TED UNIVERSITY

2024 Spring Semester

CMPE 492 Senior Project II

FINAL REPORT

FocuZone

Team Members:

Şevval Kaplan - 37463197404

Hüseyin Sina Ceylan - 10003284536

Şevval Yardımcı - 55780306880

Eren Sakarya – 12521841116

Table of Contents

1.Introduction

This final report will provide a comprehensive overview of the completed version of our FocuZone project. It will equip you with in-depth knowledge about every aspect of our project, including its functionalities, architecture, engineering solutions, implementation, and testing phases. By the end, you will have a thorough understanding of all the intricate details that went into bringing FocuZone to its final form.

1.1 Purpose of The System

The main purpose of our FocuZone project is to help individuals with ADHD with the difficulties they face in managing their daily tasks and maintaining their focus. ADHD can often cause difficulties in concentration, hyperactivity and disorganization in people. This can significantly affect a person's productivity in performing daily tasks and quality of life. While medication and therapy are common treatments, many people need additional support to effectively manage their symptoms in daily activities.

To provide this essential support, we developed FocuZone, an application designed not only for individuals with ADHD but also for anyone struggling with focus and productivity. The main goals of FocuZone are as follows:

1. **Create a Structured Study Routine:** FocuZone helps users create a more structured and organized study plan, making it easier to manage daily goals and tasks. The to-do list feature allows the user to keep track of the tasks they need to complete in an organized manner. On the other hand, it brings a disciplined approach to daily activities by keeping users focused and reminded of upcoming tasks with notifications.
2. **Improve Work Efficiency:** FocuZone allows users to create personalized study sessions through the Study Assistant feature, allowing users to work and take breaks according to their own periods. This personalized approach helps maintain focus and prevent burnout. Thus, the user will have more efficient working patterns.

3. **Optimize the Work Environment:** The aim is to minimize distractions with the Sound Level Meter and Work Environment Photo Analysis features. FocuZone measures ambient noise levels and analyzes the workspace for possible distractions. Thus, it helps create the optimum working environment, further increasing the user's concentration and productivity.

FocuZone application is a comprehensive support system designed to help individuals with ADHD to cope with daily challenges. Our aim is to make daily life more manageable for users by improving focus, organization and productivity. FocuZone embodies our commitment to using technology to create meaningful solutions to real-life problems faced by individuals with neurological differences.

1.2 Design Goals

We set several design goals to ensure that the system we created is reliable, user-friendly, provides accurate data, and attracts users' attention. These goals are important to ensure effective adoption of the FocuZone app.

1.2.1 Effective User Experience:

We designed an interface that is easy to use and prioritizes simplicity to ensure that the users of our application can achieve the results they want quickly and accurately. This design approach ensures that users can efficiently utilize all the features of FocuZone.

1.2.2 Privacy and Security:

Ensuring the privacy and security of our users' data is of great importance to us. Being aware that users' personal information is protected with the utmost care and confidentiality, we aim to provide peace of mind when using the application.

1.2.3 Real-Time Data Processing:

One of the important features of our application is the ability to process real-time data. We ensured that the photos uploaded by users from their gallery were analyzed in the fastest and most effective way. Likewise, with the sound level detection feature, the sound level of the environment is analyzed in real time. In this way, it is possible to provide users with the most up-to-date results and respond quickly when necessary.

1.2.4 Scalability and Universality:

One of the key features of our application is scalability and adaptability. FocuZone can easily adapt to users' needs and changing conditions. Thanks to this feature, it can work smoothly on different devices. This adaptability allows us to meet the changing needs of users and always provide the best experience to the user. FocuZone aims to make all users work more efficiently, not just individuals with ADHD.

1.2.5 Technological Infrastructure and Updatability:

Our applications technological infrastructure, supported by powerful algorithms and up-to-date libraries, is designed to continuously improve the user experience. By regularly updating technology versions, new features are added to our application and performance is increased. In this way, FocuZone always stays up to date and benefits from the latest technological innovations.

1.2.6 Data Analytics and Performance Improvement:

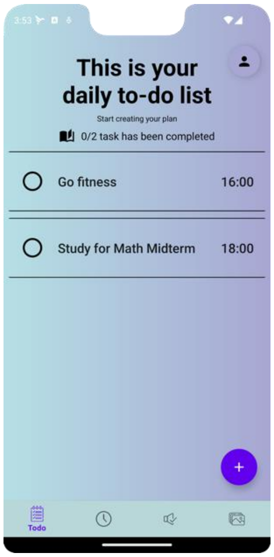
FocuZone aims to continuously optimize user experience by focusing on data analytics and performance improvement. By analyzing users' behavior and preferences, we make data-based decisions to make our application more effective. In this way, we aim to increase the performance of FocuZone and allow users to work more efficiently.

1.3 Functional Properties

Our application offers four main functional properties that provide comprehensive support for managing tasks, maintaining focus, and creating an optimal working environment. We outlined these key functionalities in detail.

