



App Storyboard

Nofar Sitalkol Eichler & Sharona T. Levy

Technical Overview

The system will consist of four main components:

- **Assistance in understanding instructions:** The core of the application that reads the student's question and provides an appropriate response.
- **Application Entry:** Student registration and instructions on how to use the application.
- **Teacher Management and Monitoring Tool:** A tool for teachers to track and manage student progress.
- **Log Extraction Tool:** Partial logging of student interactions with the application will enable analysis of usage patterns. If the dataset is too large, the project can be implemented in two phases. If only one group is used, predefined hooks should be created for future development of the logging system and teacher management system.

Devices, Operating Systems, and Language

We aim for the application to function on mobile devices and computers (tablets would be a great addition) running Android and iOS, with support for Hebrew.

AI Bot Training Data

We are currently preparing numerous examples of these tasks to train the AI bot. This includes Hebrew text inputs that students might provide and the expected responses from the system. Additionally, a preliminary study gathered assisted teaching practices used by human teachers. This information has been structured into a sequence of support actions—essentially, a toolkit of assistance strategies the AI can apply in various support scenarios. We aim for the developed application to incorporate this data.

Shortcuts

1. [Task mediation using an AI-based chatbot](#)
2. [Admin panel for teachers](#)
3. [The login screens and registration system for the application](#)

1. Task Mediation via AI Chatbot

This is the core component and the heart of the application. The idea is to enable students with special needs to receive immediate, accessible assistance in understanding instructional tasks.

1. Task Mediation via AI Chatbot

Test Mode and Practice Mode

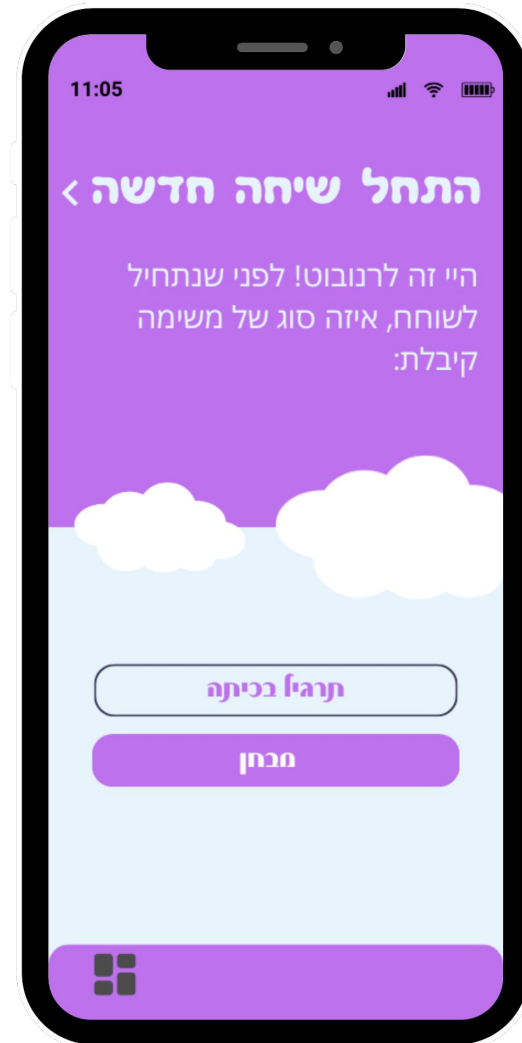
We want the application to include options for selecting between two modes: "Practice" or "Test." It is likely that we will not develop the Test Mode at this stage, but we want to be prepared for future development.

In Test Mode, the agent will provide minimal assistance. It will be able to perform tasks such as providing an example, breaking down a task into steps, and simplifying instructions—with a maximum of two repetitions (i.e., up to three examples, three levels of instruction simplification, and three levels of task breakdown).

Practice Mode will include two options, which we aim to compare:

- **Student Selection** – The student chooses the type of assistance they need using dedicated buttons (Break Down, Demonstrate, Explain).
- **Agent Selection** – A dialogue-based interaction simulating a teacher-student exchange. The agent will implement a pedagogy that we have not yet fully defined, but it will be based on professional teaching practices, balancing independent student work with guided problem-solving support.

We need to consider how a student enters Test Mode and ensure they do not switch back to .Practice Mode independently



Flowchart of the Smart Assistance Agent: Test Mode vs. Practice Mode



Focus and Prioritization in Case of Time Constraints

If time constraints do not allow for the full design of both modes, we will prioritize designing the classroom practice mode over the test mode.

The second mode (test) could be marked in a way that users understand it as a "development zone" and that it is not currently available in the application. This could be indicated by a gray lock icon and/or an informational tooltip that appears when hovering over the lock icon.

