



TED UNIVERSITY

2023 Fall Term

CMPE 491 Senior Project

Analysis Report

FocuZone

Team Members:

Şevval Kaplan - 37463197404

Hüseyin Sina Ceylan - 10003284536

Şevval Yardımcı - 55780306880

Eren Sakarya - 12521841116

Table of Contents

1. Introduction	3
2. Current System	3
3. Proposed System	3
3.1. Overview	4
3.2. Functional Requirements	4
3.3. Nonfunctional Requirements	6
3.4. Pseudo requirements	7
3.5. System models	8
3.5.1. Scenarios	8
3.5.2. Use Case Models	14
3.5.3. Object and class model	18
3.5.4. Dynamic models	19
3.5.5. User interface - navigational paths and screen mock-ups	22
4. Glossary	27
5. References	28

1. Introduction

This document contains a detailed analysis of the "FocuZone" mobile application, which is our application developed especially for ADHD patients. Our FocuZone application aims to minimize a person's distraction and increase their focus, with its four main features. These features include a to-do list feature, a study assistant, a feature that measures the sound level in the environment and tells you whether you are in an environment with a distracting sound level, and features that analyze the photo of the environment to be worked in terms of color after uploading it and detect distracting items that will recommend changing their location. In this report, we will analyze the requirements, system models, interfaces, and features of the system we developed in detail.

2. Current System

We are currently creating the general skeleton and code structure of our project, so there is no current system at the moment. However, we are trying to reduce the difficulties experienced by people with ADHD in daily life with the multi-functional features in the system we are creating for our application. Previous systems generally focus on a single feature, but our aim is to meet many needs of our users at the same time with the application we have created. While doing this, we aim to attract more users to our system by offering comfortable use. Our project in this direction is under development in sprints.

3. Proposed System

This project is designed for ADHD patients, which is very common in new generations. At the same time, people with attention deficit disorder, even if they are not diagnosed with the disease, will also benefit greatly from our application. The aim of our system is to try to prevent people with such complaints from not being able to focus on any task due to distractions in daily life, falling behind in daily life, and experiencing loss of success. We will offer this to our users through a mobile application to make it easier to access. Each of the four features in our mobile application aims to establish the working environment and daily working order in the most efficient way for a person who suffers from distraction due to ADHD. We will make choices that will minimize distraction in elements such as color choices and notification frequency in the screen designs and general system of this application. We will discuss more detailed information in the subheadings below.

3.1. Overview

The main purpose of doing this application is that we want to minimize the problems experienced by people who are diagnosed with ADHD and who have attention deficit disorder even if they are not diagnosed with ADHD because they cannot maintain their focus in their daily lives and cannot carry out many of their tasks. Thanks to this application, these people who have focus problems will be able to organize their daily lives more easily and spare time for other things in life. There are four features in our application to solve this attention problem, and all of these features help the person create his daily routine. Our first feature, the to-do list feature, will allow the person to organize what they will do that day from scratch and remember what they will do by receiving certain notifications. The second feature will be used as a work assistant. It will keep track of the person's working and rest times for the user in certain periods. Thus, we will help the person to work more focused. As the third feature, we added the sound level meter feature. This feature takes the sound level of the environment through the phone's microphone and makes dB measurements within the application, showing the user whether the environment has a sound level that may cause distraction. Our last feature is a feature that shows the ratio of distracting colors in the work area in this photo with a chart, by uploading a photo of the area to be worked on by the user before starting to work, and at the same time, identifying the items that may increase the distraction in the environment and therefore need to be changed, is made through the photo and conveys these to the user. We will offer all these features through a mobile application. All interfaces of this application are user-friendly and have been created by choosing colors that will not distract the user.

3.2. Functional Requirements

- The system must ensure that once the user has passed the promotional screens that appear when the application is first installed, these promotional screens do not appear before the user again.
- If the user has an account and the account is verified, they should be able to log in to the application with the appropriate e-mail and password. If they tries to log in with incorrect information, they should receive a warning.

- If the user does not have an account, the user can become a member of the application after filling in the information correctly in the account creation area. Afterward, they should be able to log in to their account after confirming the confirmation e-mail sent to their e-mail.
- When the user forgets his password, they must go to the relevant section and write their e-mail, and then the e-mail that will enable them to change their password must be sent to the user.
- The system should be able to keep the user's necessary information in its database.
- The system must ask the user for permission to send notifications when first entering the application's to-do list feature.
- The system should enable the user to create tasks with the name, description, and time setting specified by the user in the to-do list feature.
- When the end time of the tasks approaches, the system must send notifications to the user about the task at certain intervals.
- The system should show the user how many tasks were determined that day at the top of the to-do screen and show how many they have completed. Also, it should keep data on how many tasks have been completed.
- The system should allow the user to navigate between screens with the bottom navigation bar.
- The system should enable the user to set their own work and rest minutes in the Pomodoro feature of the application.
- The system must correctly execute the countdown indicator on the Pomodoro screen, where the user can track the working time. At the same time, it should be able to show the end of the periods with designated icons.
- The system should be able to manage the flow during the Pomodoro countdown, such as stopping it.
- When the user wants to use the sound meter level feature, the system must ask the user for permission to access the microphone.
- With the system sound meter level feature, after measuring the sound of the environment as accurately as possible, it should be able to display the result correctly in the necessary places on the screen.
- In the last feature, the system must obtain permission from the user to access the user's gallery.

- The system must send the photo sent by the user to its database.
- This photo taken from the database should be delivered to the user after operations such as object detection and color scale extraction are carried out.
- In the profile dialog, the system should be able to show the user how many tasks the user has completed in total, how many hours they have worked, and how many work sessions they have completed, by taking from the database.
- The system should enable the user to log out of their account using the logout button in the profile dialog.

3.3. Nonfunctional Requirements

Usability: It is necessary to create easy-to-use mobile application interfaces designed in accordance with general Android interface standards for users' use. The designs of these interfaces should also not be boring for users, and we need to increase the user satisfaction rate to the highest possible level. In this way, our application will be more preferred, and the number of our users will increase.

Maintainability: Problems detected with the feedback from the application should be repaired as soon as possible. The application should be updated at regular intervals and the application should be kept as up to date as possible with new technologies. In addition, certain e-mails should be accessible for technical support needs and a response should be received as soon as possible.

Performance: The directions and other library usage used in the application should be as optimal as possible so that we can make the use of the application as smooth as possible. The transition between screens should be seconds, and at the same time, unit conversions within some applications or data received in a certain period of time should be as fast as possible. The general structure of the requests sent from the application should be structured as well as possible and efforts should be made to shorten the response time. However, the system should remain available for most of the year.

Reliability: Calculations and unit conversions in operations such as sound level measurement and photo analysis made within the application should be made as error-free as possible. If any errors or miscalculations are detected, these problems must be resolved as quickly as possible. A low error rate in these parts will increase confidence in the application.

Security: User information will never be shared with third parties and will be kept encrypted. Situations such as access to the gallery within the application and microphone access will not continue without user permission and the user's privacy will be protected. All these confidentiality processes will be implemented as necessary in accordance with the law.

Scalability: Even though our application did not have many users when it was first released, we always have to consider potential users in the future. Therefore, in order for our application not to be affected by the sudden increase in the number of users using our application in the future, we need to establish a structure that can respond appropriately to this rapid increase and scale our application in this way. After establishing this structure, we will always be able to respond faster to sudden user increases and our application will not have a negative impact from such situations.

Testability: It would be very good for our application to create some test classes so that we can easily detect the errors in our application, apart from the feedback. The most practical way would be to establish automatic self-running systems for these tests for unit conversions and UIs.