1.3.1 To Do List Feature:

This feature allows users to efficiently track and organize their daily tasks and goals. Users can prioritize tasks and receive notifications as deadlines approach. When they complete a task, they can mark it as "completed" by selecting the round button on the side. This design ensures that the list of tasks done and to be done is presented to the user more efficiently. By incorporating a user-friendly interface, the To-Do List feature helps users manage their time effectively and ensures that important tasks are not overlooked, providing an intuitive and streamlined experience.



3:53

This is your daily to-do list

Start creating your plan

0/2 task has been completed

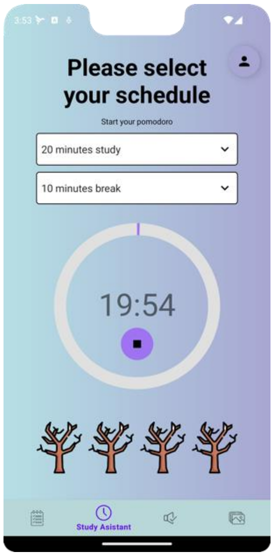
○ Go fitness 16:00

○ Study for Math Midterm 18:00

To-Do List Feature

1.3.2 Study Assistant Feature:

This feature allows users to customize study periods and breaks according to their personal preferences. The app shows the remaining time as a countdown. Thus, the user is encouraged to take regular breaks, preventing burnout and supporting balanced study sessions. The user-friendly design allows users to easily adjust their work schedule, increasing their productivity in a simple and intuitive way. Additionally, with each successfully completed study period, the trees on the screen come to life, providing a visual reward and a sense of accomplishment.



Please select your schedule

Start your pomodoro

20 minutes study

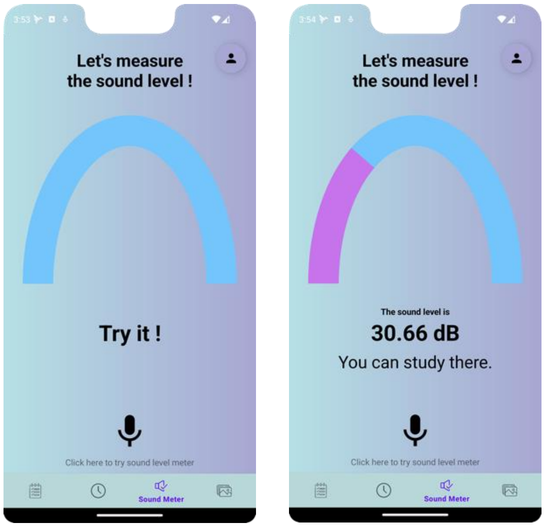
10 minutes break

19:54

Study Assistant Feature

1.3.3 Sound Level Meter Feature:

This feature provides feedback on ambient noise levels by measuring sound levels in the user's environment in real time. In environments where the sound level is high, the user is warned and told to choose a quieter working environment. With the user-friendly design, focus and productivity are maintained with minimum effort.


Let's measure the sound level !

Try it !

Click here to try sound level meter

Let's measure the sound level !

The sound level is

30.66 dB

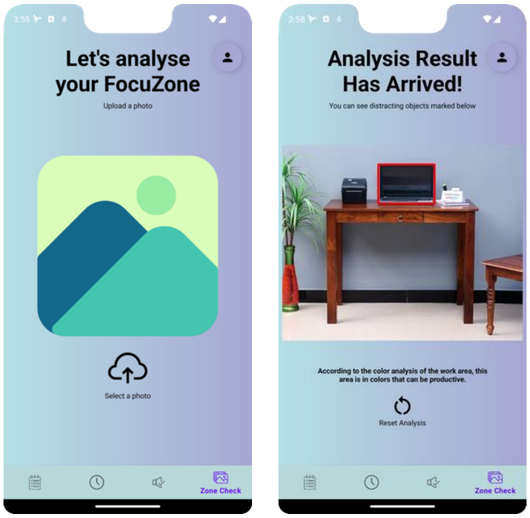
You can study there.

Click here to try sound level meter

Sound Level Meter Feature

1.3.4 Working Environment Photo Analysis Feature:

This feature allows users to analyze their work environments for efficiency. Users can select and upload a previously taken photo to the application for visual analysis. First, the application's Python-based algorithms perform a color analysis of the environment. Then, distracting objects in the environment are identified. Users receive information on whether the environment's colors are conducive to productivity and see the distracting objects marked in the photo. By providing personalized recommendations, this feature helps users create an optimal workspace that minimizes distractions, enhancing focus and productivity. The user-friendly design ensures that users can easily capture and analyze their workspace, providing a seamless and intuitive experience that supports effective work habits. The most important point for us is ensuring the reliability and accuracy of the results from the data we process. The operations have been completed with high efficiency and accuracy, delivering satisfactory outcomes for the users.



Let's analyse your FocuZone

Upload a photo

Analysis Result Has Arrived!

You can see distracting objects marked below

According to the color analysis of the work area, this area is in colors that can be productive.

