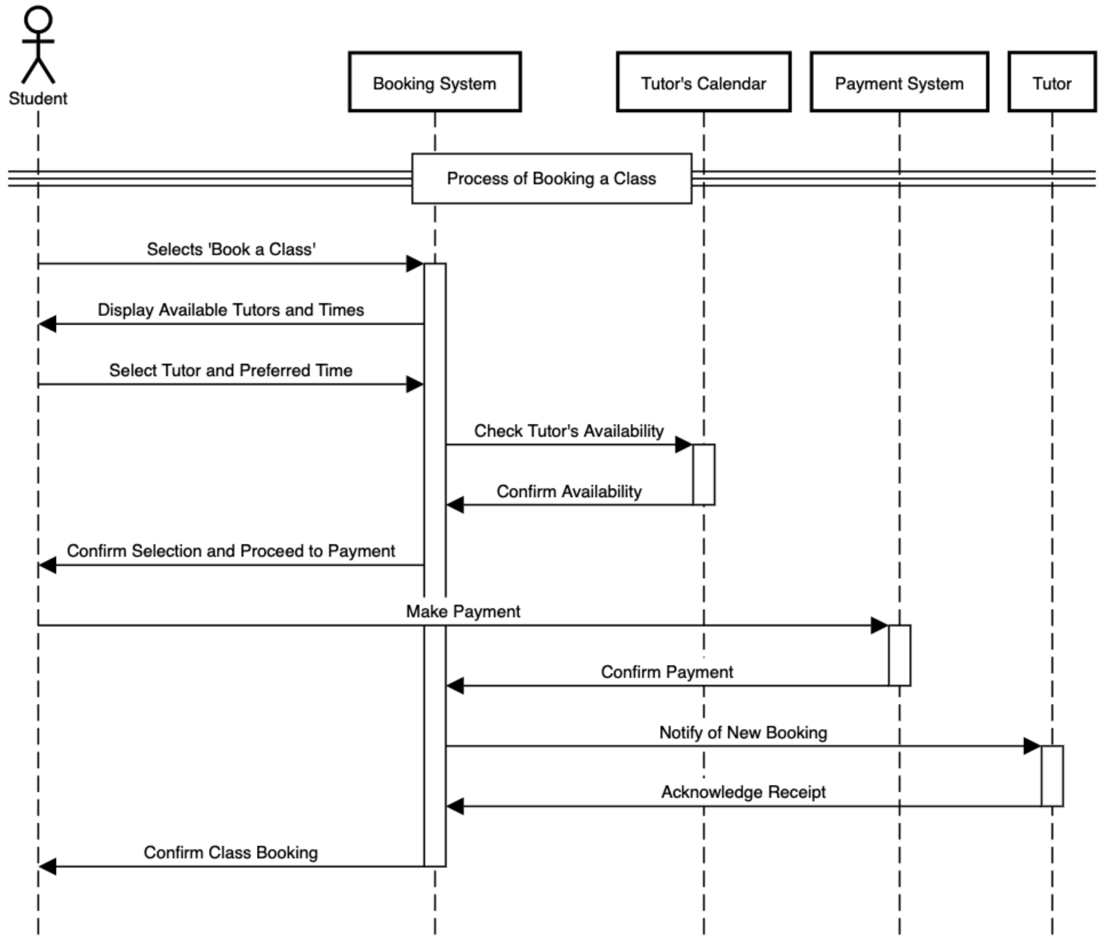
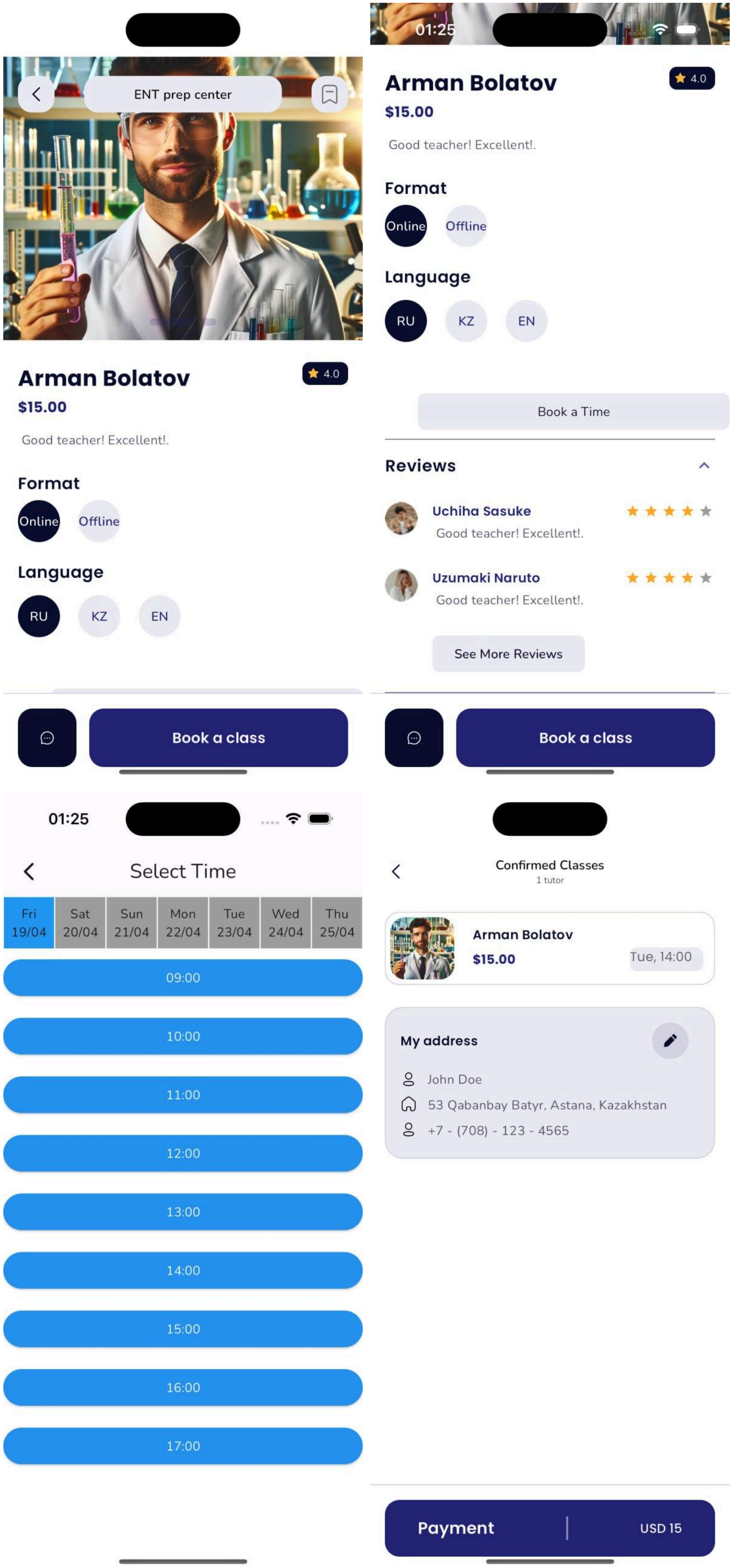