3.4. Pseudo requirements

- All structures we use in database, object detection, color detection, and mobile application development, which we use in our project's features such as sound level measurement, task creation, and image analysis, must be in harmony with each other and work quickly. While doing these, we should use the latest versions and up-to-date libraries of the tools we will use as much as possible. Since the release version of our mobile application will be released on the Android operating system, we should use the most current version of Android Studio and it would be best to keep the minimum SDK of the project at 26 for now. This limitation is because we aim to make our application work on older phones as well. Java will generally be used in Android Studio, but Kotlin can also be used in some cases. In addition, the user database of our application must be constantly running and fast.

- The interfaces in our application are designed to be fluid and not distract the user, and they should be that way. In order to offer better interfaces to the user, we need to use libraries such as Lottie prepared for Android. By doing these, we need to make the interfaces of our application simple and useful.
- All data we keep in our application will be kept considering all legal situations, and we will be able to obtain the data from the user in line with the permissions we receive from the user within the application. In other words, we will not be able to access or receive any data without the user's knowledge and permission. With these, our application will have a protected and encrypted structure against external threats. Thus, it is aimed to protect our users against any external attack in a way that they will not be affected by this bad situation.

3.5. System models

3.5.1. Scenarios

Scenario 1: Introduction to the application

Description: User downloads the application, and the introduction section appears to inform the user about the application.

Steps of the scenario:

1. User downloads the application.
2. User opens the application.
3. An Introduction page appears.
4. User selects “Next” button to see the other parts of the introduction section.
5. At the end of the introduction section user can select “Get Started” button to use the application.

Scenario 2: User Register

Description: New user wants to register to the application.

Steps of the scenario:

1. User opens the application.
2. User selects the “Create New Account” button.
3. User enters name, email, and password.
4. User clicks on the “Sign up” button.

5. The system sends a verify email message to the users email account.
6. If the user verifies the email the system creates a new account and stores the user's information. Else the account is not created.

Scenario 3: User Login

Description: Existing user wants to login to the application.

Steps of the scenario:

1. User opens the application.
2. User enters their email and password.
3. User clicks on the "Login" button.
4. The system verifies the user's information.
5. If the user's email and password are compatible the user is logged in, else an error message is displayed on the screen.

Scenario 4: User Forget Password

Description: Existing user forgets their password.

Steps of the scenario:

1. User opens the application.
2. User selects the "Forget Password ?" button.
3. User enters their email.
4. User selects the "Send" button.
5. The system sends mail to the user's email.
6. User can create a new password.
7. The system updates and stores the user's new password.
8. User can login to their account by using their email and new password.

Scenario 5: User View To-do List

Description: User wants to view the task list.

Steps of the scenario:

1. User opens the application.
2. User selects the To-do feature icon at the bottom of the screen.
3. User can see the tasks that they added. Also, they can see the total count of the tasks and the count of the tasks that they have completed.

Scenario 6: To-do List Task New Creation

Description: Logged in user wants to create a new task.

Steps of the scenario:

1. User opens the application.
2. User selects the To-do feature icon at the bottom of the screen.
3. User selects the plus icon at the right bottom of the screen.
4. The “New Task” page opens.
5. User enters the task details such as Task name, description, and select time.
6. User selects the “Save Task” button.
7. The system adds the new created task to the users To-do list and updates the list.

Scenario 7: To-do List Task Deletion

Description: User wants to delete a task from the To-do list.

Steps of the scenario:

1. User opens the application.
2. User selects the To-do feature icon at the bottom of the screen.
3. Users selects the task that they want to delete from the list.
4. The “Edit Task” page opens.
5. User selects the trash icon to delete the selected task.
6. The system removes the task from the user’s To-do list.

Scenario 8: To-do List Task Editing

Description: User wants to edit a task from the To-do list.

Steps of the scenario:

1. User opens the application.
2. User selects the To-do feature icon at the bottom of the screen.
3. User selects the task that they want to edit from the list.
4. The “Edit Task” page opens.
5. User can adjust the details that they want to change.
6. User selects “Save Task” button.
7. The system updates the task details.

Scenario 9: To-do List Task Completion

Description: User wants to mark a task as completed in their To-do list.

Steps of the scenario:

1. User opens the application.
2. User selects the To-do feature icon at the bottom of the screen.
3. User selects circle button next to the task that they want to mark as completed.
4. The system marks the task as completed by crossing over the task with a line.

Scenario 10: Study Assistant Creating New Study Session

Description: User wants to create a new study session.

Steps of the scenario:

1. User opens the application.
2. User selects the Pomodoro feature icon at the bottom of the screen.
3. User can set the duration of the study by filling the first box and the breaks of the study by filling the second box. (e.g., 40 minutes study and 15 minutes break)
4. User selects the “start/pause” icon on the middle of the screen.
5. The system starts the pomodoro session and displays a countdown.

Scenario 11: Study Assistant Pause Study Session

Description: User wants to pause the study session.

Steps of the scenario:

1. User opens the application.
2. User selects the Pomodoro feature icon at the bottom of the screen.
3. User selects the “Pause” icon on the middle of the screen.
4. The system pauses the study session.

Scenario 12: Study Assistant Resume Study Session

Description: User wants to resume the study session.

Steps of the scenario:

1. User opens the application.
2. User selects the Pomodoro feature icon at the bottom of the screen.
3. User selects the “Start” icon on the middle of the screen.
4. The system resumes the study session.

Scenario 13: User Use the Sound Checker For The First Time

Description: User wants to control the sound level of the environment for the first time.

Steps of the scenario:

1. User opens the application.
2. User selects the Sound Check feature icon at the bottom of the screen.
3. The system asks for permission to access the microphone.
4. User gives the permission to use the microphone, else the system displays an error message.
5. User selects the “Microphone” icon on the screen.
6. If permission is given the system starts to measure the sound level of the environment.
7. The system displays the sound level of the environment on the screen.

Scenario 14: User Use the Sound Checker

Description: User wants to control the sound level of the environment. (not for the first time)

Steps of the scenario:

1. User opens the application.
2. User selects the Sound Check feature icon at the bottom of the screen.
3. User selects the “Microphone” icon on the screen.
4. The system starts to measure the sound level of the environment.
5. The system displays the sound level of the environment on the screen.

Scenario 15: User Use the Zone Checker For The First Time

Description: User wants to analyze the environment for productivity for the first time.

Steps of the scenario:

1. User opens the application.
2. User selects the Zone Check feature icon at the bottom of the screen.
3. The system asks for permission to access to the user’s gallery.
4. User gives the permission to access the gallery, else the system displays an error message.
5. User selects the “Select a photo” icon on the screen.
6. User selects and uploads the image of the environment that they want to analyze.
7. System analyzes the image.
8. System displays the results of the analyze on the screen.

Scenario 16: User Use the Zone Checker

Description: User wants to analyze the environment for productivity. (not for the first time)

Steps of the scenario:

1. User opens the application.
2. User selects the Zone Check feature icon at the bottom of the screen.
3. User selects the “Select a photo” icon on the screen.
4. User selects and uploads the image of the environment that they want to analyze.
5. System analyzes the image.
6. System displays the results of the analyze on the screen.

Scenario 17: User View My Profile

Description: User wants to access My Profile page.

Steps of the scenario:

1. User opens the application.
2. User selects the icon on the right top corner of the screen.
3. System opens the My Profile page.
4. User can see the total count of the tasks, total hour of the study, and total count of sessions.

Scenario 18: User Logout

Description: User wants to logout from the application.