Zone Check Feature

1.4 Non-functional Properties

1.4.1 Performance:

The application must provide smooth and efficient performance. Transitions between screens should occur within seconds, and data processing for features like the sound level meter and photo analysis should be quick and responsive. The system should handle high data loads without significant delays, ensuring a seamless user experience.

1.4.2 Security:

User data, including personal information and photos, must be securely stored and encrypted. Access to sensitive features such as the microphone and photo gallery should require user permission. The application must comply with legal standards for data protection, ensuring that user privacy is always maintained.

1.4.3 Scalability:

The application should be designed to handle a growing number of users and data without degradation in performance. As the user base increases, the system should scale effectively to accommodate the additional load, maintaining high performance and responsiveness.

1.4.4 Compatibility:

FocuZone must be compatible with a wide range of devices and operating systems. It should provide a consistent experience across different platforms, including various versions of Android, ensuring accessibility for all users.

1.4.5 Availability:

The application should be available and operational at all times. Downtime should be minimized, and maintenance should be scheduled during off-peak hours to reduce disruption. Users should have access to the application whenever they need it.

1.4.6 Reliability and Continuous Availability:

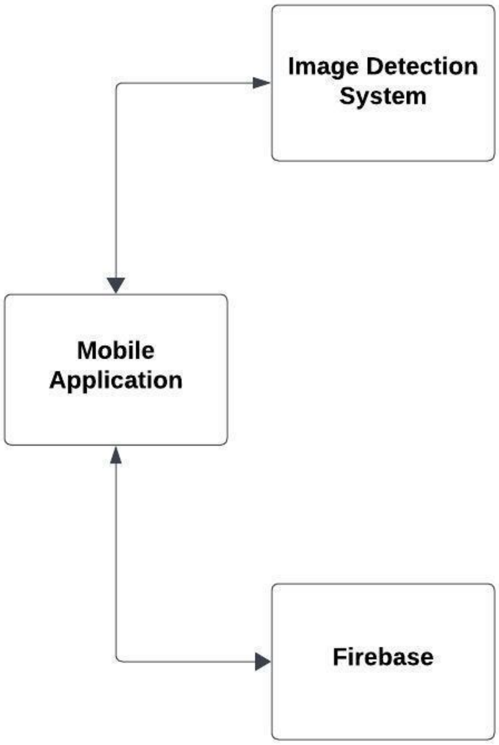
The system should be reliable, consistently performing its intended functions without failure. It should provide accurate and dependable results, particularly for critical features like task management, study assistant, and environmental analysis. Continuous availability ensures that users can rely on FocuZone for their productivity needs.

1.4.7 Data Privacy and Ethics:

The application must prioritize user data privacy and adhere to ethical standards in data handling. Personal data should never be shared with third parties without explicit user consent. The system should process and store data ethically, ensuring user trust and compliance with legal regulations.

2. Final Architecture and Design Details

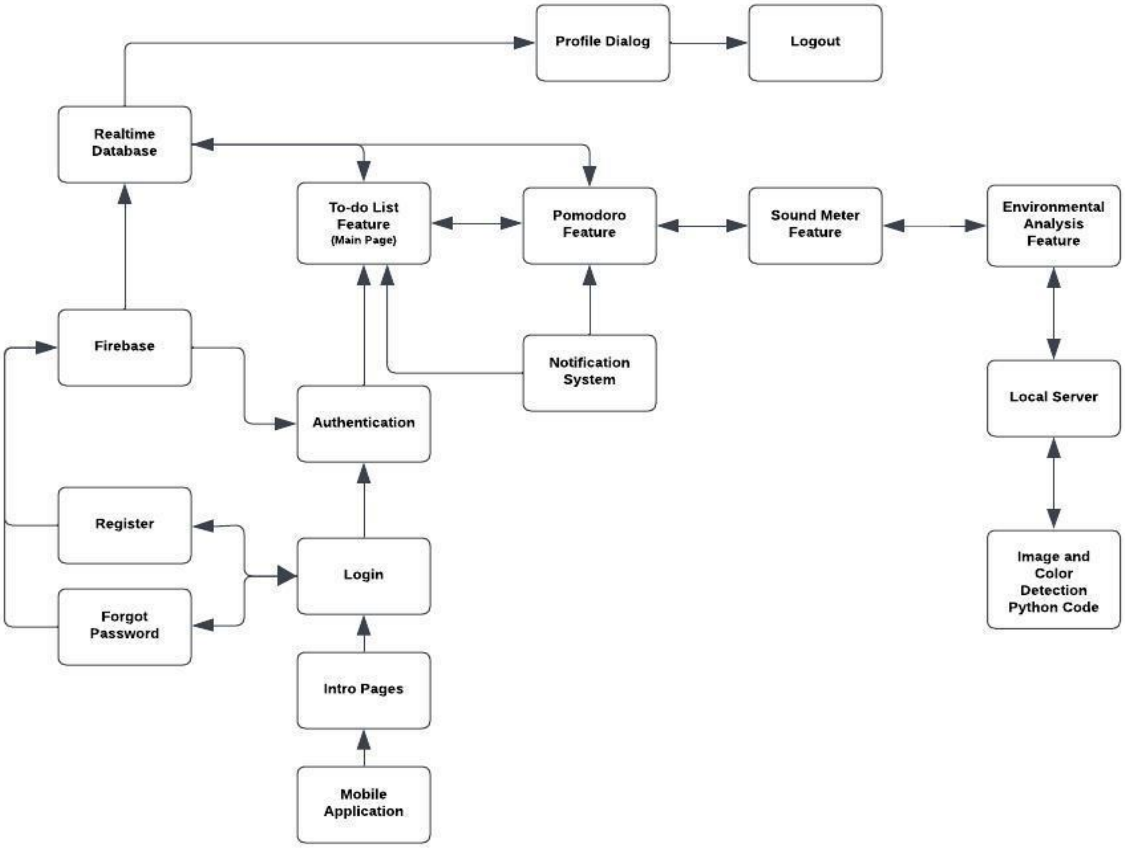
In this project, we created an application for individuals with ADHD and people who have focus problems due to distraction, so that they can do their work more focused by reducing distractions. In addition to the project being mobile application focused, the Firebase system was used for the backend parts along with the mobile application, and a local server was established with a Python code for the detection methods used for the environmental analysis feature in the application, enabling this feature to perform detection.