"Area under construction" display



1. Task Mediation via AI Chatbot

Functions to Integrate into the Chatbot

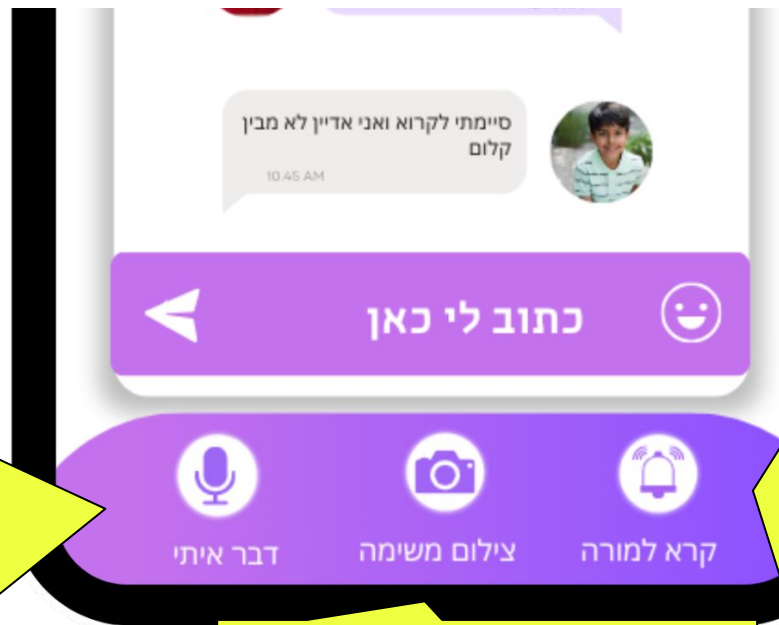
- **Task capture and upload** – Converting an image of a task into text, and if needed, translating it into English (in case there are no Hebrew-supported interfaces).
- **Call teacher button** – Pressing this button will send a notification to the teacher's mobile device, and the front indicator button will flash. Since the application is intended for multiple students, this feature should be designed so that when several students press "Call Teacher," the application organizes them in a queue, displaying the request order and timestamp.
- **"Talk to me"** – A feature for reading the verbal content aloud (to reduce cognitive overload and assist students with reading difficulties) and recording voice input (allowing students to record their questions instead of typing). This requires a speech-to-text API.
- **Agent response satisfaction** – After each agent response, a question will appear: "How satisfied are you with the agent's response? 1 - Poor, 3 - Average, 5 - Excellent," with a selection between 1-5.



1. Task Mediation via AI Chatbot

Functions to Integrate into the Chatbot

"Talk to me" – A feature for reading the verbal content aloud (to reduce cognitive overload and assist students with reading difficulties) and **recording** voice input (allowing students to record their questions instead of typing). This requires a speech-to-text API.



Taking a photo of a task and uploading it to the system

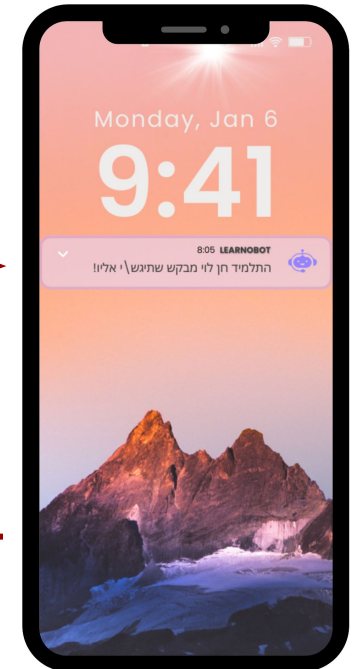
Call teacher button – Pressing this button will send a notification to the teacher's mobile device, and the front indicator button will flash. Since the application is intended for multiple students, this feature should be designed so that when several students press "Call Teacher," the application organizes them in a queue, displaying the request order and timestamp.