Steps of the scenario:

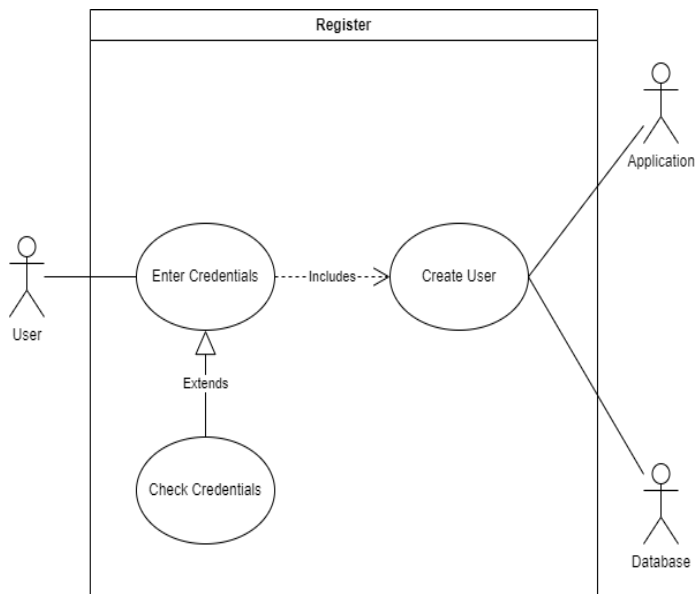
1. User opens the application.
2. User selects the icon on the right top corner of the screen.
3. System opens the My Profile page.
4. User selects the “Logout (Door)” icon.
5. System logouts the user from the application.

3.5.2. Use Case Models

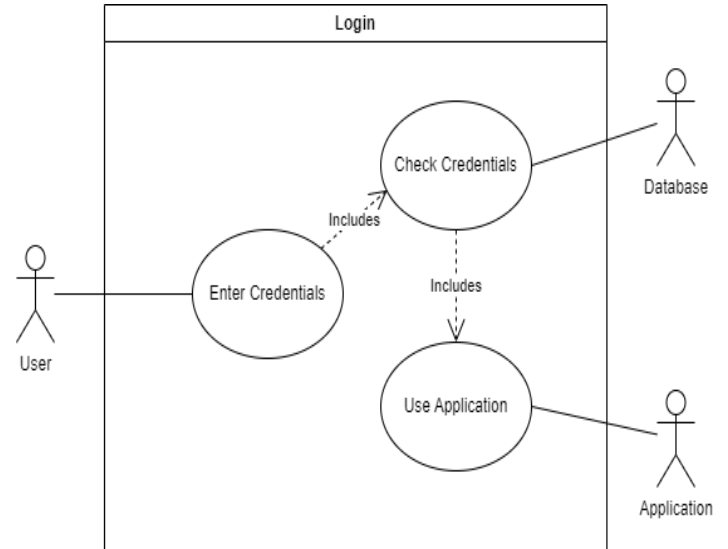
Use Case Diagram of The General Application



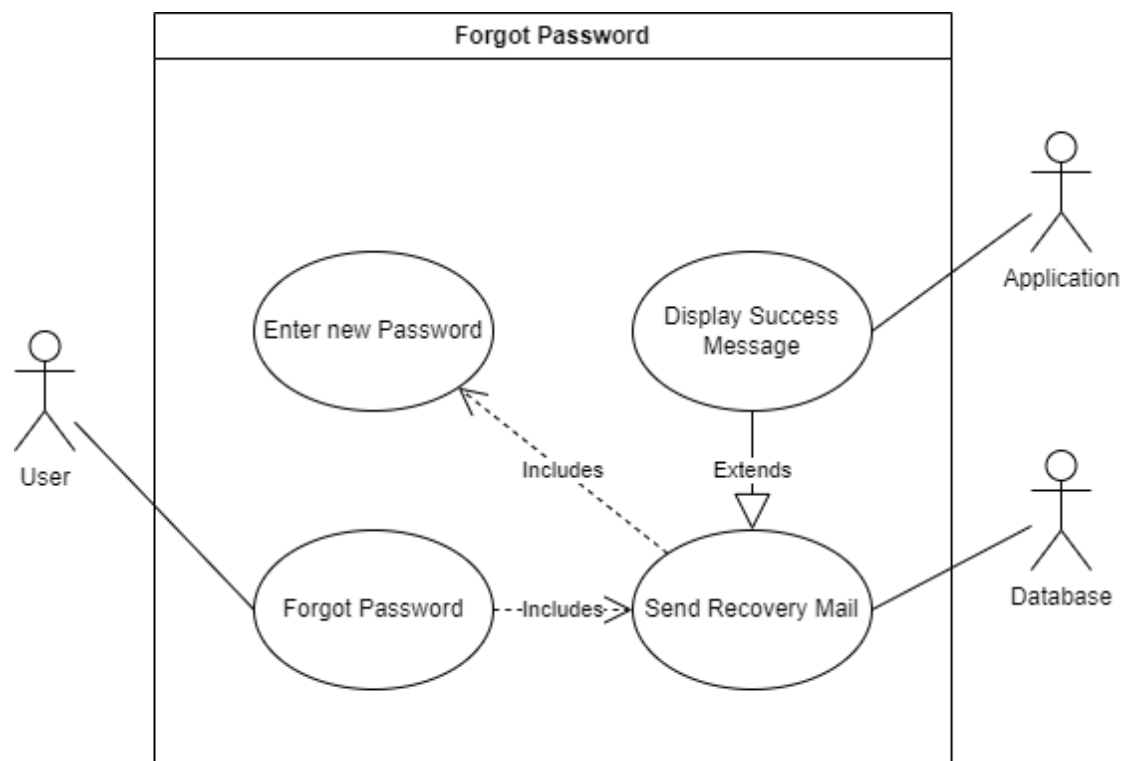
Use Case Diagram of Registration to Application



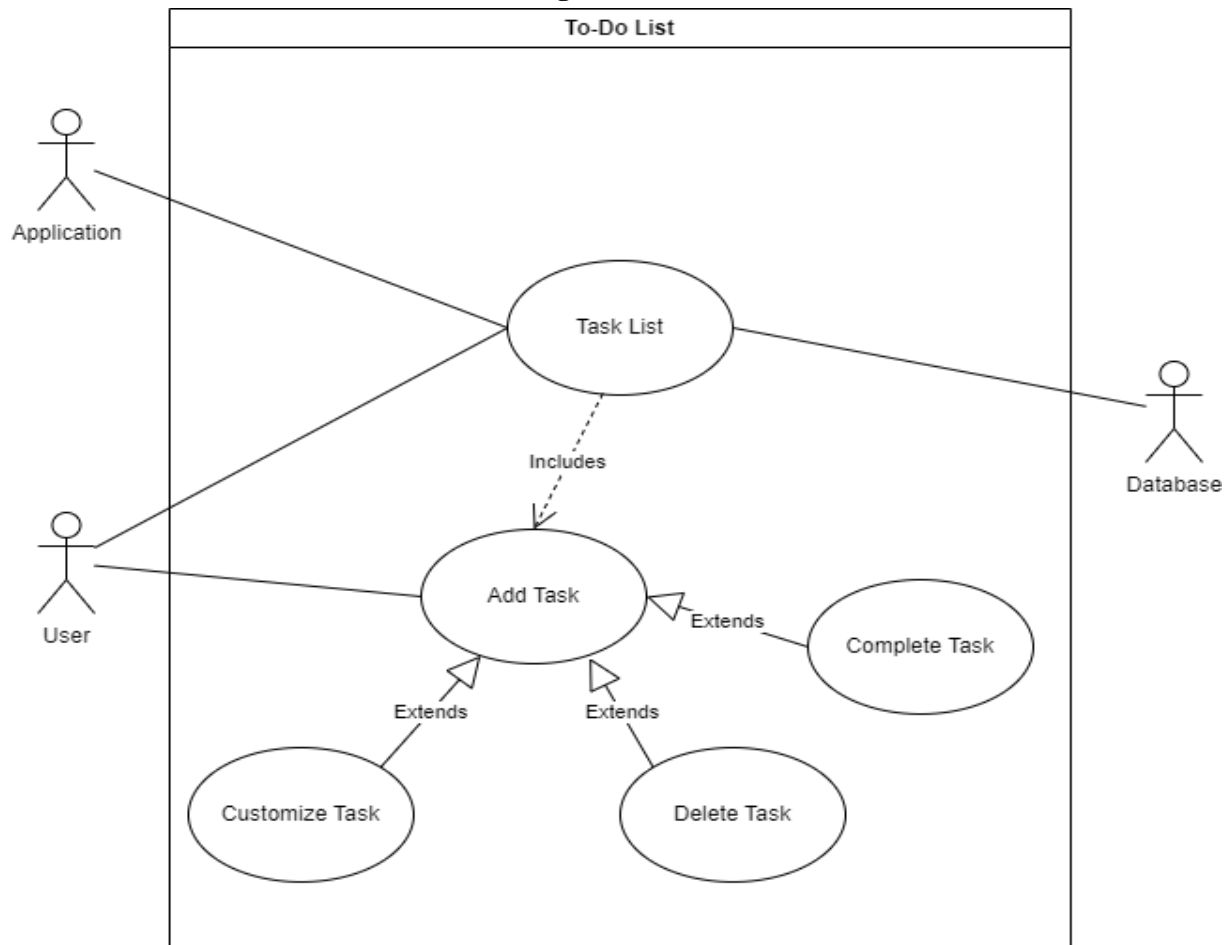
Use Case Diagram of Login to Application



