

**EZSTUDY**

LET US HELP U!



**Eli Greenblatt- 316302934**



**Naor Ladani - 318664190**

**Itamar Cohen - 209133826**



**Evyatar Yosef - 207467820**

**Roy Asraf - 302211958**

**Software Engineering**



# Division of Roles

Eli Greenblatt - Team Lead

Naor Ladani - Messages management between Teacher and Student, Meeting Scheduler

Itamar Cohen - Student and Teacher registration, updating profile of the users and view their details

Evyatar Yosef - Database Management and connection to the project, Searching for Teachers

Roy Asraf - Front End, Designing the screens and transitions, Links of teacher

# What is EZstudy?

- Find teachers or students for private lessons.
- Schedule appointments.
- Stay updated.



# Why do we need it?

- Spam Messages in unwanted forums.
- Convenience user interface.
- Reserves.



# How do we solve this situation

- Convenient Android Application
- Appealing user interface design
- Minimalist way to schedule a private lesson
- A targeted approach to identify the most suitable teacher.
- Teachers can upload links and information



# Our app main idea and goals

**Our main idea is to make the process of scheduling private lessons at the university a simpler process.**

- Schedule private lessons quickly and easily.
- Find the private teacher that suits.
- Tutor will be able to describe himself
- Schedule according the tutor.
- Chat between the tutor and the student.

# System Overview

- Android Application using Android Studio.
- Data base: firebase
- two users: **Teachers** and **Students**.
- Both of the users firstly have to register in order to use the app.

**EZSTUDY**  
*Let us help u'*



Username

Password

Login

Register

# System Requirements

## Non-Functional Requirements

1. Performance
2. Security.
3. Usability.
4. Direct.

## Functional Requirements

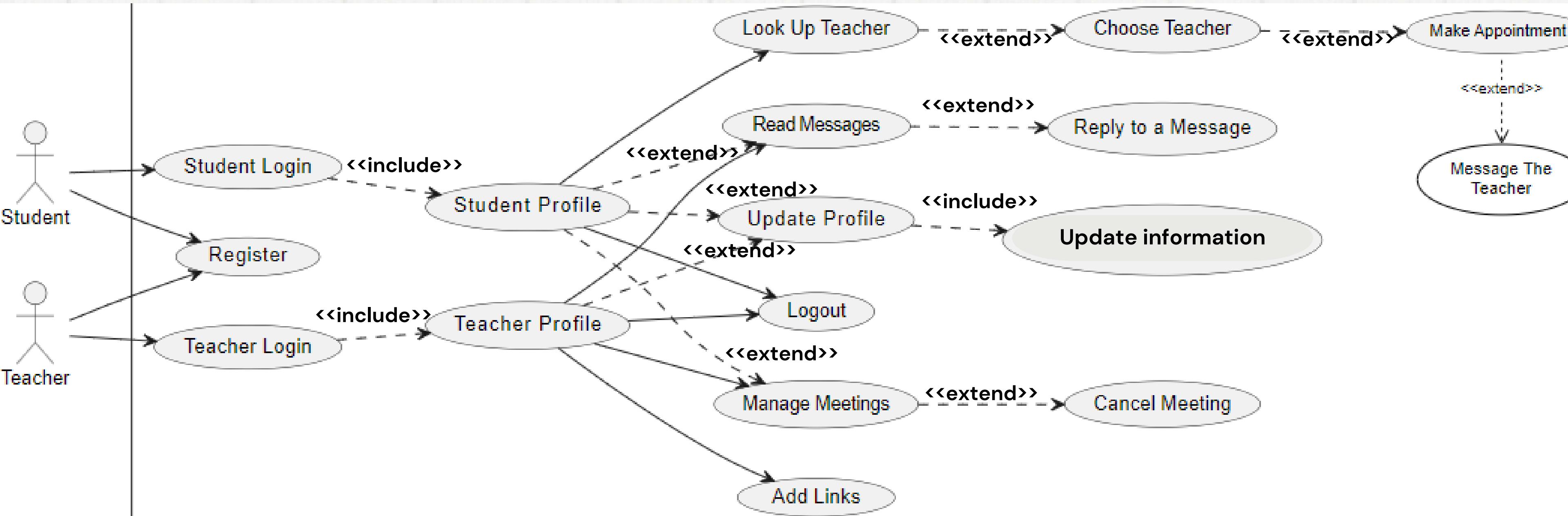
1. User Registration.
2. Teacher Profile Creation.
3. Sharing Links.
4. Profile Update.
5. Teacher Search.
6. Lesson Scheduling.
7. Messaging System.