Flowchart of the Smart Assistance Agent: "Call Teacher" button

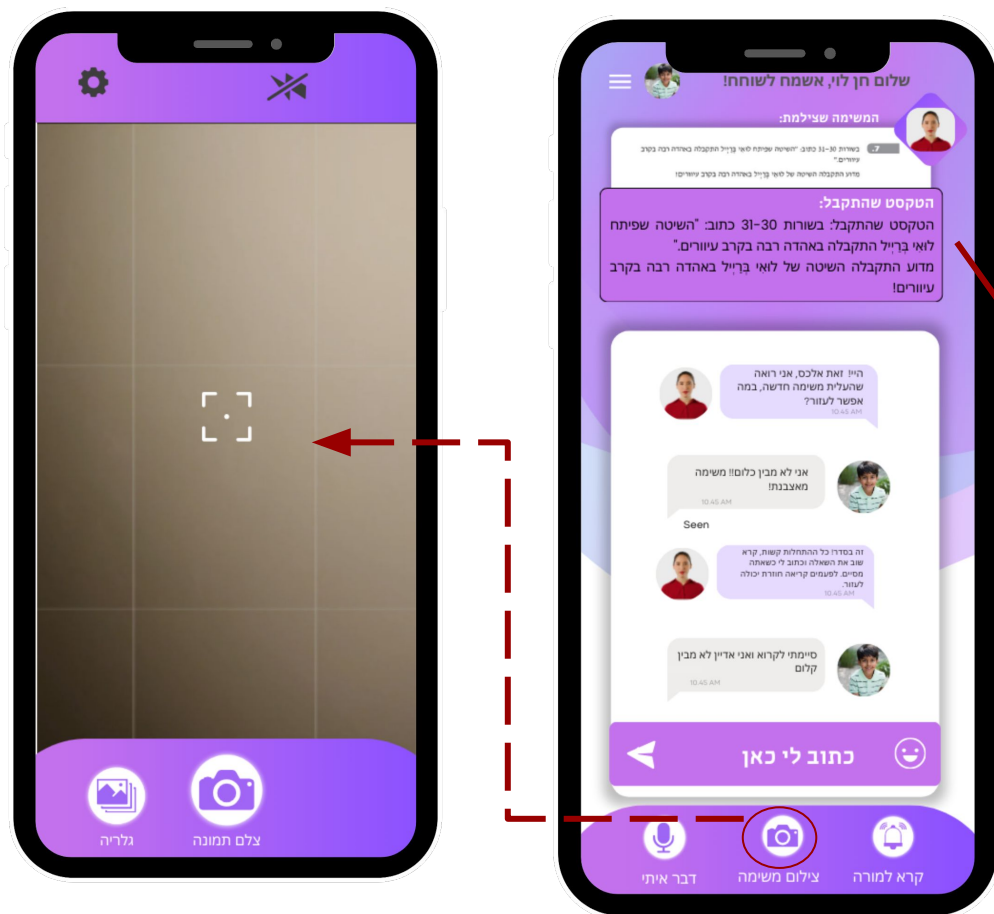


When the student clicks on "Call Teacher," a notification will be sent from LernoBot to the teacher's mobile. This is how it will look:

We would like the front indicator light to flash when receiving the alert because sometimes teachers switch to silent mode while they are working.



Flowchart of the Smart Mediation Agent: "Capture Task" Button

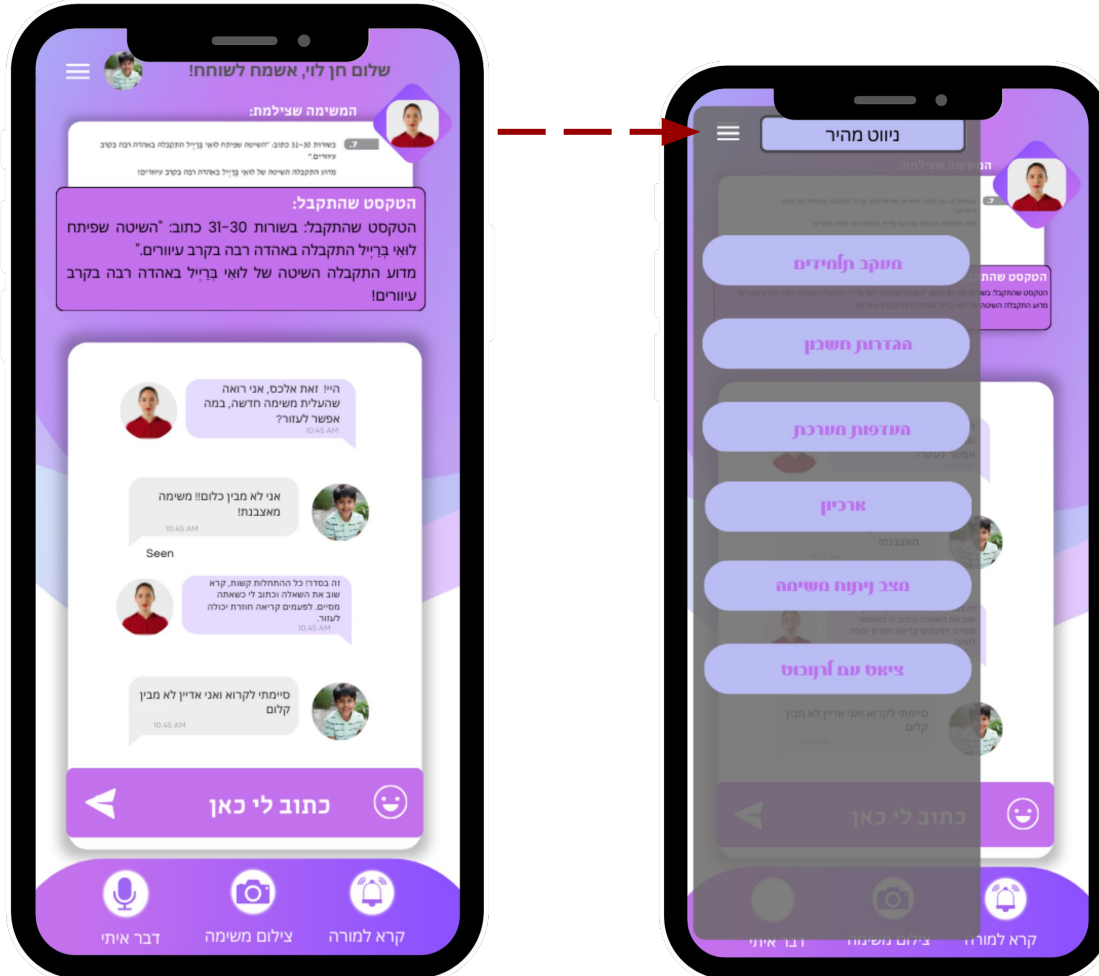


Capture the Question and Upload to the System:

The system should be able to convert an image to text (similar to what is done in Google Translate, for example), then display the image in the display window and the text in the purple box below.



Flowchart: Quick Navigation Button and the Side Screen that Opens



1. Example of Mediated Interaction with Learnobot

Here is an example of how we envision a mediated interaction with the chatbot, including the decision-making hierarchy and the use of mediation strategies, based on the results of the preliminary study and the algorithm derived from the reports of teachers who teach students with learning disabilities.

1. Task Mediation via AI Chatbot

Additional Functions to Integrate If Feasible

- **Avatar integration** – Embedding an AI-generated avatar (e.g., created with Synthesia AI) that synchronizes lip movements with the dialogue.
- **Reporting and teacher notifications** – A preliminary study identified low student motivation (e.g., lack of cooperation) as a potential barrier to AI-mediated interaction. Some teachers expressed concern about whether an AI-assisted interface could handle such situations.
 - As a possible solution, when interaction stalls (e.g., a student repeatedly responds "I don't know" or engages in unrelated conversation), the chatbot should have a defined option to call the human teacher.
 - Additionally, we want to include an option for reporting difficulties or unusual situations encountered during the mediated interaction.