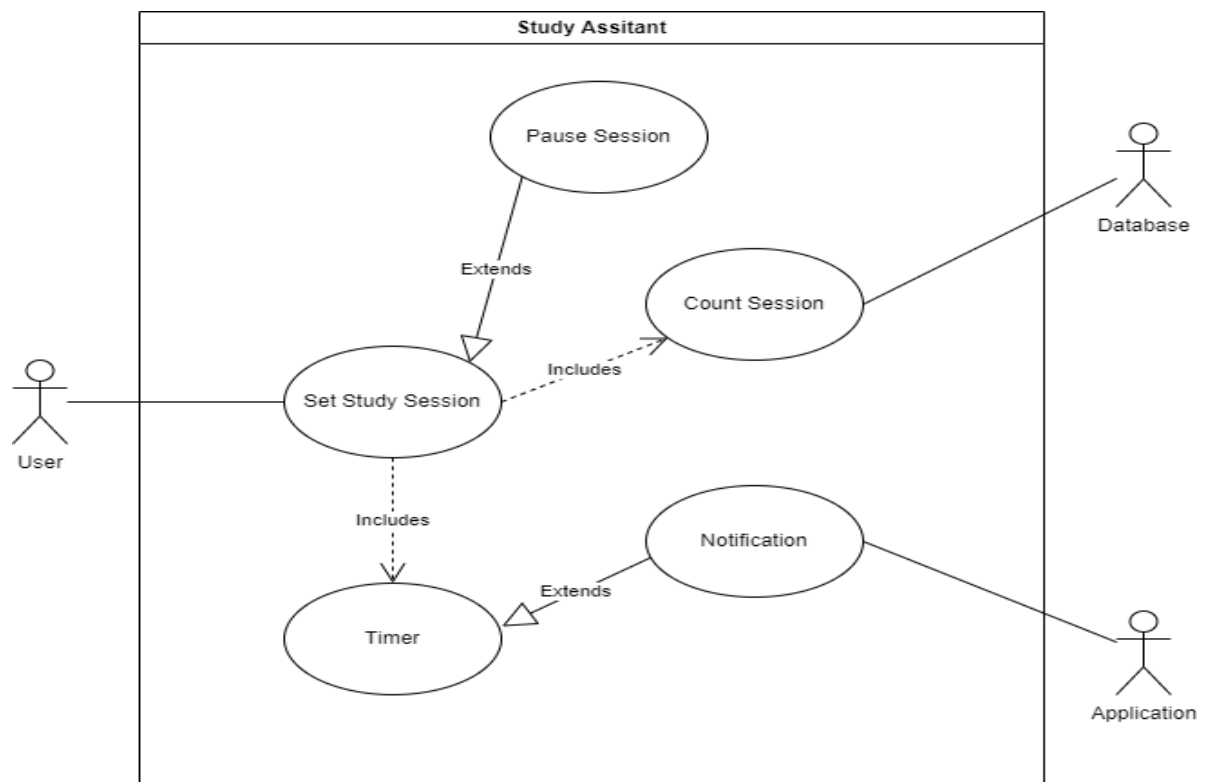
Use Case Diagram of Forgot Password Service