# Main Features

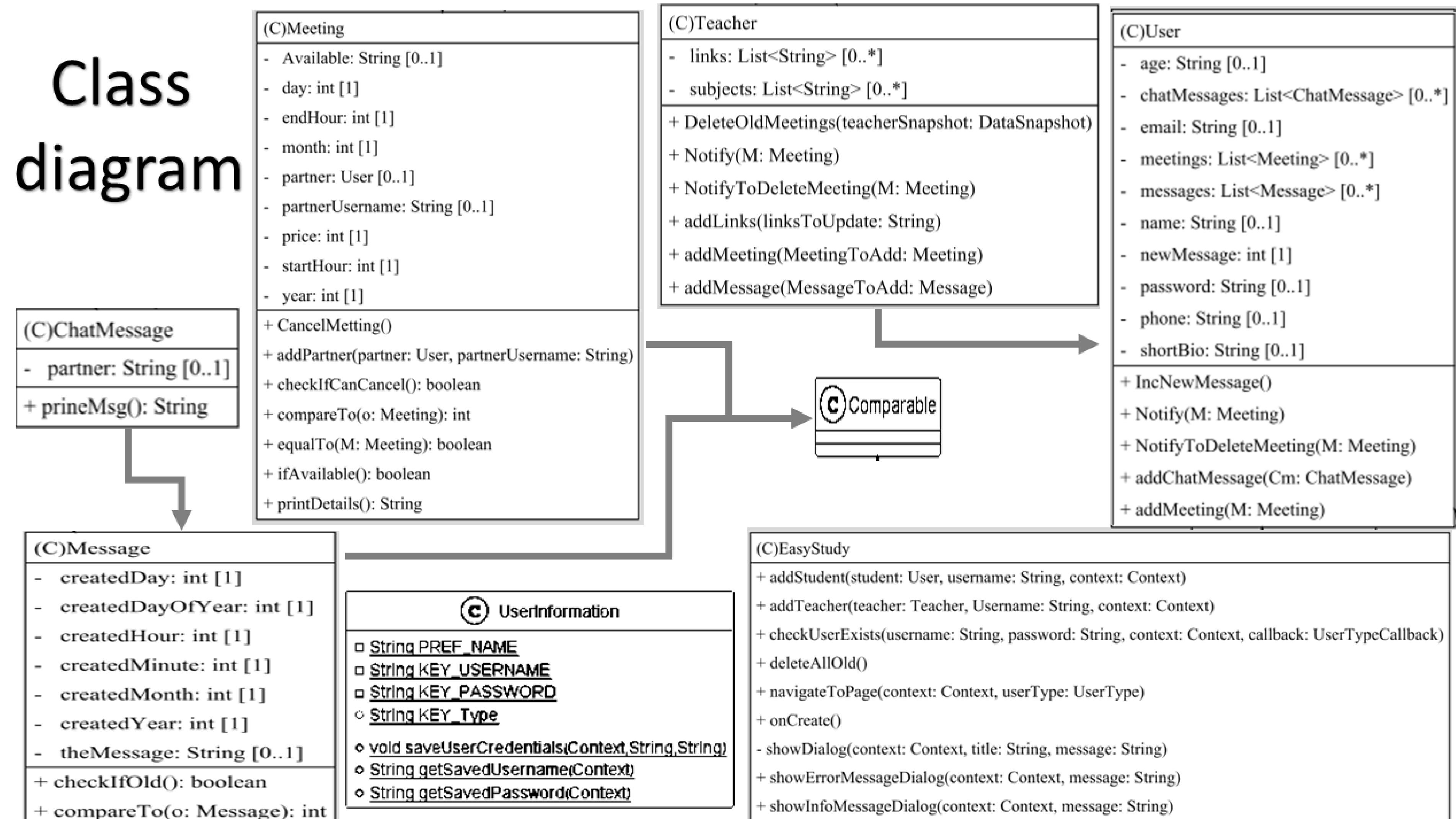
- 1. User Profile Creation** - Separate profiles for student and teacher.
- 2. Profile Update** - Anytime update your profile for your needs.
- 3. Sharing Links** - Teacher can publish efficient links.
- 4. Teacher Search** - Students can search by name and subject.
- 5. Lesson Scheduling** - Teacher publish when he available and student have to schedule the lesson.
- 6. Realtime Availability** - Appointments availability updates in realtime.
- 7. Meetings Management** - Students and teachers can simply delete their meeting.