Figure 9. Sequence diagram for booking a class



ENT prep center

Arman Bolatov

★ 4.0

$15.00

Good teacher! Excellent!.

Format

Online Offline

Language

RU KZ EN

Book a class

01:25

Arman Bolatov

★ 4.0

$15.00

Good teacher! Excellent!.

Format

Online Offline

Language

RU KZ EN

Book a Time

Reviews

Uchiha Sasuke ★★★★★

Good teacher! Excellent!.

Uzumaki Naruto ★★★★★

Good teacher! Excellent!.

See More Reviews

Book a class

01:25

Select Time

Fri 19/04	Sat 20/04	Sun 21/04	Mon 22/04	Tue 23/04	Wed 24/04	Thu 25/04

09:00

10:00

11:00

12:00

13:00

14:00

15:00

16:00

17:00

Confirmed Classes

1 tutor

Arman Bolatov

$15.00

Tue, 14:00

My address

John Doe

53 Qabanbay Batyr, Astana, Kazakhstan

+7 - (708) - 123 - 4565

Payment | USD 15

Figure 10. Mobile app screenshots for booking a class

4. Messaging

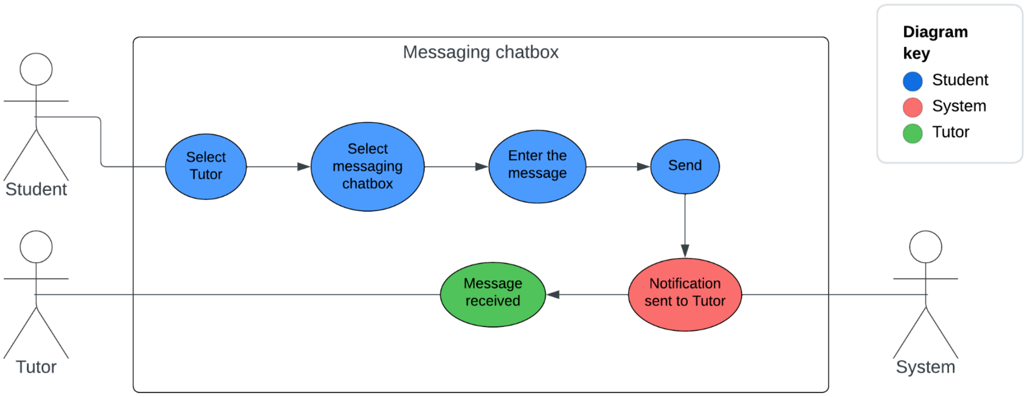


Figure 11. Use case diagram for Student messaging a Tutor

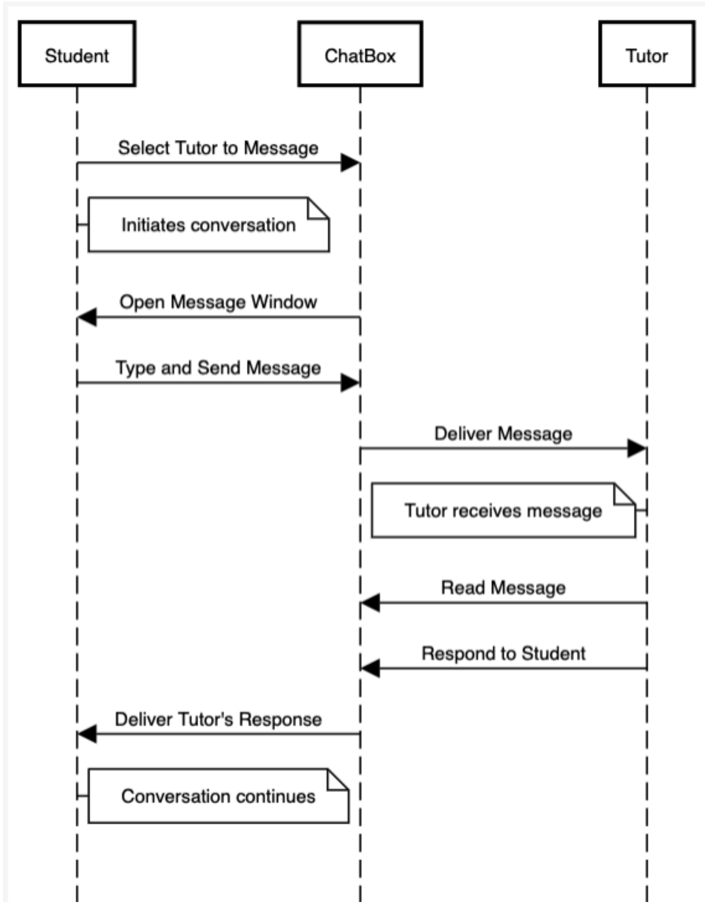
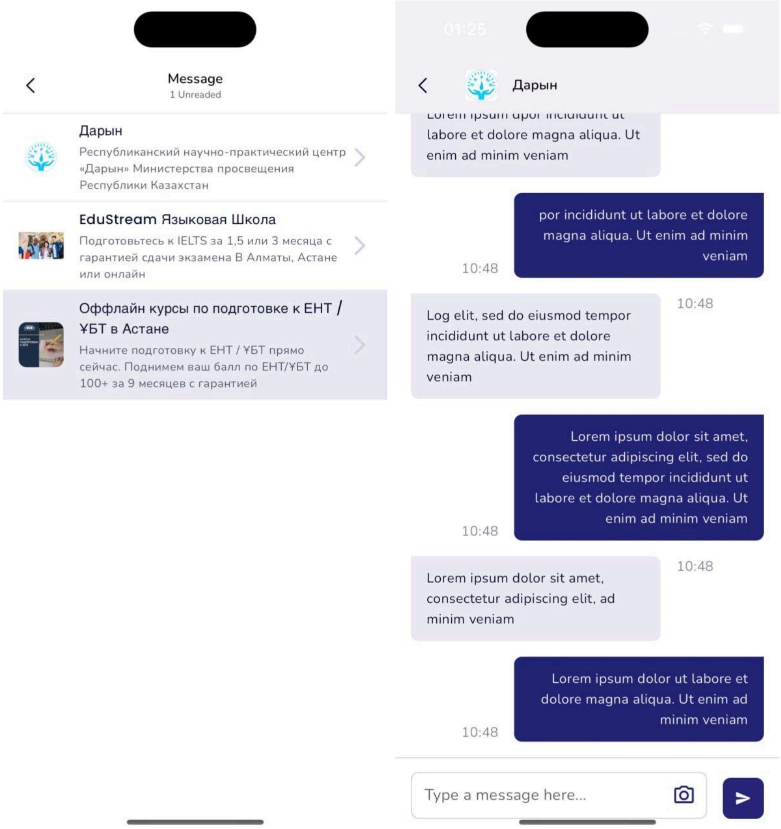


Figure 12. Sequence diagram for Student messaging a Tutor



Message
1 Unreaded

Дарын
Республиканский научно-практический центр «Дарын» Министерства просвещения Республики Казахстан