General Final Architecture

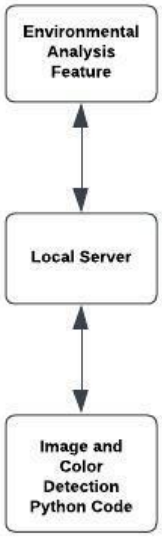
2.1 Final System Architecture

We can observe the final system architecture of our project in the chart below. Since all parts of our application are created through the mobile application, the main focus is on the mobile application part. The Firebase system is also managed with the Realtime Database system through the application's codes. Our detection system does this by creating a local server from which it can receive and send data for detection operations. All these systems work interconnectedly.



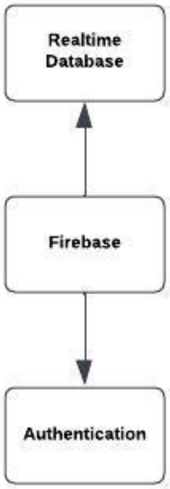
Final System Architecture

2.2 Subsystems


Environmental Analysis Feature
Local Server
Image and Color Detection Python Code

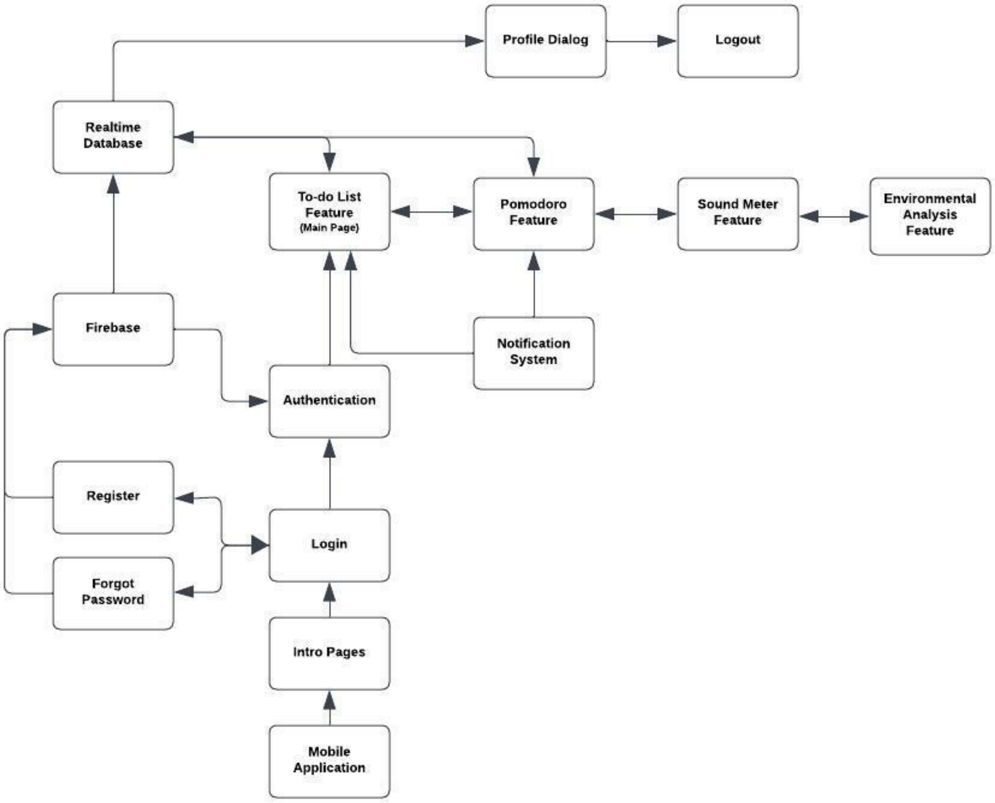
Local Server Part Flowchart

We created the image and color detection system here with Python code along with the model we trained. The structure here, together with the trained model, marks distracting objects on the photo sent from the application and sends it back to the user. We also make a color analysis of the photo coming to the system from the Environmental Analysis feature and send this information to the mobile section. We communicate here with the local server we created using Flask in the Python code.