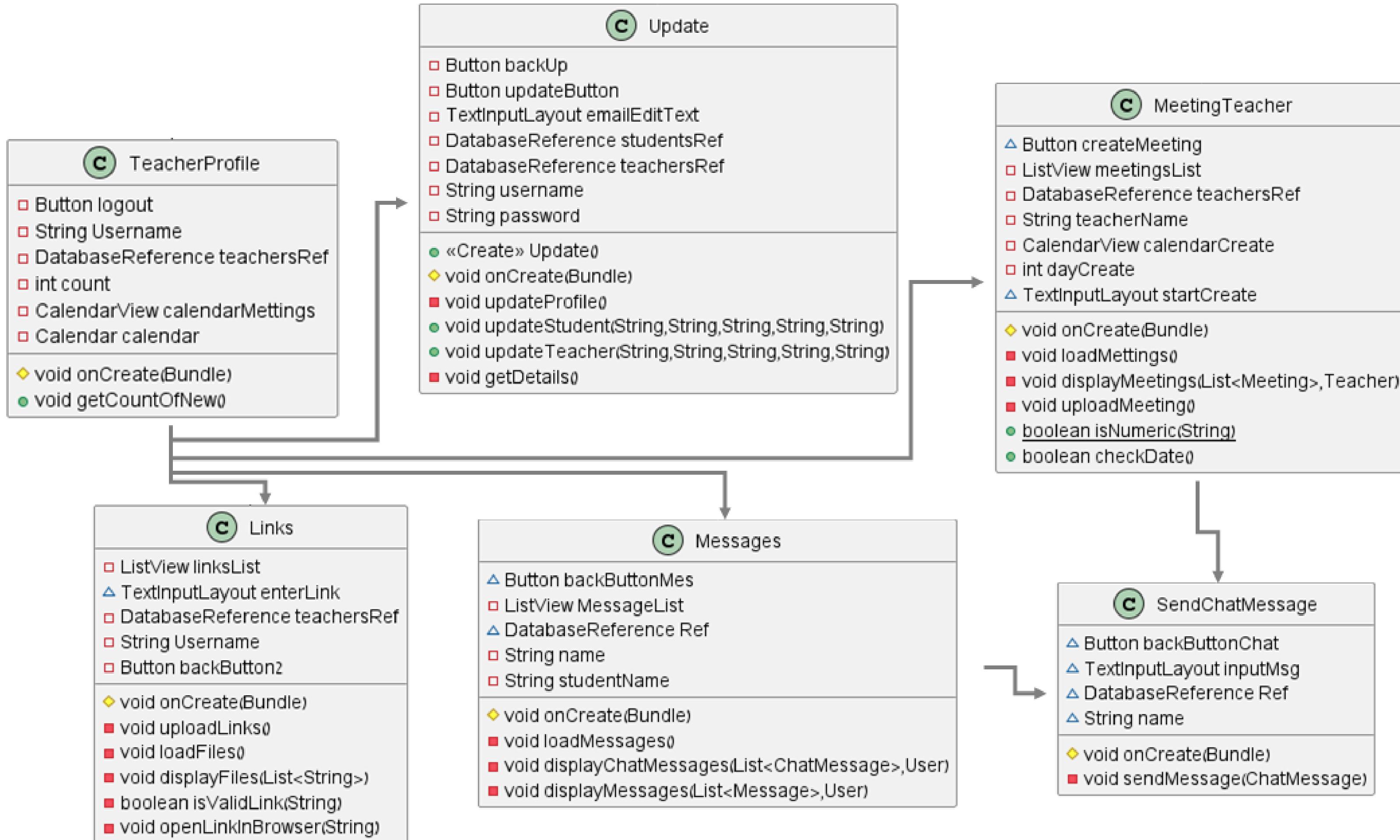
# First Diagram

# Use Case Diagram:



# Class diagram





# **Differences between design and implementation**

**In the original planning we planned that the students will be able to pay with credit card after the lesson is over.**

**We wanted to make the app work like a feed that the teachers post statuses and it will appear on a feed screen  
students can browse the feed and choose teachers based on their interest in their posts.**

# System Architecture

## MVVM Model View ViewModel



**Model:** Firebase serves as the Model. It represents the data storage, responsible for storing and retrieving information about teachers, students, scheduled lessons, and any other relevant data efficiently and safely.

**View:** Responsible for presenting the graphical user interface elements to the users. Includes components such as buttons, images, text fields, meeting schedules, and links, providing an intuitive and visually appealing user experience.

**ViewModel:** intermediary component – Handles user operations, such as scheduling lessons, updating profiles, and managing meetings, by interacting with the Model and updating the View accordingly. Responsible for the flow of data between the View and the Model.

# System Architecture



## 1. Front-End:

The front-end of the application is downloaded and installed on the user's device. provides an intuitive interface for users to interact with the application and access its features.

### **STUDENT:**

- 1. Student Account:** Securely log in & can view and update their profiles.
- 2. Teacher Search & Scheduling:** Search by preferences , schedule lessons,
- 3 .Meetings Management:** View and manage meeting schedules.
- 4. Messaging System:** Send and receive messages with teachers.

### **TEACHER:**

- 1. Teacher Account:** can log in &update personal details and links.
- 2. Meeting Management:** Teachers can manage meeting schedules, check availability, and publish time windows when they are available for teaching.
- 3. Messaging System:** Teachers can send and manage messages with students.

# System Architecture



## 1. Back-End (Firebase):

- Firebase serves as the back-end for the application, providing server-side logic and real-time database functionality.
- It manages user authentication, stores users information, lesson schedules, users messages and other relevant data.
- Firebase's real-time database allows for efficient communication between the front-end and back-end, ensuring that data is updated in real-time.

**Thank you  
very much!**