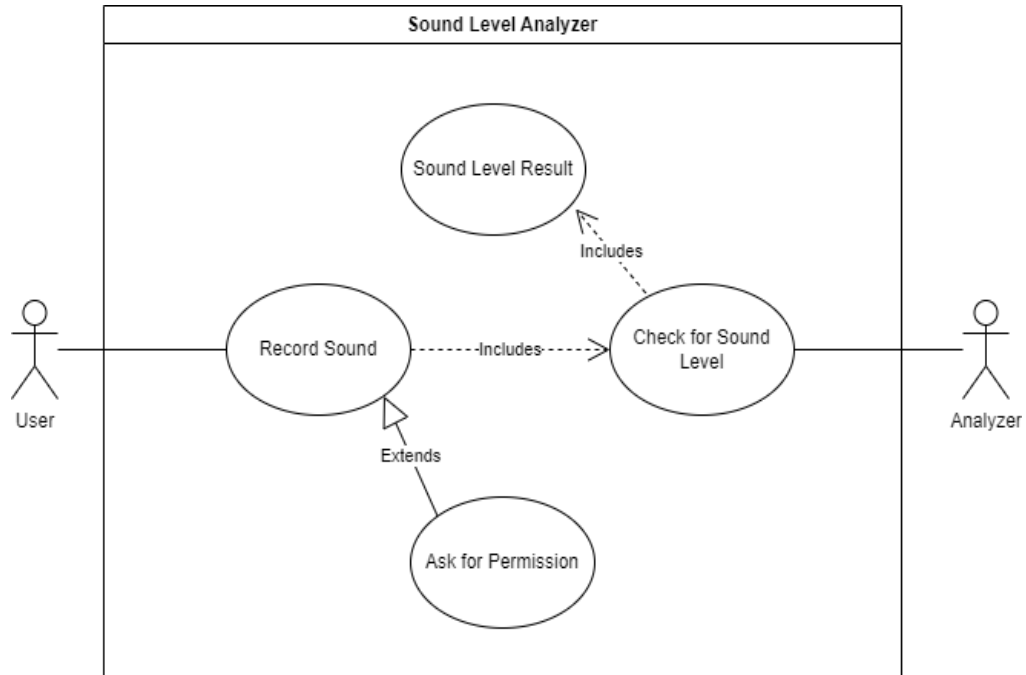
Use Case Diagram of the To-Do List



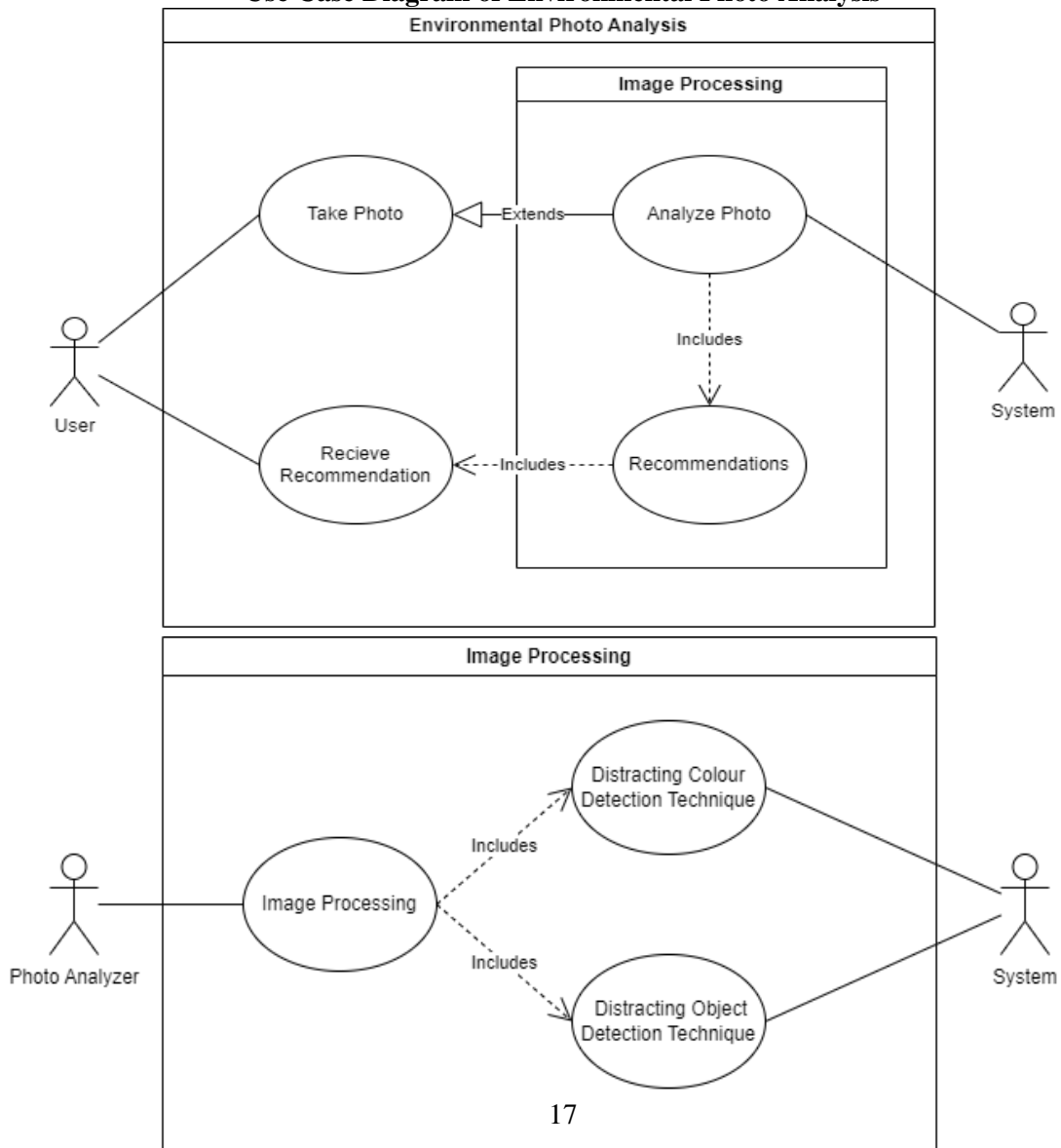
Use Case Diagram of Study Assistant



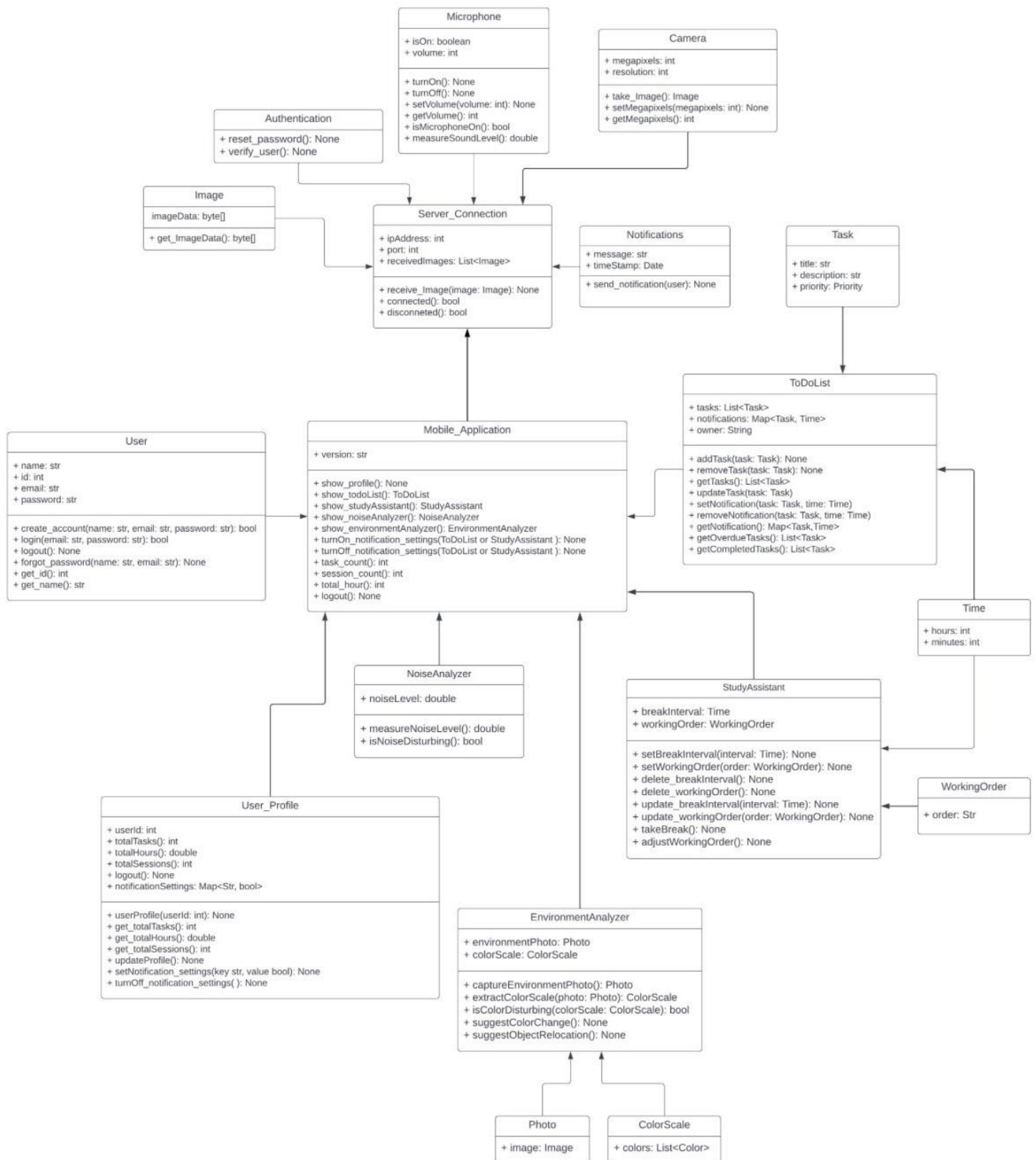
Use Case Diagram of Sound Level Analysis



