
Bilkent University

Department of Computer Engineering

# Senior Design Project

*EyeSight*

# Analysis Report

Group Members: Cemil Şişman, Derviş Mehmed Barutcu, A A M Jubaeid Hasan Chowdhury, Mustafa Azyoksul, Onur Mermer

Supervisor: Varol Akman

Jury Members: Fazlı Can, Hamdi Dibeklioğlu

Analysis Report
March 23, 2020

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Design Project course CS491/2.

# Contents

# 1. Introduction

In this technological era, most of the productions and developments are aimed at the entertainment business. In our opinion, technology should be used to help handicapped people as much as it's used for entertainment purposes.

Using this motto, EyeSight is intended to hold visually impaired people's hand to make their life less challenging. Despite the rapid development of computers, visually impaired people still use a stick to navigate themselves through obstacles. EyeSight is imagined to become a new eye for those people. Using the camera of the smartphones, EyeSight will describe the user's surroundings such as objects, walls, humans etc. to them in realtime and also navigate them to where they want to go, even for small distances such as going from kitchen to the bedroom etc. Users will also be able to register their relatives, friends and family members to EyeSight. The app will recognize human faces and will inform users specifically on who those people are.

EyeSight will have a user friendly UI for visually handicapped people. It will be accomplished by using voice overlay and taking voice inputs from the users which will be used to help them interact with the app. We are determined to ensure that visually impaired users will have no issue in using the app without any help from other people. Because the deadlines of this project are limited, we decided to implement EyeSight just for indoor usage. EyeSight will initially not be suited for outdoors because of the vital risks that might occur.

# 2. Current System

There is a mobile app called Seeing AI[1] that is developed by Microsoft which is a similar app in terms of intentions. It's also aimed to help visually impaired people but its interface isn't suited for users to use it without help. It has different sections for objects and human faces. It's also not working in realtime, you need to upload photos from the library of the phone for the app to work. We are also developing a navigation system. Overall, we are using this app as a stepping stone to improve the quality of EyeSight.

There are also other apps such as money identifiers, object identifiers[2] etc. But none of those apps are comprehensive enough in our vision. There is also an app called Be My Eyes that connects a person to the visually impaired user via video conference[3]. This person navigates the user through obstacles. This app requires a person to help the visually impaired user. However, EyeSight is intended to help its users by navigating without any help from other people.

BuzzClip is another app to help visually impaired people, it uses ultrasound to navigate its users through the obstacles[3]. To us, it's also not comprehensive enough. We are going to be describing the view of the user to try to make them feel like normal people.

# 3. Proposed System

## 3.1. Overview

Eyesight is an object detection system which aims to be used by the people with visual impairment. It will be implemented as a smartphone app for android to make it accessible to use. Main feature of the system is to provide a constant stream of data about the environment to the user which has visual impairment to increase awareness of the user about its environment with minimal effort. In order to decrease the effort of the user about usage of the app properly, a user interface for a person with visual impairment will be prepared which includes interaction with smartphone screen, volume buttons on the phone and proper audio feedback together with text-to-speech features. In addition to the main data feed provided via voice by the app, a navigation system for indoors will be implemented to help users to navigate and give specific room informations to inform the user about the environment. In order to give more information about the people in the environment, specifically to recognize a family member, the data of family members with their face images will be kept in the app to recognize them and provide that information to the user. During all the user interaction and feedback to the user both audio feedback and text-to-speech features will be implemented as an audio overlay system in the app to make the app easy to use.

Object detection feature should work with a smartphone with a camera that can record a video. In order to gain the sense of distance, a separate sensor should be used since proximity sensors of the smartphones have very short ranges for our use. Gyroscopes of smartphones will be used to ensure that the user holds the phone in a suitable position to gather visual data properly. App should give the object information together with any other obstacle user faces in their way to give the sense of indoor environment in audio format. Navigation feature in the system should work with specific symbols in the environment that are prepared and registered to the app beforehand so it can process the location information of the user, give the data about the room they are in and navigate them through their destination.

## 3.2. Functional Requirements

### 3.2.1. Object Detection

- EyeSight should process the video inputs from the camera and identify the objects in the frame in real time.
- EyeSight should tell the objects' name, their color and their position in the frame to the user.
- If the app detects an emergency situation or the blind person presses the emergency button, the app should automatically call the guardian's phone number which should be registered through the family members registration system.
- Distance of the objects and the depth should be also approximately calculated with the help of separate sensors to inform the user.
- Registered family members that are captured by camera should be detected by the app and the app should notify the user about the family member's profile.
- Registered signs for navigation system should also be detected to help the user through navigation in the registered place.

### 3.2.2. Navigation with Symbols

- App should provide proper settings to create a location profile for the user to register the signs in the place for later navigational usage.
- During the registration process proper instructions should be given to a helper who does not have visual impairment as opposed to regular user so that helper can register the location to the app.
- As the signs, colored printed circles should be used for easy detection and accessibility for the user since they can print as many of the signs as they want. System should save the placement of each specific sign with associated id and group the signs by their room in the location so the user can be informed about room information and the system can use this information to implement navigation.