Firebase Part Flowchart

We handled the Authentication system and database system of the application with Firebase. When you register to the application, a confirmation e-mail is sent to the user's e-mail, thanks to the Firebase system, and thus the user's account is activated. If the user has forgotten his password or wants to change it, he can do this thanks to the password change e-mail sent to his e-mail, thanks to Firebase. We also use Firebase when logging in. We also use the Firebase Realtime Database system to keep data in some parts of the application. Thanks to this system, we can easily handle our database operations.



Mobile Application Part Flowchart

The mobile application part, which is the main part of the project, has the feature of being a part of the junction of other subsystems because our project was made application-oriented and other systems are therefore designed to be integrated into the mobile application side. Here we can generally see the screen flow of the application and its connections to systems such as the notification system. The flow of our mobile application section is designed to be completely user-friendly and does not contain any confusion. After logging in to the user account, the user will come to the To-do list feature screen and will be able to easily switch between the desired features from this screen with the bottom navigation bar. In addition to being able to see some saved data through the profile dialog section on each screen, you will also be able to log out of the application.

2.3 Hardware / Software Mapping

The hardware equipment required for our project to work is quite simple. Since our project is a mobile application, the user must have an Android phone with an operating system above SDK 26. The reason why this Android phone was chosen as SDK 26 and above is that we want to be able to serve a larger number of Android devices. The presence of a camera on this Android device is required to ensure that it can effectively use the Environmental Analysis feature, which is one of the features of our application. At the same time, the reason why we need a phone on the Android operating system is that we develop our application for Android devices using Java and Kotlin languages via Android Studio.

We also set up a local server with the Python code created to run the detection system in the Environmental Analysis feature of our project. With this local server, the photo sent from the Environmental Analysis feature of our application is sent to this local server and after the necessary detection operations are carried out, it reaches the user again through this server. For now, we do this with the Flask library, which allows us to open local server operations in Python, but if a larger and permanent server is desired, we can also put this code on a fixed server for a certain fee from the server providers.

In the Firebase system, we use both the authentication part and the Realtime Database part here. The Firebase system works very easily and quickly based on these two features, and it is also very easy to set up this system with code from within the application. At the same time, we can perform these transactions without paying any fee.

As a result, our project is a project that can be developed very cheaply in terms of hardware and software. The fact that the development cost is cheap is also a beneficial feature that increases the general use and accessibility of the project. At the same time, we took care to develop our application in the software section in a way that would tire mobile devices as much as possible. In this way, we aimed to ensure user satisfaction in the hardware section.

3. Engineering Solution

3.1 The Economic Aspect

3.1.1 Cost Analysis

The economic requirements of our project in terms of software and hardware are quite low. Below you can see the cost situations of the project in a real company:

Software Costs:

- Work on the mobile application side
- Work on the Python side

Hardware Costs:

- A phone with Android operating system and minimum SDK 26

Application Publishing and Optional Costs:

- Developer account fee required for the application to be published on the Google Play Store
- Server fee if you want to publish the Python code on a fixed server instead of a local server
- Optionally using the paid version of Firebase

Apart from these, ongoing maintenance costs for the system may also constitute an expense.

3.1.2 Scalability

We developed our application by considering scalability. For now, instead of making the infrastructure of the application for many users to use at the same time, we have made it for a more average number of users, but we have created this infrastructure in a way that it can be easily developed with a small expense according to the situations that may arise in the future. Thus, we aimed to keep the cost of scalability to a minimum depending on future situations. Apart from these, ongoing maintenance costs for the system may also constitute an expense.

3.2 The Ethical Aspect

3.2.1 Privacy-related Situations

Encryption techniques were used to ensure that users did not experience any privacy-related problems. Thanks to this technique, users' passwords are end-to-end encrypted and made secure. At the same time, it was aimed to ensure that the user does not have any concerns about privacy by not sharing any information with any other parties.

3.2.2 User Consent

We have created a structure that requires user permission to use some of our features in the application effectively. We took care not to enable any features of the phone without the user's permission. Thus, we also wanted to protect our users in terms of ethical situations.

3.2.3 Fairness

All ideas of everyone involved in the project regarding its course and development were valued and given equal importance. The decision-making processes were carried out transparently, and the project also progressed by paying attention to the issues of bias and prejudice.

3.3 The Social Aspect

3.3.1 User Experience

Thanks to feedback from users, the missing aspects of the application will be eliminated. At the same time, additional features requested by users may be considered to be integrated depending on their suitability for the application. Since the most important development element for our application is the feedback of the users, we attach great importance to this situation and the analysis process of the feedback will be carried out meticulously.