2. Teacher Management Panel

Through the management panel, we want the teacher to have the option to:

- Change **system preferences** - display, sound, chatbot settings (choose between test mode and practice mode)
- Access the **"Archive"** - where chat records and information gathered from interface use will be saved.
- Modify **account settings** - update account details such as username and password change.
- **Manage and track student lists** - we want the application to include an option to create different profiles for different students, with all the student cards created by the teacher appearing on this screen. When a teacher works with a particular student, they can open that student's card and continue the conversation with the smart agent from there. This option will allow us, the researchers, to collect and organize usage data under the same user consistently.

**To ensure ease of operation and data entry into the system (e.g., when creating a new student card), we want the application design to include an option to access the management features of the "Teacher Panel" also from a computer (as mentioned in the technical introduction, which was presented above).*



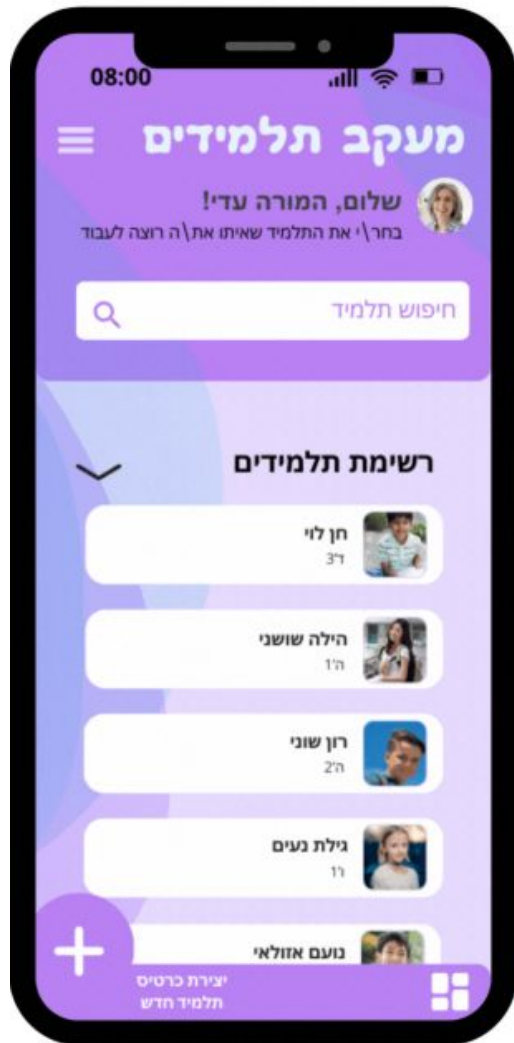
2. Teacher Management Panel



Student Tracking:

The student profiles created by the teacher will be collected and displayed on this screen.

We would like to include an option that allows the teacher to open the student profile and continue working with it each time under the same "profile." This is aimed at collecting and consolidating usage data under the same user consistently for the current study (in the future, we may also want to include the option to track student performance).