- Users with visual impairment should be notified about room information if they entered another room and should be able to get help from the app to help him navigate from one room to the other via voice commands.

### 3.2.3. Audio Overlay

- Since the target audience for the app are people with visual impairment, proper audio feedback should be constantly provided.
- In the menu, while the user pressing and dragging his finger on the smartphone screen, names of the buttons which are placed and divided in the screen should be called by the app so the user can pick the option they desire.
- During image capturing and processing detected objects, obstacles, family members, room and navigation information should be provided to the user via audio feedback.
- App should notify the user via voice if the camera cannot get a proper image to give information due to poor positioning so that the user can adjust the position of the smartphone so the camera can get a proper view angle to get an image and process it.

### 3.2.4. Family Members Registration

- The user's relatives can take a photo of their faces to introduce themselves to the application. After they take their faces and enter their names, each time they enter the camera frame, EyeSight will tell their names to the user.

- Registration for the family members should require family member data -name, surname, relativity, contact information like phone numbers etc.- and family members picture so that recognition may become possible during object detection.

- Helpers who do not have any visual impairment should help the user during the registration process but once the data is saved the user should be able to access the data and make calls to the family members without any help from the others.

## 3.3. Nonfunctional Requirements

### 3.3.1. Usability and Accessibility

- Since the target of the application is visually impaired people, the application should have features that help users vocally.
- Visually impaired users might need help from others to use some of the features of the application.
- Both the users with visual impairment and helpers who do not have visual impairment should easily use the features which aim specifically at them without any training or help from the other group. If required, instructions should be provided by voice.

### 3.3.2. Scalability

- The app should be scalable to any number of users. The processing power of the backend image detection servers needs to be scalable to any number of users.

### 3.3.3. Extensibility

- Since the outdoors might increase the user's chance of facing a vital circumstance, the scope of the application is initially aimed at indoors only. But it is always possible to extend the application to be usable outdoors.

### 3.3.4. Performance

- Taking the frame from the camera of the user, processing it and creating the voicemail shouldn't take more than 1 seconds. It might take a shorter or longer time depending on the complication of the scene.

### 3.3.5. Reliability

- Reliability is a major requirement for the application. It should always be working precisely most of the time to avoid any undesirable circumstances. The margin of error is very small since the errors might result in vital situations.

## 3.4. Pseudo Requirements

- EyeSight will be a mobile application on the Android platform.
- The mobile app will be implemented using Android Studio[4].
- The mobile app portion of the system will be implemented using Java and machine learning modules will be implemented using Python.
- Users are required to have an Android device with Android 5.0 Lollipop or higher, in order to use the application since we will use API level 21 during the development.
- As the version control software and the source code management system, GitHub will be used.
- To convert the commands on the application to voice, Google's text-to-speech API will be used[6].
- To process user voice input, Google's speech-to-text API will be used[7].
- Systems development life cycle of the system will be carried out according to object oriented programming principles.
- All of the unit tests and the final product tests will be carried out in detail before reaching out to the end-user.
- The application language will be in English only.
- Personal data will be used to enhance the user experience and it will be secured from theft and malicious use. The user will be in control of what happens with their personal data.

## 3.5. System Models

### 3.5.1. Scenarios

#### 3.5.1.1. Scenario 1 - Install

**Use Case:** Install

**Actors:** Companion

**Entry Condition(s):** In the android application store, companion is able to download and install the application.

**Exit Condition(s):** When install process is completed

**Flow of Events:**

1. Companion will use the application store the visually-impaired person's phone has to download and install the application.
2. Companion accepts the permissions and confirms the install.
3. Most of the process is handled by OS itself.

#### 3.5.1.2. Scenario 2 - Launching the Application

**Use Case:** Launch

**Actors:** Companion, User

**Entry Condition(s):** Tapping on the app icon on the device or through sound control.

**Exit Condition(s):** Tapping home button, tapping power button or battery running out.

**Flow of Events:**

1. User launcher the sound control such as Google Assistant on their phone.
2. User says "launch eyesight"

**Alternative Flow of Events:**

A. If Companion is using the phone,

a. Companion taps on the app icon on the device screen.

#### 3.5.1.3. Scenario 3 - Sign Up

**Use Case:** Launch

**Actors:** Companion

**Entry Condition(s):** After the Companion installs the mobile app and launches the mobile app, the user is prompted to sign-up with the Google account.

**Exit Condition(s):** The companion or user signs in with the user's Google account.

**Flow of Events:**

1. Companion enters the app on a device that hasn't been logged in yet.
2. Companion is visually and voice-prompt to sign in using the user's Google account.
3. To sign-up, the companion or user confirms the sign-up.
4. Companion or user is directed to the main screen.

**Alternative Flow of Events:**

A. If no Google account is connected.

a. Companion is prompted to connect a Google account or login with the user's Google account in the app.