3.3.2 Community Engagement

Since we attach great importance to receiving feedback, we aimed to establish our community by creating communities on some social media. Thanks to this, we will be able to objectively observe all the opinions about the application within our community, and at the same time, we will be able to communicate with users faster and bring our application to a better level faster. Creating a community will not only improve our application but will also have a positive impact on the company's image.

3.3.3 Social Benefits

Since our application is an application that aims to increase the focus of people with ADHD and attention deficit disorder, we aimed to make a project that is directly beneficial to society. Since many people in society have focus problems, these people cannot complete their tasks quickly in their daily lives, and therefore they experience general difficulties. We aim to provide comfort for these people with our application. In this way, we want to ensure that people can get their work done faster and have more time for themselves.

4. Implementation and Development Details

4.1 Image Processing

Image processing is one of the components of Environmental Photo Analysis feature, responsible for processing the inputted image and gathering the colour analysis. The feature was developed using Python using the capabilities of OpenCV. OpenCV library provides efficient and robust image processing enabling the gathering of clear information about the colours.

4.2 Object Detection

Object detection covers the majority of the Environmental Photo Analysis feature as it is the main part of that subsystem. It is responsible for finding and detecting around specified distracting objects and shiny objects by drawing bounding boxes around them to present the results to the user later on. The feature was developed using Python, OpenCV, and YOLOv8 as the main detecting model. The reason for using these technologies was as they were fast and secure enough for deploying it onto a server without any security issues with databases.

4.3 Mobile Application

The project consists of a mobile application where users can sign-up, login and logout to use the features. The application, with its friendly interface, enables users to help reduce the effects of ADHD while studying. The application was developed using Java Kotlin and Android Studio having a database for use on Google Firebase which provided a responsive and scalable solution. The mobile application has the features Pomodoro, To-do List, Sound Level Meter and Environmental Photo analysis. The application can also send notifications to users for reminders.

4.4 Pomodoro Study Assistant

As one of the main features of the application, pomodoro with a friendly and easy to use interface helps users organize their study times with a timer. The creates study and rest intervals with user preferences and reminds the users when to stop or start studying. The feature was created with Java and Android Studio to be added into the application.

4.5 To-Do List

To-Do List creates a basic list of tasks that users can add, remove or update depending on their preferences. The list pushes notifications to users whenever the time is up to remind them of their tasks. The list is created with Java and deployed onto Google Firebase to be stored.

4.6 Sound Level Meter

Sound Level Meter measures the noise in a room based on decibels (dB) and advises users not to study there if the dB is too high for comfort. The feature was created using Java using android libraries and displayed on the mobile application.

5. Testing

5.1 Test Planning

The test plan focuses on the strategies for validating the functionalities and the performance of the application. It describes the testing objectives, scope, and the resources required for each testing phase.

5.2 Unit Testing

Unit testing focuses on each component of the systems one by one, such as object detection, pomodoro etc. This testing mechanism ensures that each unit performs as they are intended to without the integration.

5.3 System and Integration Testing

This phase is important as some of the features were not tested on the mobile application before the integration. When everything was integrated into the mobile application, first tests were done using Android Studio. Different pictures were uploaded for photo analysis and different scenarios were tested to make sure they were working as they were intended to with decent speed.

5.4 Acceptance Testing

Acceptance testing was conducted to make sure that the entire system met the specified requirements and is ready for deployment to an Android Device. System's functionalities were validated once more with gathering feedback for any adjustments.

5.5 Performance Testing

Performance testing focused on the system's performance, responsiveness, stability, and scalability under different conditions. This included assessing the photo analysis' effectiveness and efficiency, the mobile application's responsiveness and performance, and the other feature's robustness and ability to handle user interactions. Performance testing was aimed to identify any performance issues or system issues.

5.6 Test Cases

5.6.1 Test Case 1: Sign-up, Login, Log-out

Steps:

1. Navigate to the Login/Registration Screen
2. Enter Credentials
3. Verify Login
4. Navigate to Register/Signup screen.
5. Attempt to sign up for a new account.
6. Verify user credentials via email.
7. Verify sign up.
8. Log-out and verify log-out.

Expected Results:

- The system should successfully operate and let the user login/logout.
- Signup should be verified via e-mail.

5.6.2 Test Case 2: User Navigation Testing

Steps:

1. Login as a user
2. Navigate to between different screens within the application.
3. Verify that the navigation is simple, smooth and responsive.
4. Go in and out between different screens/features.
5. Verify that all screens display the accurate features.

Expected Results:

- Navigation should be smooth and responsive.
- In-and-out navigation out of features should function correctly.
- All screens should display their respective features.

5.6.3 Test Case 3: Image Detection Testing

Steps:

1. Access the Environmental Photo Analysis feature.
2. Try to analyse the environment without access to the gallery.
3. Upload a picture of a study area.
4. Verify that the detection and colour analysis work correctly in the study area.
5. Verify that user gets correct results out of the feature feedback.

Expected Results:

- The feature should not access the gallery without asking for permission for the first usage.
- The analysis should be accurate, and the relative feedback should be correct.