EduStream Языковая Школа
Подготовьтесь к IELTS за 1,5 или 3 месяца с гарантией сдачи экзамена В Алматы, Астане или онлайн

Оффлайн курсы по подготовке к ЕНТ / ҰБТ в Астане
Начните подготовку к ЕНТ / ҰБТ прямо сейчас. Поднимем ваш балл по ЕНТ/ҰБТ до 100+ за 9 месяцев с гарантией

01:25

Дарын

Lorem ipsum upor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam

por incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam
10:48

Log elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam
10:48

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam
10:48

Lorem ipsum dolor sit amet, consectetur adipiscing elit, ad minim veniam
10:48

Lorem ipsum dolor ut labore et dolore magna aliqua. Ut enim ad minim veniam
10:48

Type a message here...

Figure 13. Mobile app screenshots for Student messaging a Tutor

5. Rating and review

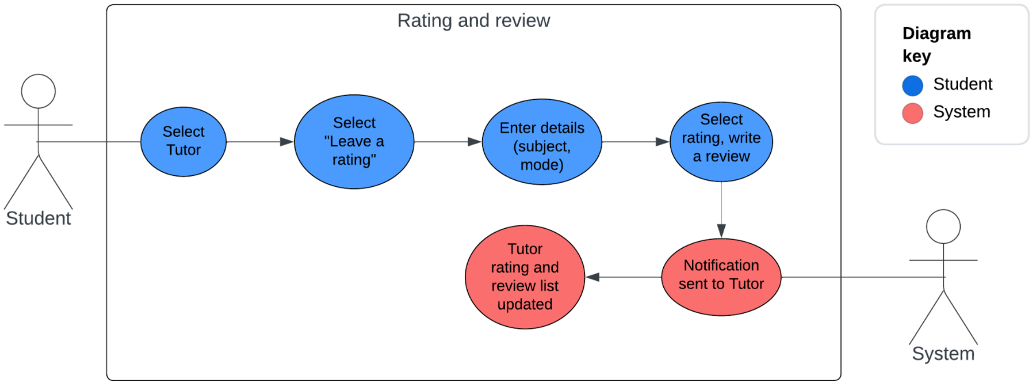

Rating and review
Student
Select Tutor
Select "Leave a rating"
Enter details (subject, mode)
Select rating, write a review
Notification sent to Tutor
Tutor rating and review list updated
System
Diagram key
● Student
● System

Figure 14. Use case diagram for Student leaving a rating and a review for a Tutor