Use Case Diagram of Environmental Photo Analysis

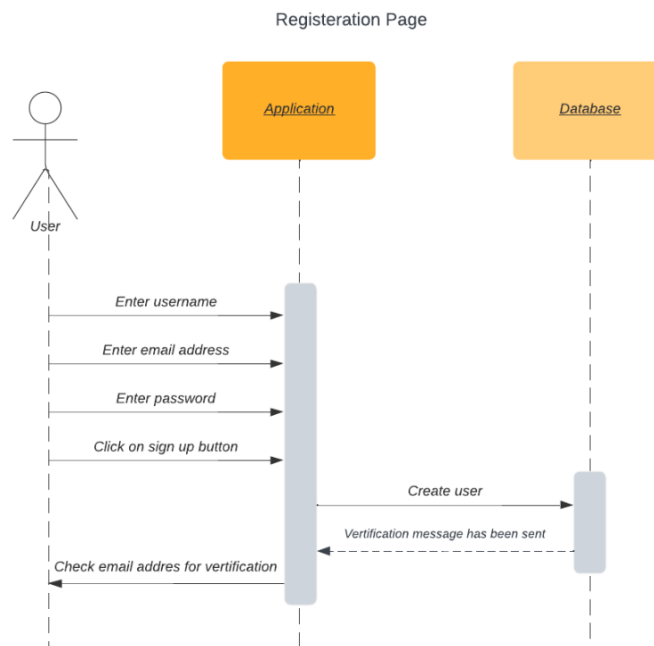


3.5.3. Object and class model

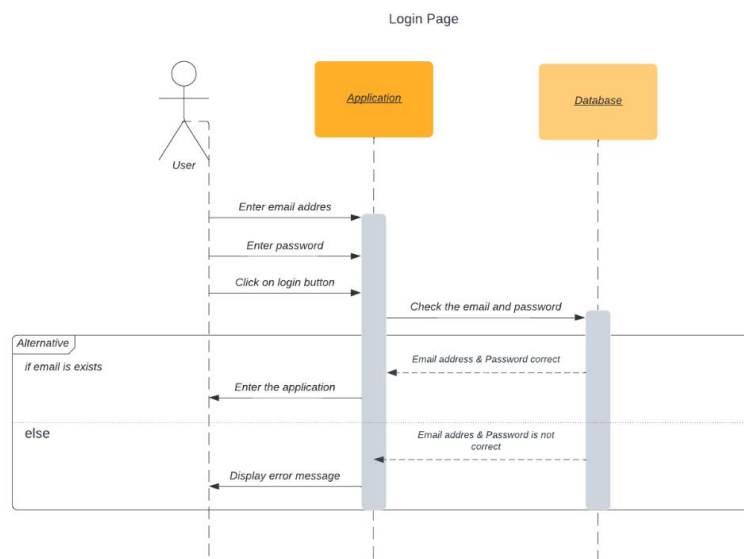


3.5.4. Dynamic models

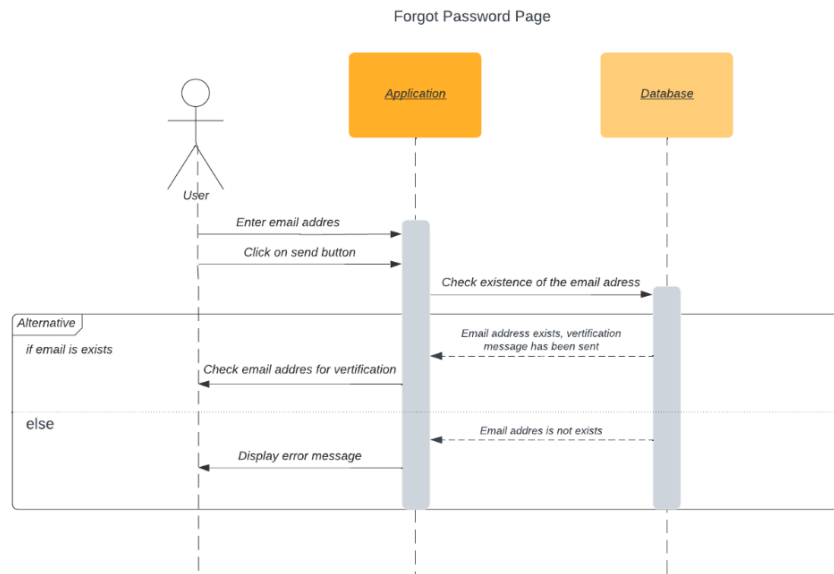
Sequence Diagram of Registration Page



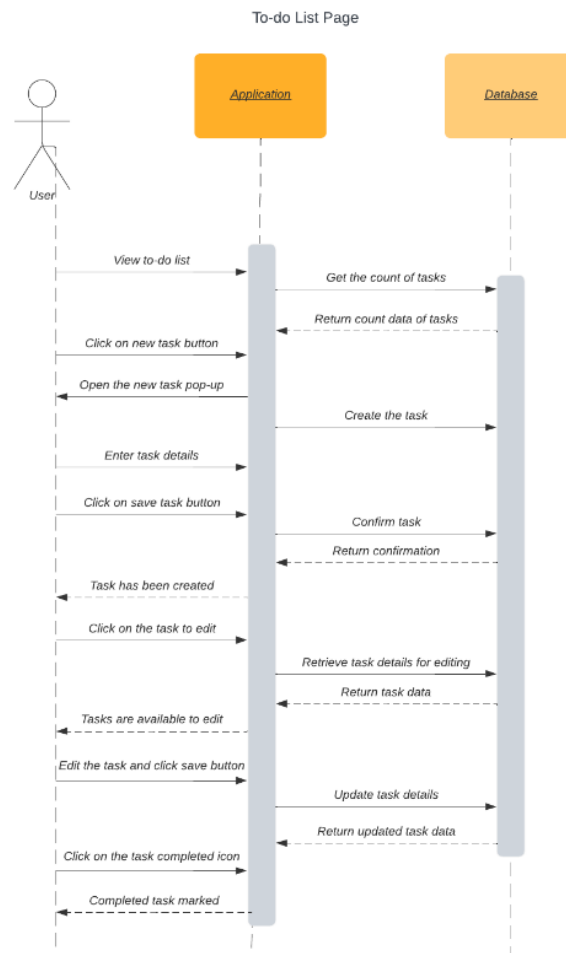
Sequence Diagram of Login Page



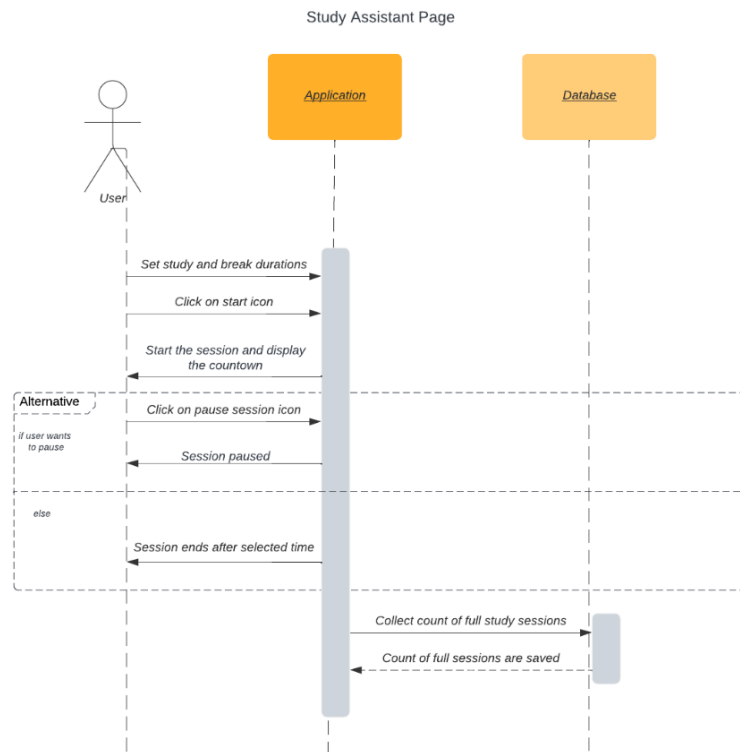
Sequence Diagram of Forgot Password Page