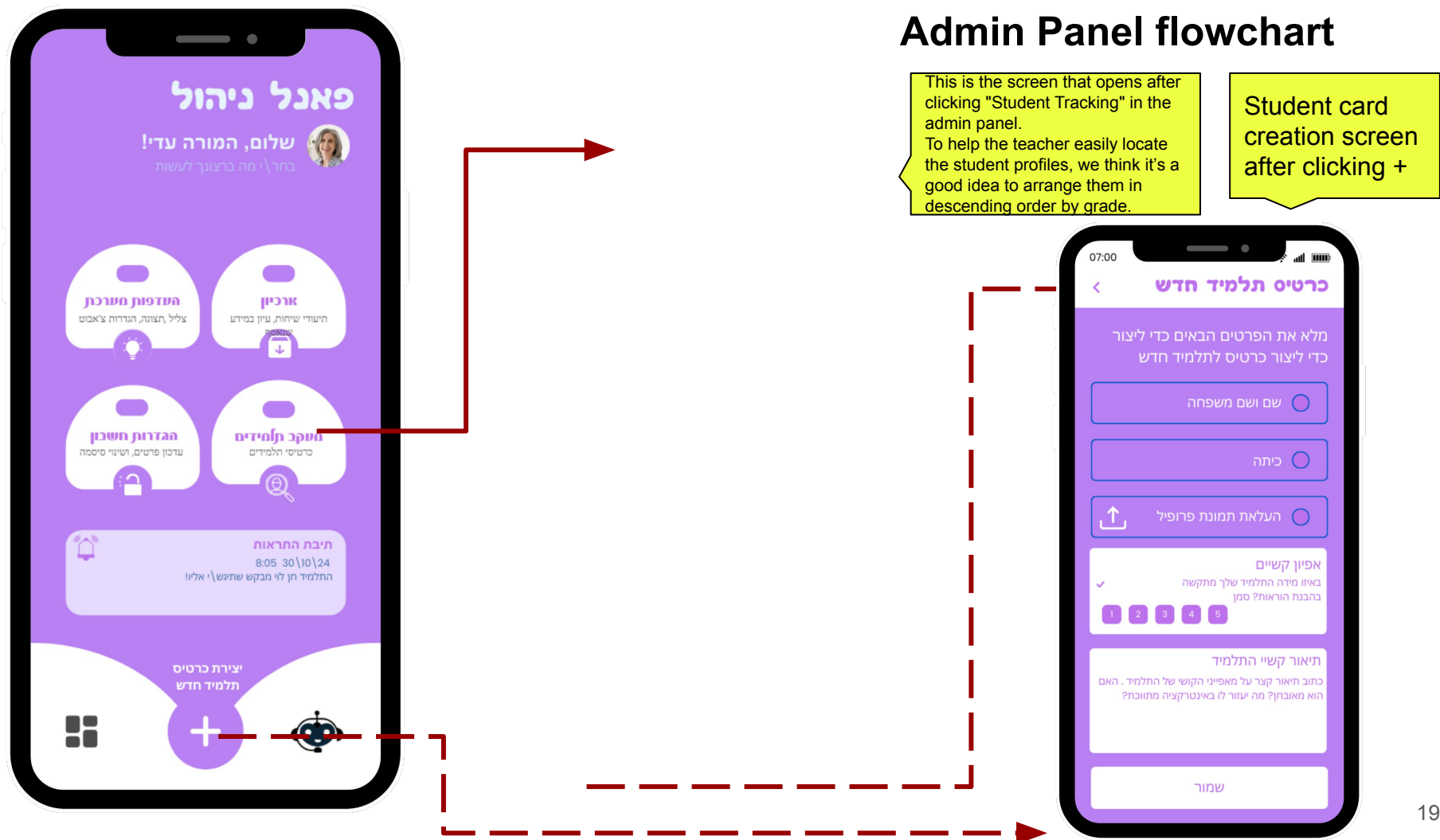
Creating a New Student Profile

The idea is to allow the creation of different profiles for individual students.

The student profile creation screen will enable the collection of background information about the student, who is a participant in the study, and store this information in an organized manner within the system. On the profile creation screen, the teacher will be prompted to fill in details such as full name, grade, rate the student's difficulty in understanding instructions, and provide a free-form description of the student's difficulties.

To help teachers easily locate student profiles, we believe it would be beneficial to design the application so that student profiles are organized by grade, in descending order.

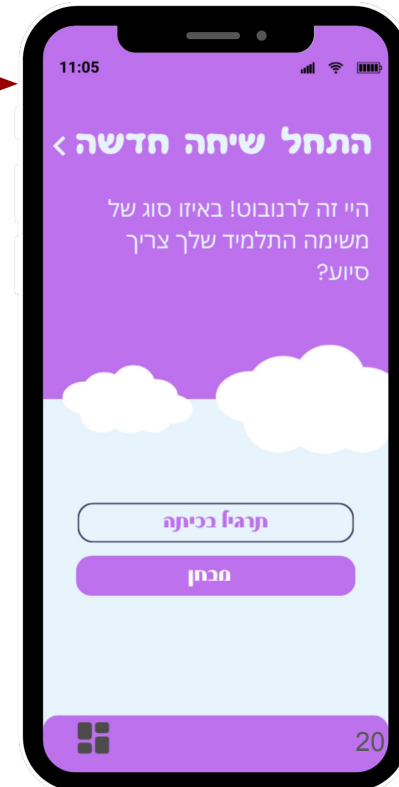
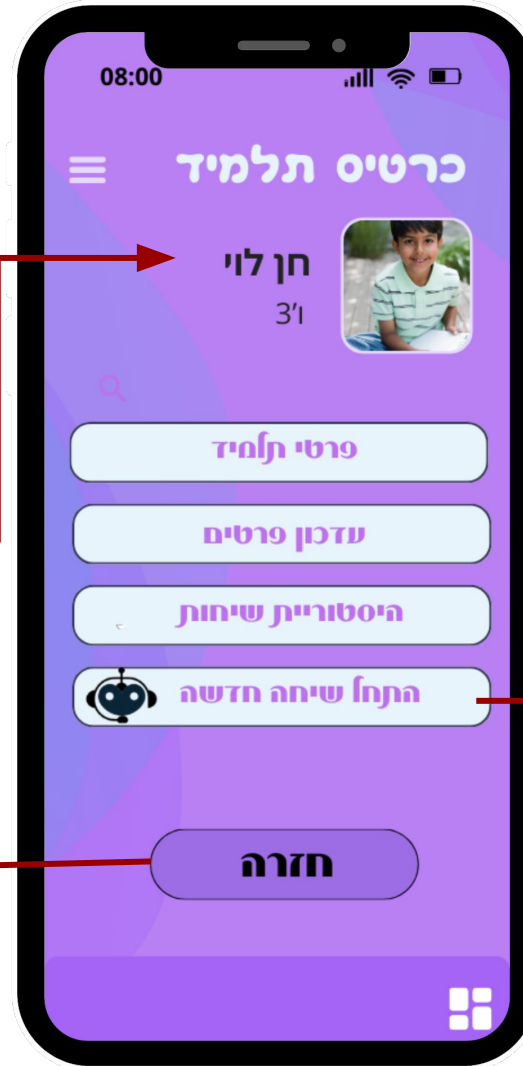
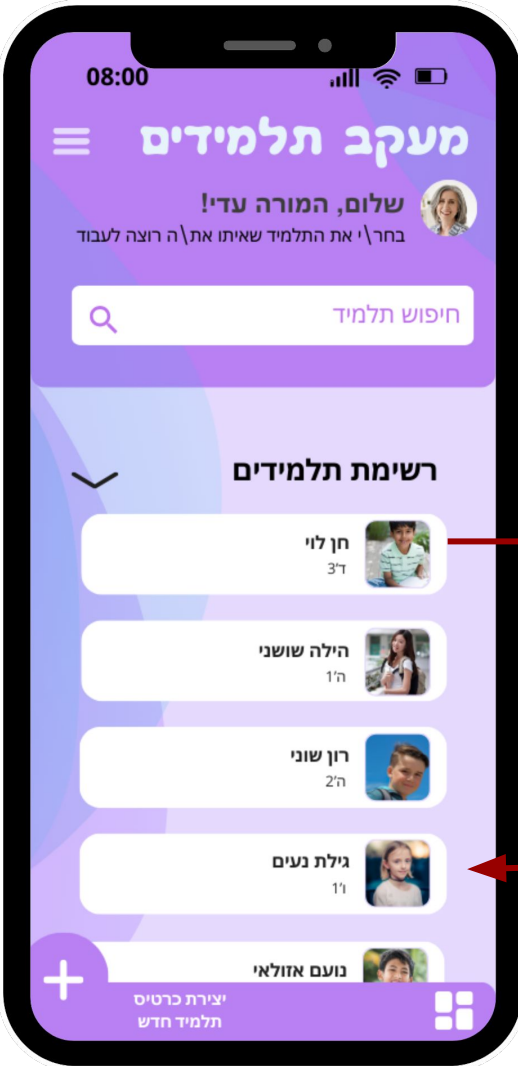
Admin Panel flowchart