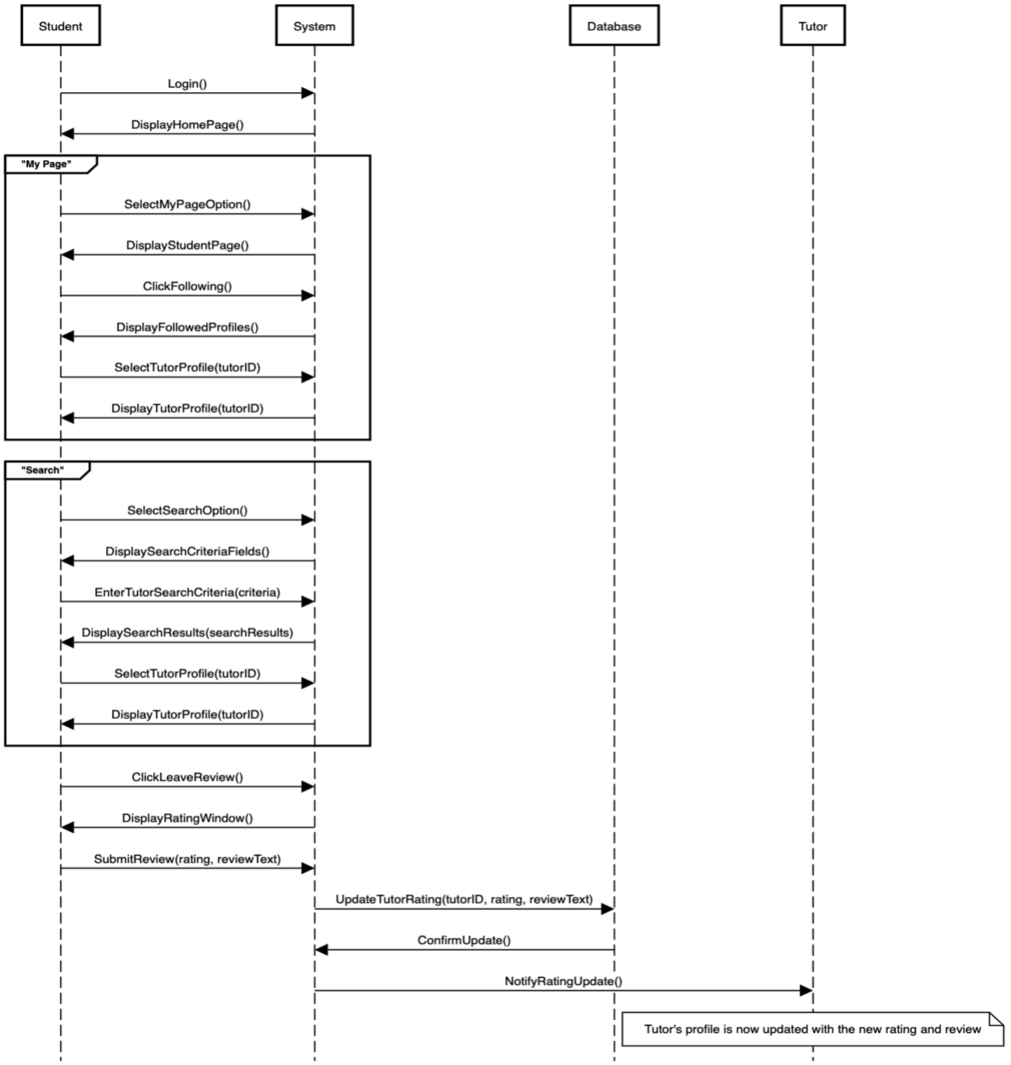
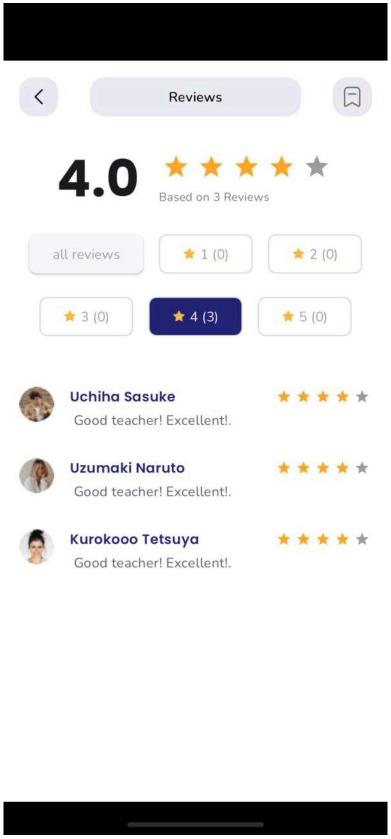

Student
System
Database
Tutor
Login()
DisplayHomePage()
"My Page"
SelectMyPageOption()
DisplayStudentPage()
ClickFollowing()
DisplayFollowedProfiles()
SelectTutorProfile(tutorID)
DisplayTutorProfile(tutorID)
"Search"
SelectSearchOption()
DisplaySearchCriteriaFields()
EnterTutorSearchCriteria(criteria)
DisplaySearchResults(searchResults)
SelectTutorProfile(tutorID)
DisplayTutorProfile(tutorID)
ClickLeaveReview()
DisplayRatingWindow()
SubmitReview(rating, reviewText)
UpdateTutorRating(tutorID, rating, reviewText)
ConfirmUpdate()
NotifyRatingUpdate()
Tutor's profile is now updated with the new rating and review

Figure 15. Sequence diagram for Student leaving a rating and a review for a Tutor



Reviews

4.0 ★★★★☆

Based on 3 Reviews

all reviews | ★ 1 (0) | ★ 2 (0)

★ 3 (0) | ★ 4 (3) | ★ 5 (0)

Uchiha Sasuke ★★★★☆

Good teacher! Excellent!.

Uzumaki Naruto ★★★★☆

Good teacher! Excellent!.

Kurokooo Tetsuya ★★★★☆

Good teacher! Excellent!.