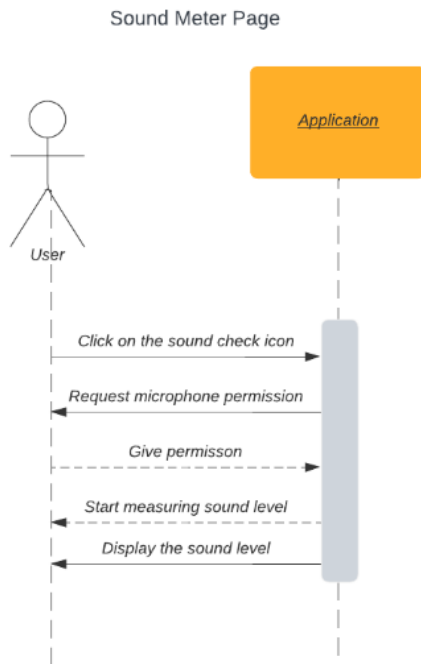
Sequence Diagram of To-Do List Page



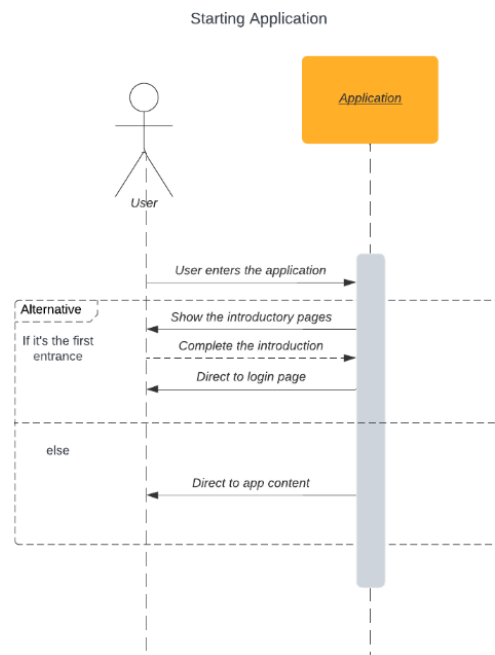
Sequence Diagram of Study Assistant Page



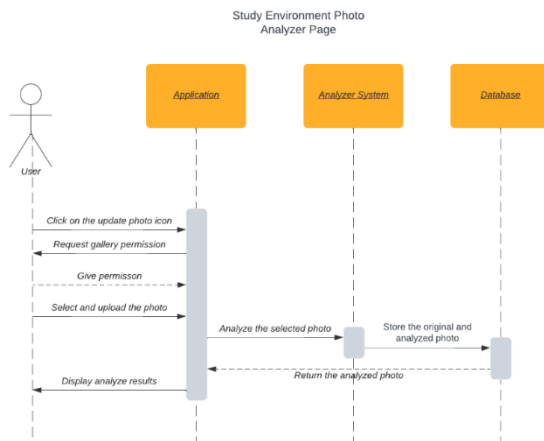
Sequence Diagram of Sound Level Analysis Page



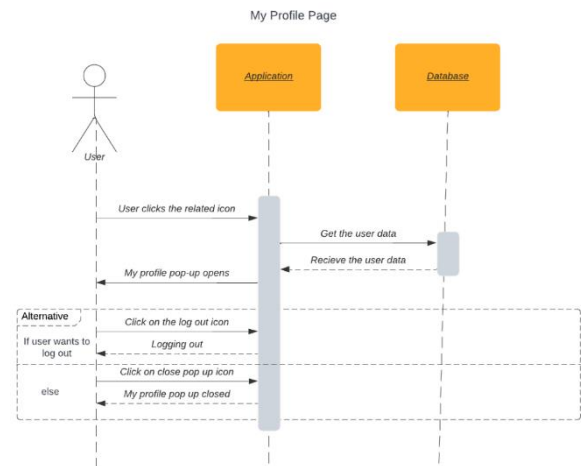
Sequence Diagram of Application Entrance



Sequence Diagram of Environmental Photo Analyzer

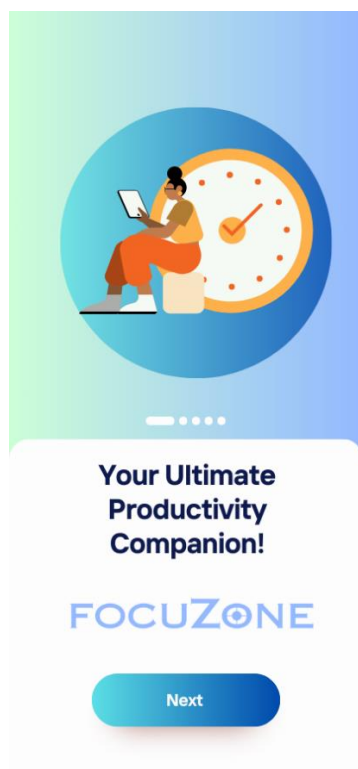


Sequence Diagram of My Profile Page

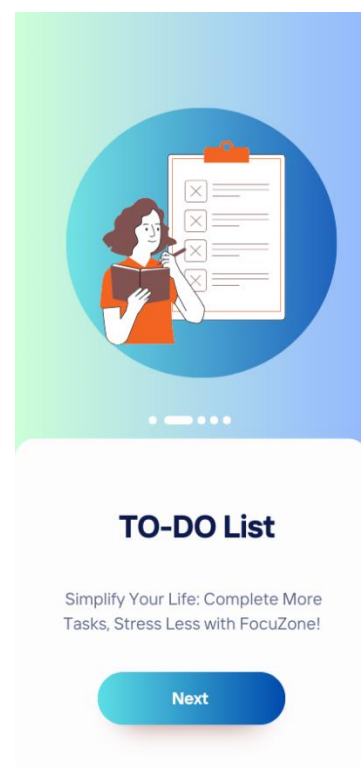


3.5.5. User interface - navigational paths and screen mock-ups

Mock-up design of intro page-1



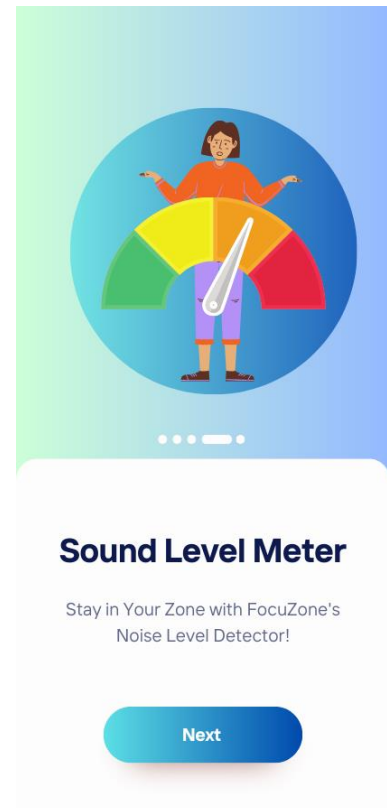
Mock-up design of intro page-2



Mock-up design of intro page-3



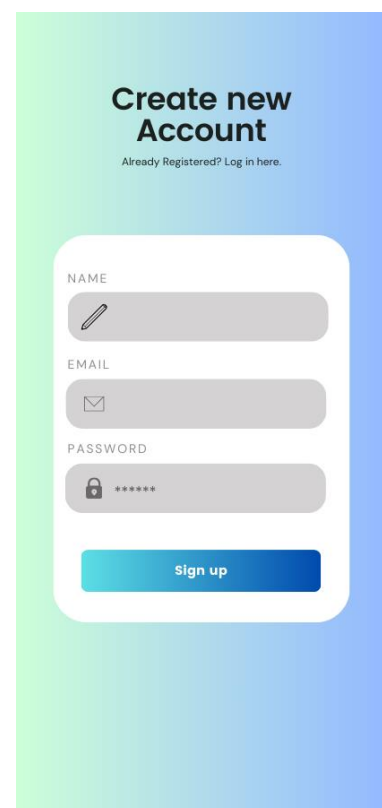
Mock-up design of intro page-4



Mock-up design of intro page-5



Registration Page



Login Page

The Login Page features a blue gradient background. At the top, the word "Login" is displayed in a large, bold, black font, with the subtitle "Sign in to continue." below it. A white rounded rectangle contains the login form. It has two input fields: "E-MAIL" with an envelope icon and "PASSWORD" with a lock icon and masked text "*****". Below these is a blue "Login" button. At the bottom of the white box, it says "Forgot Password? Signup!".

Forgot Password Page

The Forgot Password Page has a blue gradient background. The title "Forgot Password" is in a large, bold, black font, with the subtitle "New Password" below it. A white rounded rectangle contains the form. It has one input field labeled "E-MAIL" with an envelope icon. Below the input field is a blue "Send" button.