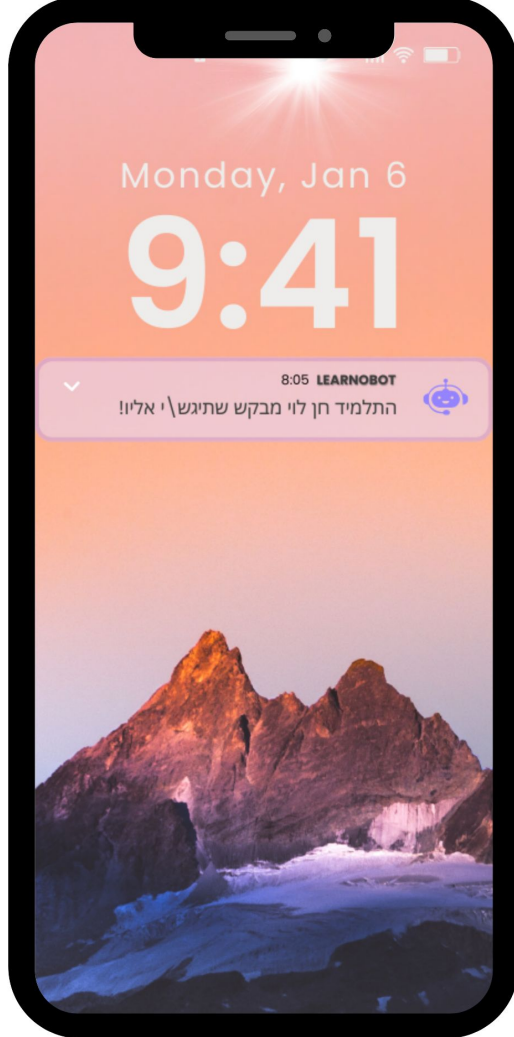


Flowchart: Student Tracking

This is the screen that opens after selecting a student from the list.