Figure 16. Mobile app screenshot for Tutor rating and review

6. Profile “following”

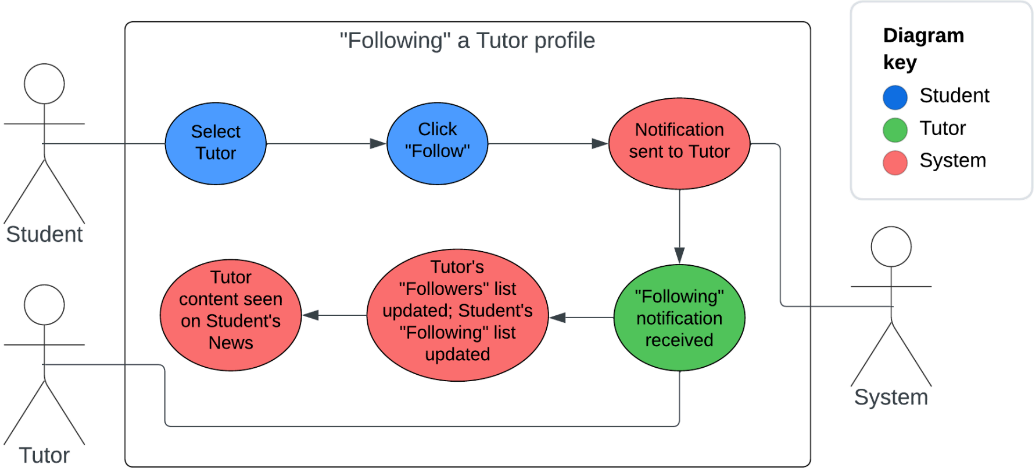

"Following" a Tutor profile
Student
Tutor
System
Select Tutor
Click "Follow"
Notification sent to Tutor
"Following" notification received
Tutor's "Followers" list updated; Student's "Following" list updated
Tutor content seen on Student's News
Diagram key
Student
Tutor
System

Figure 17. Use case diagram for Student “following” a Tutor

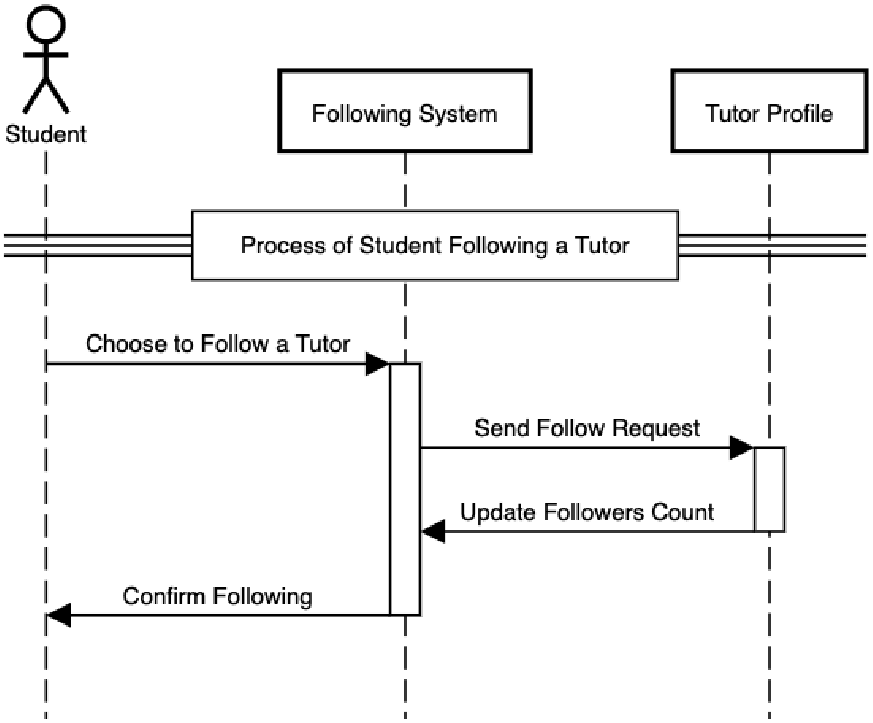
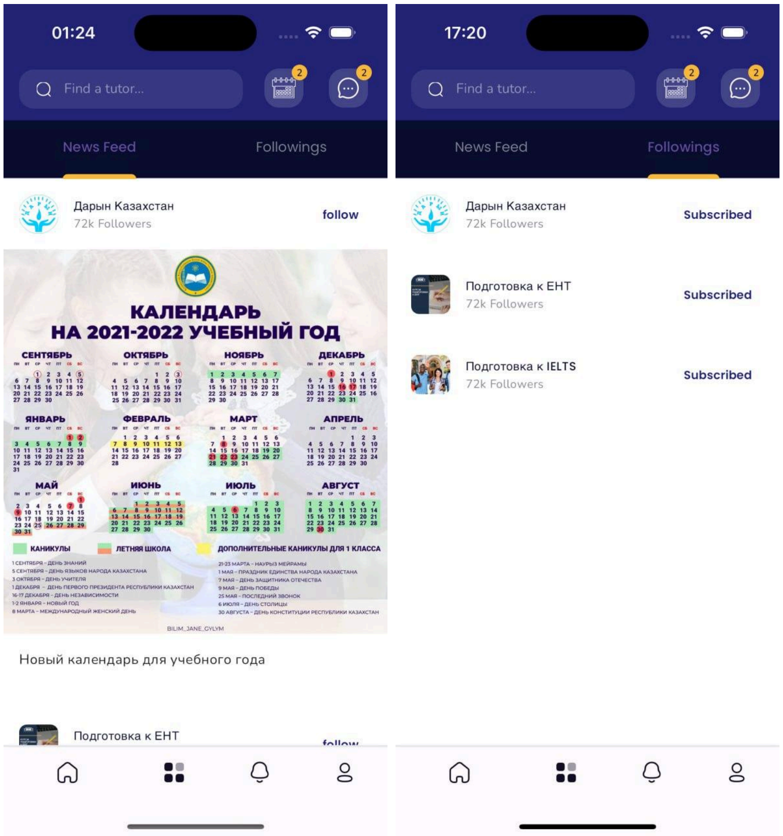