5.6.4 Test Case 4: Sound Level Measurement Testing

Steps:

1. Access the Sound Level Detection feature.
2. Try to measure the sound level without access to the microphone.
3. Verify that the application asks for microphone access.
4. Use the feature to measure the sound level in dB of the environment.
5. Verify that if the area is loud, the application gives relative feedback.

Expected Results:

- The feature should not access the microphone without asking for permission for the first usage.
- The feature should successfully measure the sound level in the area and provide relative feedback.

5.6.5 Test Case 5: Pomodoro Testing

Steps:

1. Access the Pomodoro via navigation menu.
2. Add study and break intervals.
3. Verify that the timer works as its intended to.

4. Verify that you cannot switch screens when the timer is counting down to prevent distractions.

Expected Results:

- The timer should work as intended with a responsive navigation functionality.
- Application should warn the user if they try to switch screens within the application.

5.6.6 Test Case 6: To-Do List Testing

Steps:

1. Access the To-Do List via navigation menu.
2. Add tasks to the list.
3. Update tasks that are already present.
4. Remove tasks from the list.
5. Verify that the user gets a notification when the task is up.

Expected Results:

- The list should work as intended with all the features add, update and remove.
- Make sure users get notifications when the time interval meets the specified time.

5.7 Test Case Results

According to the application's conditions we have conducted the testing and encountered success for each and every one of them. The primary challenge was getting the image detection to function correctly with all its features including the colour analysis and object detection. After some effort with YOLOv8 and OpenCV the feature was tweaked to work as intended.

6. Project Status

6.1. Project Progress Overview

The FocuZone project progressed simultaneously according to the project progress plan we determined with our teammates at the beginning. Currently, our project has been successfully finalized as we proceed along this plan. The construction stages of the project can be accessed from our backlogs on our website.

6.2 Milestones Achieved

6.2.1 Mobile Application UI/UX Design

Designing our mobile application in the most useful and simple way in terms of user experience was an important milestone in our project. Since the application aims to reduce distraction, the screens the user will use should not be distracting. We made our UI/UX designs in this direction.

6.2.2 Image Processing

In the Study Environment Photo Analysis feature of our FocuZone application, which determines whether the color distribution and objects in the study area images taken by the user and uploaded to the application. is suitable for a productive working environment, is the biggest feature that distinguishes FocuZone from other applications. So the Image Processing implementations were big milestones for our team.

6.2.3 Frontend Implementation of the Application

Implementing the application's frontend in accordance with UI/UX designs, creating features to be handled from the frontend, and then creating them to integrate with the backend.

6.2.4 Integration of Backend and Frontend

Integrating the application with the backend codes used in the Study Environment Photo Analysis feature and performing request and response operations through the local server. Some data with features such as Study Assistant and To-do list integrated into the Firebase Realtime Database system and the necessary data can be retrieved from there and displayed on the screen.

6.2.5 Mobile Application Tests

All necessary checks are carried out by our team before the application is released to the user.

6.3 Challenges Encountered

6.3.1 Local Server Creation with Flask

We have been encountered issues with the local server setup using Flask, but by reconfiguring the server settings and implementing error-handling mechanisms we successfully resolved these problems, and as a result we had a stable server environment.

6.3.2 Backend and Frontend Integration

Integrating the backend and frontend was a challenging part for our team, particularly in Study Environment Photo Analysis Feature we had issue with integrating the yolov model into our application.

6.3.3 Training the YOLOv Model

Training the YOLOv model posed difficulties due to the need for a large dataset and high computational resources. We had difficulty with the YOLOv model that we trained to detect a general mess rather than a single object.

7. Discussion of Current Issues

7.1 Discussion on Current Problems

Our main focus at FocuZone is the difficulties that individuals with ADHD face in their daily lives, especially during study sessions. People with ADHD often struggle with various aspects of maintaining focus and productivity due to their condition. They often have difficulty organizing and collectively planning their tasks, keeping track of deadlines, and choosing environments in which they can focus, which greatly reduces their productivity along with increasing their stress levels. Maintaining focus for long periods of time is particularly difficult, resulting in frequent interruptions and incomplete tasks. At the same time, individuals with ADHD have poor time and planning management skills, causing them to not be able to devote enough time to their tasks. Additionally, people with ADHD are highly sensitive to environmental distractions such as background noise and visual clutter, which significantly disrupt their ability to concentrate. These difficulties not only impact academic or professional performance, but also contribute to increased stress and frustration, making it difficult for individuals with ADHD to achieve their full potential.