The "Start Conversation" Screen Leads to the Task Type Selection Screen

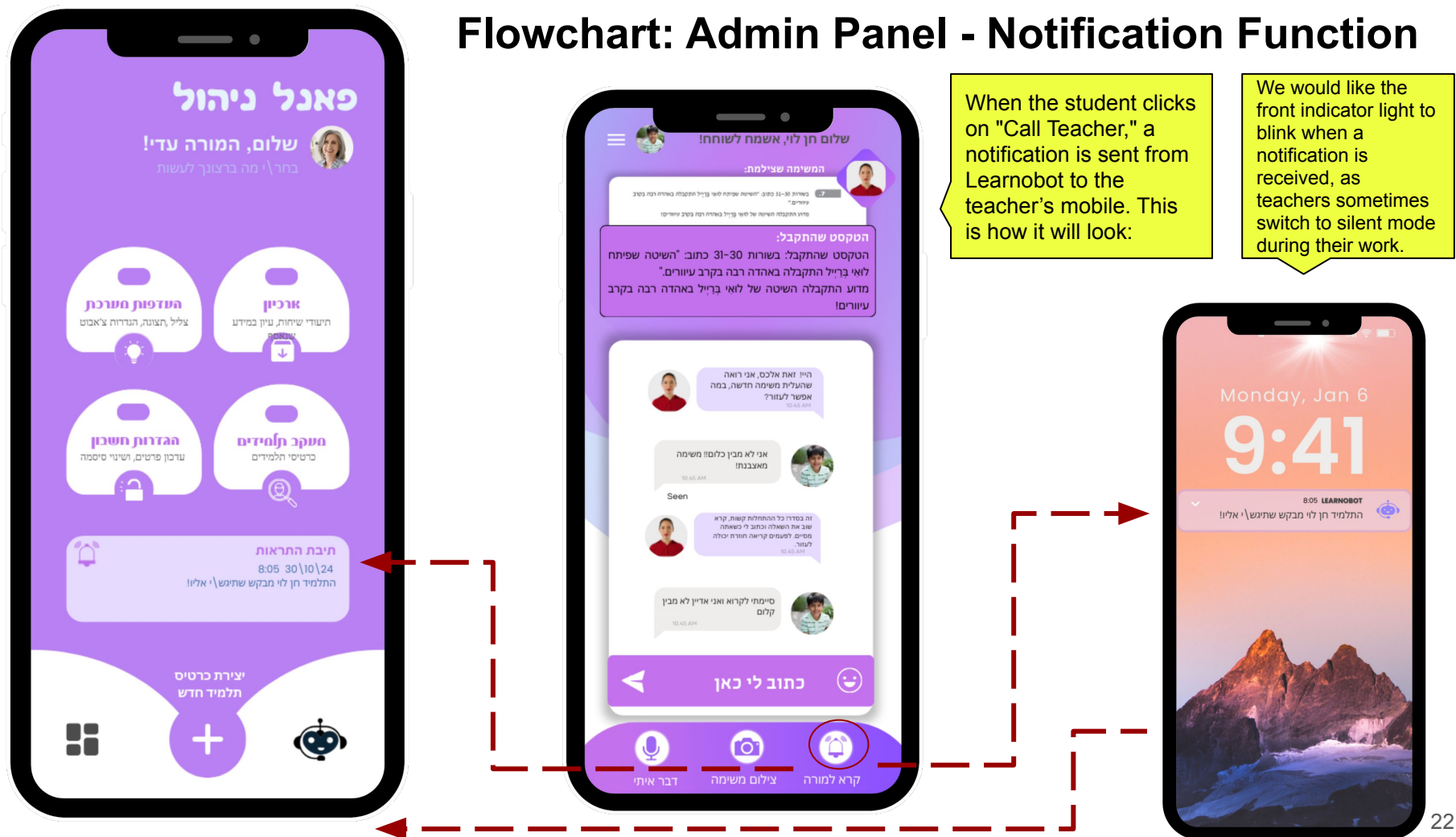




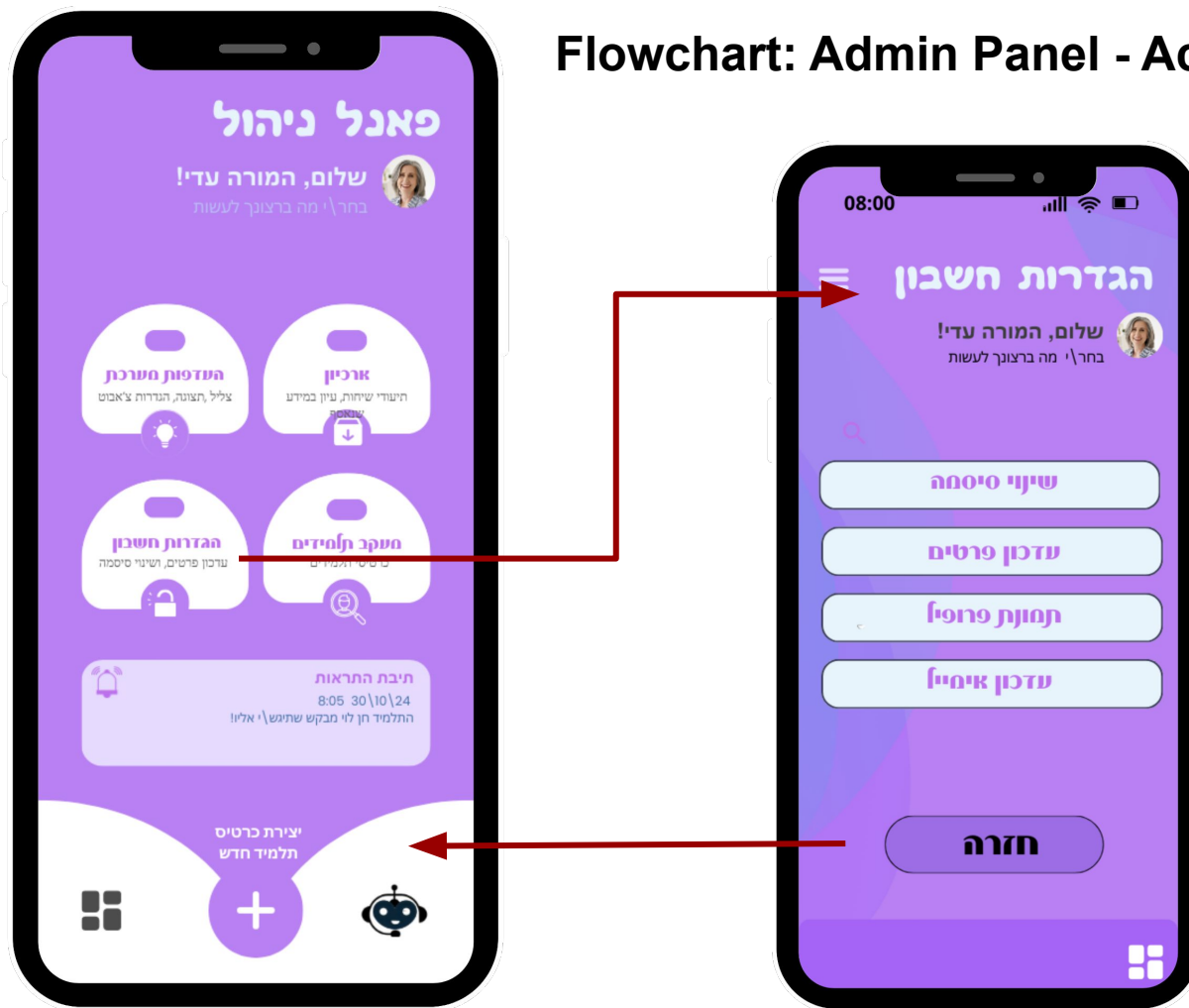
"Call Teacher" Notification Function

When the student clicks "Call Teacher," a notification is sent from Learnobot to the teacher's mobile. We want the front indicator light to blink when the notification is received because teachers sometimes switch to silent mode during their work. When multiple students click on the "Call Teacher" button, the application will organize the students in the order of their requests, which will be displayed to the teacher according to the order in which the students called.