Figure 18. Sequence diagram for Student “following” a Tutor



Дарын Казахстан
72k Followers
follow

Новый календарь для учебного года

Подготовка к ЕНТ
follow

Дарын Казахстан
72k Followers
Subscribed

Подготовка к ЕНТ
72k Followers
Subscribed

Подготовка к IELTS
72k Followers
Subscribed

Figure 19. Mobile app screenshots for news feed and Student “followings”

Team responsibilities:

Zhassulan - Front-end development

- Designed and developed the website user interface using Next.js and React
- Set up the role based authentication utilizing server-side rendering
- Utilized the cutting edge technologies like React Server Components and Server Actions for maximum performance

Dilnas - Back-end development

- Set up the database and data storage using PostgreSQL and Hibernate
- Designed the system logic
- Designed and developed APIs
- Testing and quality assurance of the backend of the system
- Collaborated with Zhassulan - frontend developer on requests handling and web app architecture

Aruay, Bibigul - Mobile app development

- Designed and implemented core UI components using Flutter widgets
- Developed modular pages for key features like user and tutor profiles, messaging, class booking, etc.
- Tested app functionality across Android and iOS device sizes

Frameworks and third-party tools used:

1. Figma - for designing initial page layouts
2. React - for implementing the UI of web app
3. Next.js - server-side rendering of the React app
4. Java - for implementing the logic of the system
5. Spring Boot - for setting up, configuring and running the app
6. Spring JPA - for storing data in relational database
7. Spring Security - to develop authentication, authorization and provide protection
8. Hibernate - to map relational database to Java objects
9. Maven - project management and build tool
10. TypeScript - for type-safe JavaScript development
11. PostgreSQL - database management system to store data and communicate with database servers
12. Flutter - for a cross-platform mobile app development

The following pubspec.yaml dependencies were utilized in building the Flutter mobile app:

- flutter: The Flutter SDK itself which provides core Flutter functionality.
- flutter_svg: Allows rendering SVG images from assets. Used for logos.
- flutter_rating_bar: Reusable widget to select and display star ratings. Used for tutor profile ratings.
- cupertino_icons: Provides Cupertino style icons for iOS-style widgets. Used for tabs, buttons etc.
- intl: Internationalization library for formatting numbers, dates etc. Used for date formatting.
- pin_code_fields: Widget for building pin/OTP input screens. Used for authentication.
- smooth_page_indicator: Customizable page indicator for viewPagers/paginations. Used for onboarding.

Project Execution (15%)

Design decisions:

In the planning stage, we created sequence diagrams and use case diagrams to map out the key user flows and interactions with the system. These served as a starting point to develop initial visual mockups using Figma. This helped ensure the visual designs followed a consequential logical flow that aligned with our analysis of intended usage.

Changes in the project:
Initially, we proposed developing the back-end APIs and services using the Go programming language. However, later on we encountered some challenges with Go, and it was decided to rather switch to Java for the back-end platform. Specifically, we opted to implement the backend services using Spring framework (Spring Boot, Spring JPA, and Spring Security). Making this change allowed us to reduce the technical risk and ensured we could deliver the project within the timeline.

What went wrong and how we dealt with the problems encountered:
With everyone having different commitments outside of the project, scheduling clashes posed an initial difficulty as some members had work in addition to studies while others focused on graduate school applications through the semester.

To make things a bit easier with our scheduling challenges, we decided to split into two smaller teams. Dilnas and Zhassulan focused on backend and frontend web development, while Aruay and Bibigul tackled developing the mobile app on Flutter. This way, each pair could check in with each other more fluidly without depending on everyone's schedules aligning.

Then, we scheduled online meetings when we hit certain project milestones or needed some clarifications regarding the project. Dividing the work this way definitely made communication simpler and allowed us to stay on track throughout the semester.

Evaluation (20%)

To evaluate whether our web and mobile applications successfully solved the problem of a seamless and effective tutor-student search and communication, we conducted user testing and collected feedback through a questionnaire.

We had 11 potential users try out both website and mobile app. After interacting with each system, the users were asked to complete a questionnaire through Google Forms. The questionnaire contained questions about the usability, design, difficulty finding any functionality and overall experience of both systems. It utilized a mixture of rating scale and open-ended questions.

We received completed questionnaires from all 11 users. The results were automatically compiled and analyzed in Google Forms. Charts and graphs were also generated to visualize the quantitative rating scale responses.

Some of the key findings from the questionnaire data include:

- On average, users rated how intuitive the systems are and ease of use of both platforms around 4.5 out of 5. This indicates that the systems have generally good usability.
- Most users (73%) did not have any trouble finding the necessary information on the platforms, however, a few noted that since the icons in the mobile application do not have separate labels, they had to try out all four main pages to fully understand what each icon means.
- As an area for improvement, a few users noted how to improve the “booking” section.

After analyzing the questionnaire results, the following changes were made to the system:

1. Previously, the class time details were only shown when initially choosing a time slot, but not afterwards when selecting other features or proceeding to payment, which caused uncertainty for users about whether they selected the correct slot. Now, the time details continue to be displayed even after selecting a slot.
2. Design improvements for better color coordination on the platform.
3. Some icons were replaced with more intuitive ones.

The full list of questions along with graphical summaries of user answers are provided in the Appendix.

Conclusion and possible future work (5%)

Overall, the goal of this project was to develop fully functional web and mobile applications for effectively connecting students and tutors, and the project was successful in delivering a this solution.

We also got to present our project at the International Scientific and Practical Conference "Industrial Development: Technologies for People and Services in the Era of Innovation" on December 5, 2023.