My Profile Pop-up Page

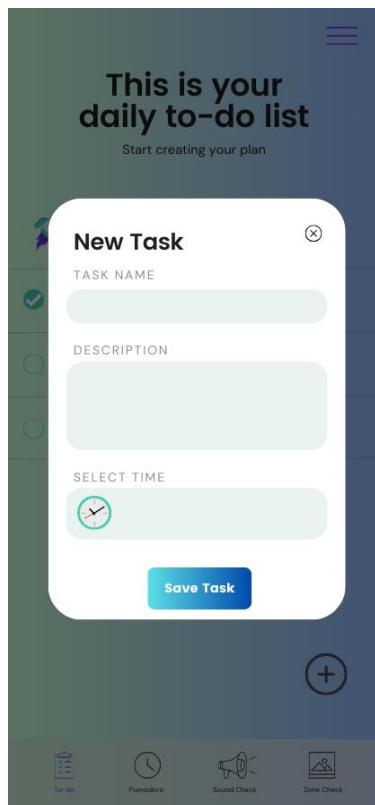
The My Profile Pop-up Page is shown as a white rounded rectangle on a dark blue background. It features a black profile icon at the top, followed by the name "Şevval Kaplan". Below the name are four colored boxes: "10 tasks" (green), "15 hr" (teal), "9 session" (blue), and a blue box with a white icon. At the bottom of the pop-up is the text "in FOCUZONE". The background page has a dark blue gradient and a bottom navigation bar with four icons: "To-do", "Pomodoro", "Sound Check", and "Zone Check".

To-Do Page

The To-Do Page has a blue gradient background. At the top, it says "This is your daily to-do list" in a large, bold, black font, with the subtitle "Start creating your plan" below it. Below this is a green box with a rocket icon and the text "1/3 task has been completed". A table lists three tasks with their completion status and times. At the bottom right is a large blue "+" button. The bottom navigation bar is the same as the previous page.

Task	Time
<input checked="" type="checkbox"/> Read 2 chapter of Les Miserables	10.00
<input type="checkbox"/> Finish math homework	12.00
<input type="checkbox"/> Study for the biology quiz	18.30

New Task Pop-up Page



This is your daily to-do list
Start creating your plan

New Task (Close icon)

TASK NAME

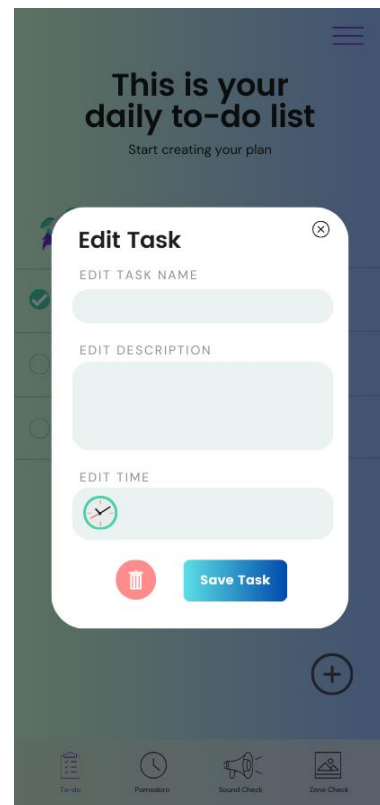
DESCRIPTION

SELECT TIME

Save Task

(Bottom navigation: To-do, Pomodoro, Sound Check, Zone Check)

Edit Task Pop-up Page



This is your daily to-do list
Start creating your plan

Edit Task (Close icon)

EDIT TASK NAME

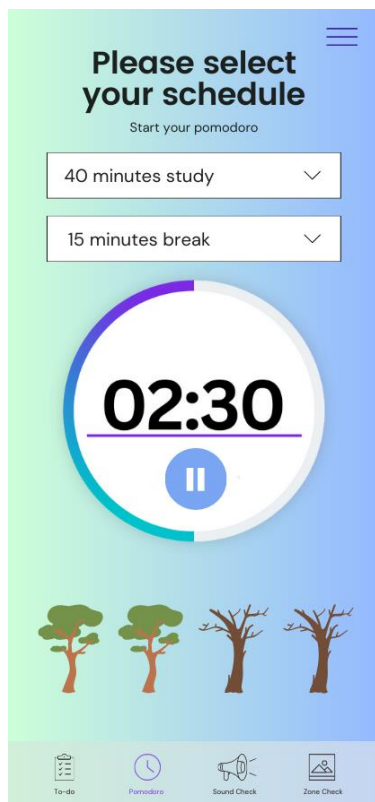
EDIT DESCRIPTION

EDIT TIME

Save Task

(Bottom navigation: To-do, Pomodoro, Sound Check, Zone Check)

Study Assistant Page



Please select your schedule
Start your pomodoro

40 minutes study (Dropdown arrow)

15 minutes break (Dropdown arrow)

02:30
[Pause icon]

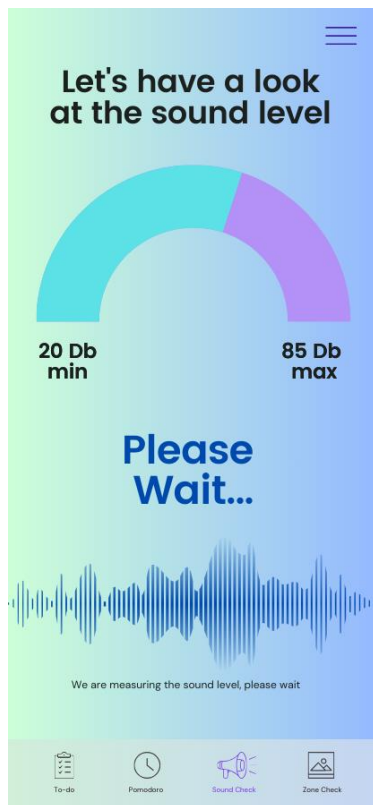
(Four tree icons representing progress)

(Bottom navigation: To-do, Pomodoro, Sound Check, Zone Check)

Sound Level Meter Page-1



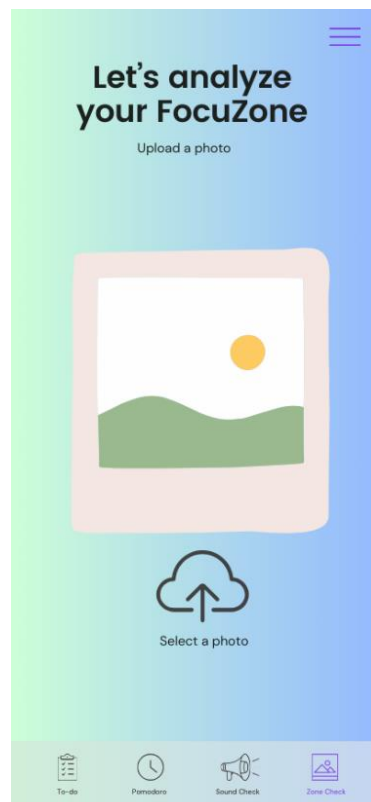
Sound Level Meter Page-2



Sound Level Meter Page-3



Study Environment Analyzer Page



4. Glossary

- **Android Studio:** It is a development environment that allows us to develop our Android mobile application with Kotlin and Java programming languages and at the same time allows us to write our codes easily with many conveniences.
- **Lottie:** It is an easy-to-use library that makes it easier for us to add some animations to our Android application and allows us to create much more effective interfaces.
- **SDK:** In general, it can be said that it is the combination of many things that enables us to develop, and the number we assigned to the minimum SDK we determined in Android Studio, which I mentioned in this report, is it determines the Android version on which it can run at the minimum level.

5. References

GeeksforGeeks. (2022b, December 2). Functional vs Non Functional Requirements.
GeeksforGeeks. <https://www.geeksforgeeks.org/functional-vs-non-functional-requirements/>

AltexSoft. (2021, July 23). Functional and non-functional requirements: Specification and Types. <https://www.altexsoft.com/blog/functional-and-non-functional-requirements-specification-and-types/>