Flowchart: Admin Panel - Notification Function



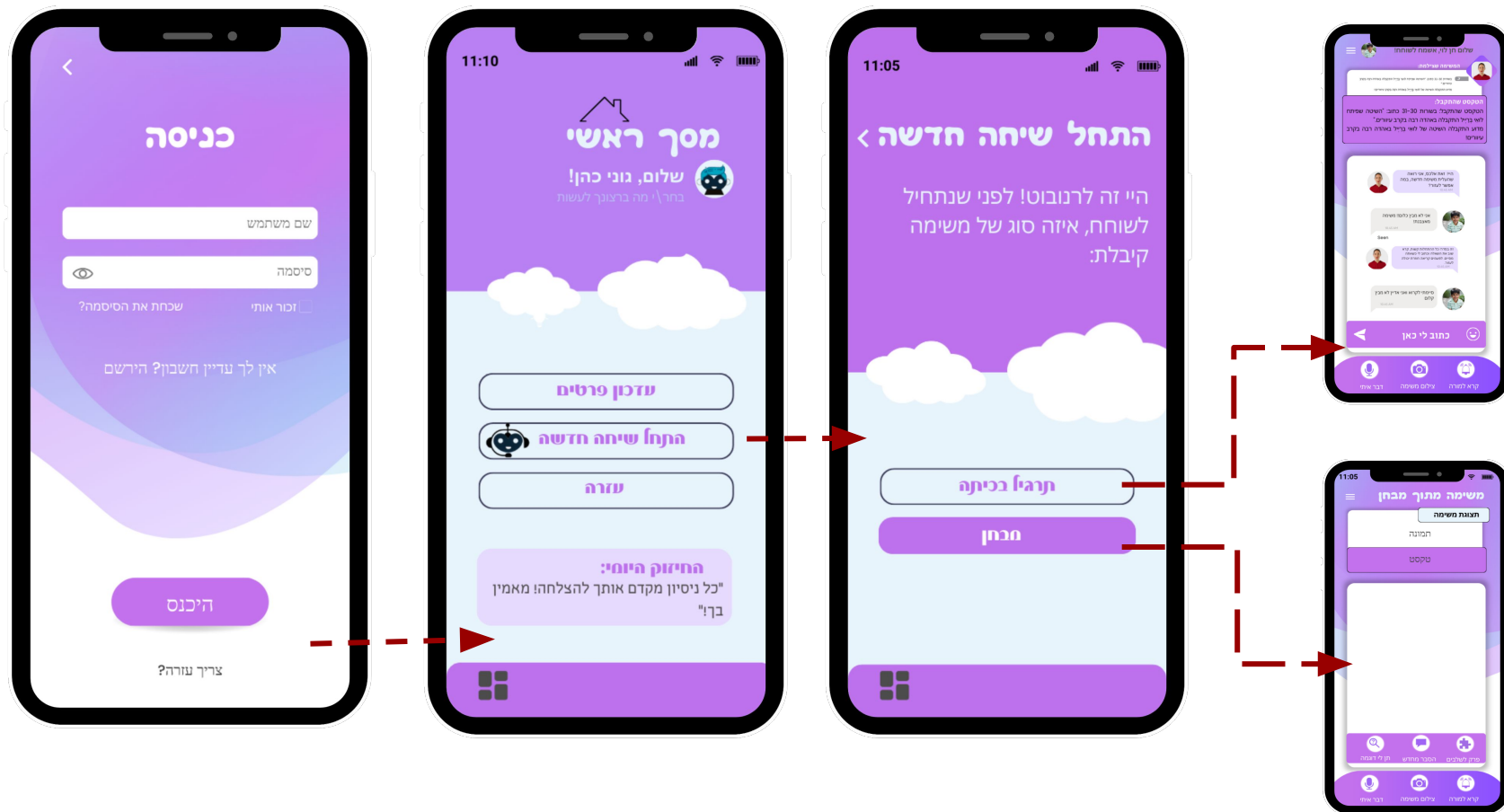
Flowchart: Admin Panel - Account Settings Screen



Flowchart: Accessing the Smart Mediation Agent Screens (from the Teacher's Environment)



Flowchart: Accessing the Smart Mediation Agent Screens (from the Student's Environment)



3. Login Screens and Registration System for the Application

To overcome ethical limitations related to participant data security, we would like the application to include a registration system for both teachers and students. The registration can be done in two modes: as a "Teacher" or as a "Student."

Choosing the "Teacher" option during registration will create an account with more advanced options than a "Student" account and will also include an "Admin Panel" screen.

Flowchart of Login and Registration Screens

An examples of the transition flow between the login screens and the registration system.