In the future, the application can be further improved by implementing scheduling reminders and cancellation policies. Current messaging system can also be enhanced to include group chats and file sharing.

References (5%)

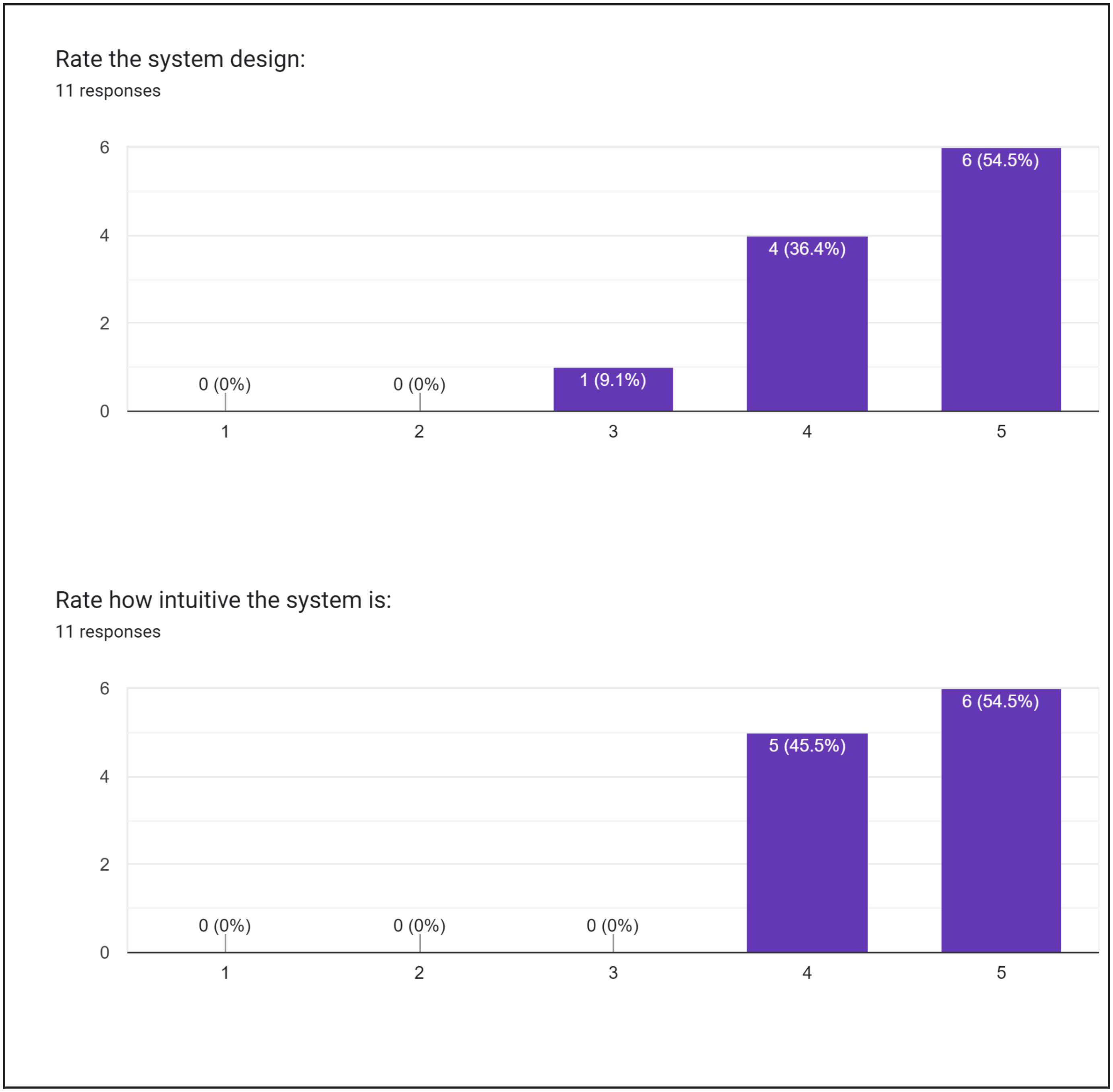
[1] LRNKEY. Online tutoring platform for private lessons. https://lrnkey.com/

[2] Preply. Best online language tutors and classes. https://preply.com/

[3] Skooli. Online tutoring. https://www.skooli.com/

Appendix

User feedback summary from Google Forms questionnaire:

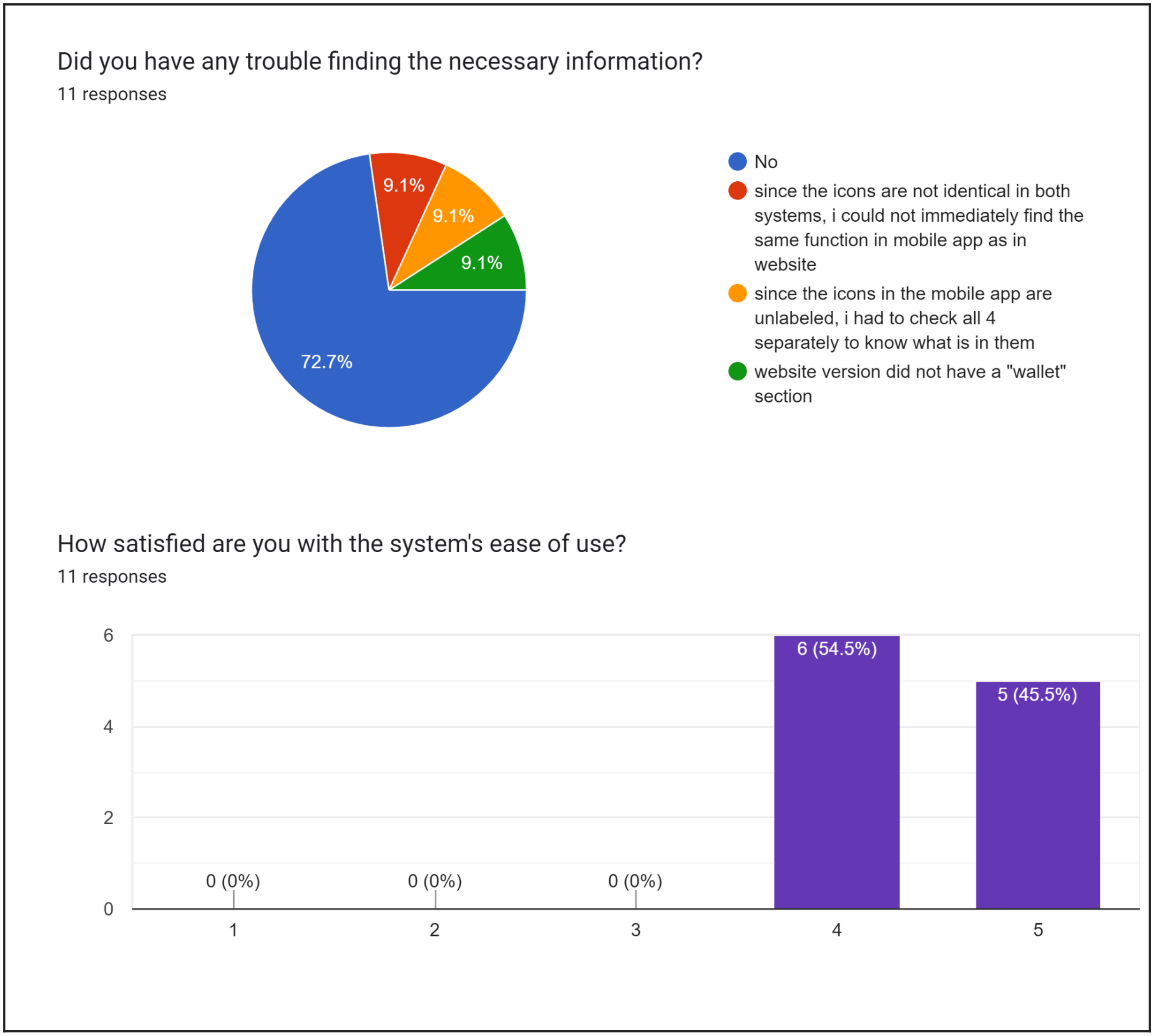


Rate the system design:

11 responses

Rate how intuitive the system is:

11 responses



Did you have any trouble finding the necessary information?

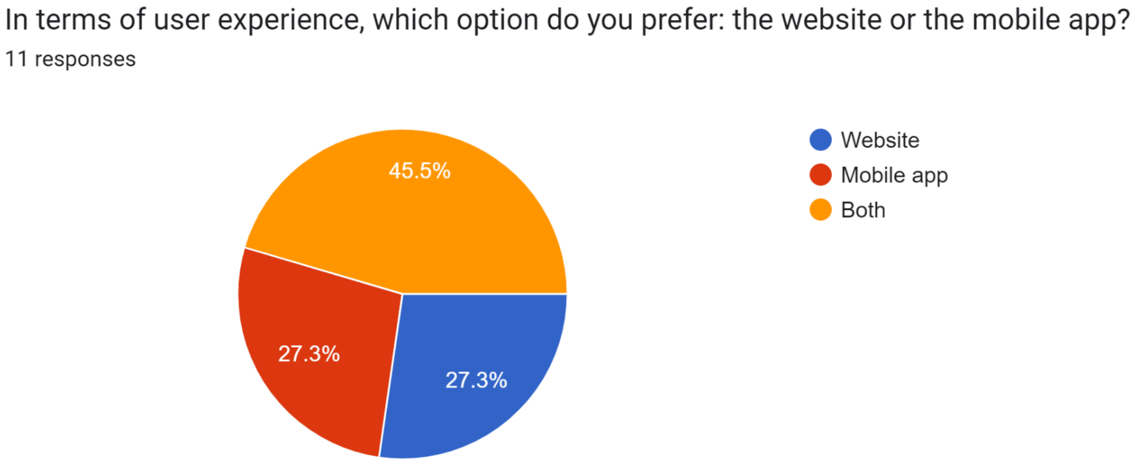
11 responses

9.1%
9.1%
9.1%
72.7%
No
since the icons are not identical in both systems, i could not immediately find the same function in mobile app as in website
since the icons in the mobile app are unlabeled, i had to check all 4 separately to know what is in them
website version did not have a "wallet" section

How satisfied are you with the system's ease of use?

11 responses

6
4
2
0
0 (0%)
0 (0%)
0 (0%)
6 (54.5%)
5 (45.5%)
1
2
3
4
5



In terms of user experience, which option do you prefer: the website or the mobile app?

11 responses

Please share your feedback about both systems:

1. I liked the color choices but there was no option for direct payment when booking a class, is it done separately?
2. I haven't seen a similar platform in Kazakhstan, i think it can help many freelance tutors (especially students)
3. I would prefer if the two systems had identical design
4. Overall, i liked the systems but i did not understand how skills and achievements are earned in the website
5. Both platforms are intuitive and easy to use, i just usually use my phone with apps more than a website
6. Both systems are nice, the website has the main icons labeled making it easier
7. I usually use mobile versions of everything
8. The class booking in calendar is only for 1 week in advance, i cannot book a class for the upcoming month for example
9. The platform is nice and will be used by many students if eventually deployed
10. While booking a class and choosing a slot on mobile app, the details are not displayed on the screen before payment so i had to re-check if it was correctly selected
11. No option of selecting only certain classes for payment/booking, no separate "wishlist" or something to save certain potential tutor profiles without actually booking a class