7.2 Relevance of the Project to Current Problems

Our project, FocuZone, aims to overcome these challenges by providing a comprehensive solution tailored to the needs of individuals with ADHD. The application includes four different features designed to improve the focus time, productivity, efficiency and many positive effects of people with ADHD or users with concentration problems. The to-do list, which includes tasks reminded by notifications, helps users organize their tasks and plan by helping them set reminders for upcoming due dates. The study assistant feature allows users to take scheduled breaks with the Pomodoro system and adjust their study patterns to better suit their needs, ensuring users have a balanced study routine without being overwhelmed. The sound level measurement feature evaluates the user's environment and tells the user to change the environment if the sound level is auditory distracting. Finally, thanks to the study environment photo analyzer feature, it detects the color scale and distracting or shiny objects in the users' working environments, informing the user about the productivity of the environment, and recommending changing the environment. By addressing these specific issues, FocuZone provides a holistic approach to improving the work habits and productivity of individuals with ADHD, improving their ability to focus and manage time effectively, and contributing to a more structured and less stressful learning experience.

8. Project Elements

8.1 Technology and Tools

Technology Stack

- Java: We used Java language when implementing the Android application. ava provides a reliable and widely supported foundation for building robust mobile applications.
- Kotlin: With Java we also used Kotlin when implementing the Android application.
- Python: We used Python for backend development and implementing the machine learning models. We did the image processing feature by using python.
- OpenCV: OpenCV library used for image processing tasks. OpenCV provides a wide range of tools for image manipulation, analysis, and machine learning integration which are crucial for our project.
- Flask: We used flask to develop the backend services.
- YOLOv8: We used YOLOv8b for object detection in our Study Environment Photo Analyzer feature. YOLOv8 is an advanced object detection model employed for detecting and tracking objects within images and video streams.

Tools and Software

- Android Studio: We used Android Studio for the development, testing, simulating and debugging of the Android application.
- Jupyter Notebook: We used Jupyter Notebook for developing, testing and visualizing the machine learning models that we wrote with python.
- Visual Studio Code: We used Jupyter Notebook for developing, testing and visualizing the machine learning models that we wrote with python.

8.2 Simulation, Analysis Algorithms and Results

As FocuZone’s developers, we tested individually to ensure that we got accurate and efficient results from each of the features we made in our project. Whether we receive notifications in the to-do list feature, whether session adjustments and transitions are correct in the study assistant (pomodoro) feature, whether it measures the sound correctly in the sound level meter feature and whether it shows the correct result accordingly. And in our most striking feature, the study environment photo analyzer, we tested in our own study environments whether each of us could get accurately detect any distractions, of course, whether these results increased our productivity.

As a result of these simulations and tests, we concluded that there will be an increase in the productivity levels of the users who will use FocuZone.

8.3 Project Management and Team Collaboration

8.3.1 Project Management Techniques

When we started our FocuZone project, we progressed throughout the project with the Agile Methodology that we decided as a group. We always progressed in the project in an up-to-date manner in coordination with the team, with the tasks that we divided and assigned into sprints and with regular meetings.

8.3.2 Roles and Responsibilities

As a team of four people which contains Hüseyin Sina Ceylan, Şevval Kaplan, Eren Sakarya and Şevval Yardımcı, we successfully completed each sprint of the project through effective teamwork and collaboration. We distributed the work in a general way, depending on the personal interest and experience of each team member. Hüseyin Sina Ceylan was responsible for the mobile implementation of the application using Java and Kotlin languages and the integration of the backend and frontent parts of the project, Eren Sakarya was responsible for the Yolov8 training part, the distracting object detection part and the shiny object detection part, Şevval Yardımcı was responsible for the distracting object detection part and detection of color scale of the environment part, Şevval Kaplan was responsible for the UI/UX designs of the mobile application and the distracting object detection part. Although we generally divided our roles this way, everyone took part everywhere.

8.3.3 Communication Tools

One of the reasons we used the Agile methodology was to always be informed about each other's project progress. We spent a few minutes with our teammates on the days we had classes at school to update each other and talk about the project. At the same time, we created a group in an application that we were all in and updated each other from there.

8.3.4 Evaluation of Team Performance

We successfully completed our project in accordance with our goal setting, thanks to the successful and smart solutions of all our teammates. We solved the problems we encountered by producing quick and effective results without causing any negativity in our project.

9. Conclusion and Evaluation

The purpose of this document is to introduce in detail our FocuZone project, which we designed to help individuals with ADHD manage their daily tasks and maintain their focus. The app includes a variety of innovative features aimed at creating structured work routines, increasing work efficiency and optimizing work environments to minimize distractions. This document aims to describe the purpose of the system, target users, system requirements, functions and operating conditions, together with system interfaces and scenarios. This comprehensive overview is intended for both collaborating institutions and business developers.

The FocuZone project specifically targets individuals with ADHD, but it is also a useful app for anyone who has trouble focusing and productivity. The developed mobile application processes various types of data to provide real-time support for task management, work planning, sound level monitoring and workspace optimization. By leveraging these functions, we aim to reduce the daily difficulties faced by individuals with ADHD, thereby increasing their productivity and quality of life.

Our most important goal is to support users to manage their tasks efficiently and maintain their focus, especially in environments where attention may be distracted. FocuZone provides real-time data processing and personalized recommendations to help users create optimal work environments and schedules. This system is designed to be accessible and useful to a wide audience; Increases productivity and focus through a user-friendly and reliable application.