#### 3.5.1.4. Scenario 4 - Use the Main Camera Functionality

**Use Case:** Use the Main Camera Functionality

**Actors:** User

**Entry Condition(s):** From the main menu

**Exit Condition(s):** Menu button on the bottom of the camera functionality

**Flow of Events:**

1. User opens the mobile app.
2. User is prompted with the main screen. User selects the "start" button.

**Alternative Flow of Events:**

A. If the app is opened the first time

a. User first goes through the steps in 3.5.1.1.

b. Steps continue from (1).

#### 3.5.1.5. Scenario 5 - Give Voice Command

**Use Case:** Give Voice Command

**Actors:** Companion, User

**Entry Condition(s):** From any point of the application

**Exit Condition(s):** Finishing giving voice command or cancelling giving a voice command.

**Flow of Events:**

1. User gives the voice command after they click the volume up button.
2. If a valid command is identified, the application starts executing the command.

**Alternative Flow of Events:**

A. If the voice command giving is cancelled

a. User takes too long to respond or gives a cancelling command saying “done”.

#### 3.5.1.6. Scenario 6 - Navigate

**Use Case:** Navigate

**Actors:** User

**Entry Condition(s):** From the main camera functionality

**Exit Condition(s):** Menu button on the bottom of the camera functionality

**Flow of Events:**

1. When the user is on the main camera functionality, the application automatically start processing the visuals from camera input.
2. Using the voice controls, the user is able to specify that they want to navigate to some place.
3. Then the application gives voice command directions to the user until the user gets to their destination.
4. The app confirms that the user arrived at their destination.

**Alternative Flow of Events:**

B. If the user terminates the navigation

a. Using the voice controls, the user is able to terminate the navigation process.

#### 3.5.1.7. Scenario 7 - Add a Family Member

**Use Case:** Add a Family Member

**Actors:** Companion

**Entry Condition(s):** Profile button from settings menu

**Exit Condition(s):** Menu button on the bottom of the screen

**Flow of Events:**

1. Companion presses the add family member button in profile
2. Companion types in family member information

#### 3.5.1.8. Scenario 8 - Get Family Member Information

**Use Case:** Add a Family Member

**Actors:** Companion, User

**Entry Condition(s):** Profile button from settings menu

**Exit Condition(s):** Menu button on the bottom of the screen

**Flow of Events:**

1. Companion or user says “family members” to get the all family member information or says “information family_memebers_name” using voice command.

**Alternative Flow of Events:**

A. If companion is not using voice command

a. Companion presses the family members button in profile.

b. Companion selects the family member he/she wants to get information of.

#### 3.5.1.9. Scenario 9 - Add an Emergency Contact

**Use Case:** Add a Family Member

**Actors:** Companion

**Entry Condition(s):** Emergency contacts button from profile menu

**Exit Condition(s):** Menu button on the bottom of the screen

**Flow of Events:**

1. Companion presses the add emergency contact button.
2. Companion types in emergency contact information.

#### 3.5.1.10. Scenario 10 - Edit an Emergency Contact

**Use Case:** Add a Family Member

**Actors:** Companion

**Entry Condition(s):** Emergency contacts button from profile menu

**Exit Condition(s):** Menu button on the bottom of the screen

**Flow of Events:**

1. Companion presses the emergency contacts button in profile.
2. Companion selects the contact to edit.
3. Companion types in information he/she wants to override.
4. Companion hits save.

#### 3.5.1.11. Scenario 11 - Remove an Emergency Contact

**Use Case:** Add a Family Member

**Actors:** Companion

**Entry Condition(s):** Emergency contacts button from profile menu

**Exit Condition(s):** Menu button on the bottom of the screen

**Flow of Events:**

1. Companion presses the emergency contacts button in profile.
2. Companion selects the contact to delete.
3. Companion hits delete.
4. Companion confirms.

#### 3.5.1.12. Scenario 12 - Call an Emergency Contact

**Use Case:** Add a Family Member

**Actors:** Companion, User

**Entry Condition(s):**

**Flow of Events:**

1. Companion or user says “call family_memebers_name” using voice command.

#### 3.5.1.13. Scenario 13 - Change Object Count Size

**Use Case:** Change Object Count Size

**Actors:** Companion, User

**Entry Condition(s):** From settings menu

**Exit Condition(s):** Menu button on the bottom of the screen

**Flow of Events:**

1. Companion slides the slider in the settings menu.

#### 3.5.1.14. Scenario 14 - Clean Information

**Use Case:** Clean Information

**Actors:** Companion

**Entry Condition(s):** From options menu

**Exit Condition(s):** Exit button on the bottom of the screen

**Flow of Events:**

1. Companion hits the clear history button in the settings menu.
2. Companion confirms.

**Alternative Flow of Events:**

A. If companion is using voice command

a. Companion says “clear history” using voice command.

b. Companion says “confirm” to confirm.

### 3.5.2. Use Case Model


Figure 1: Use Case Diagram

### 3.5.3. Object and Class Model

#### 3.5.3.1. Class Diagram

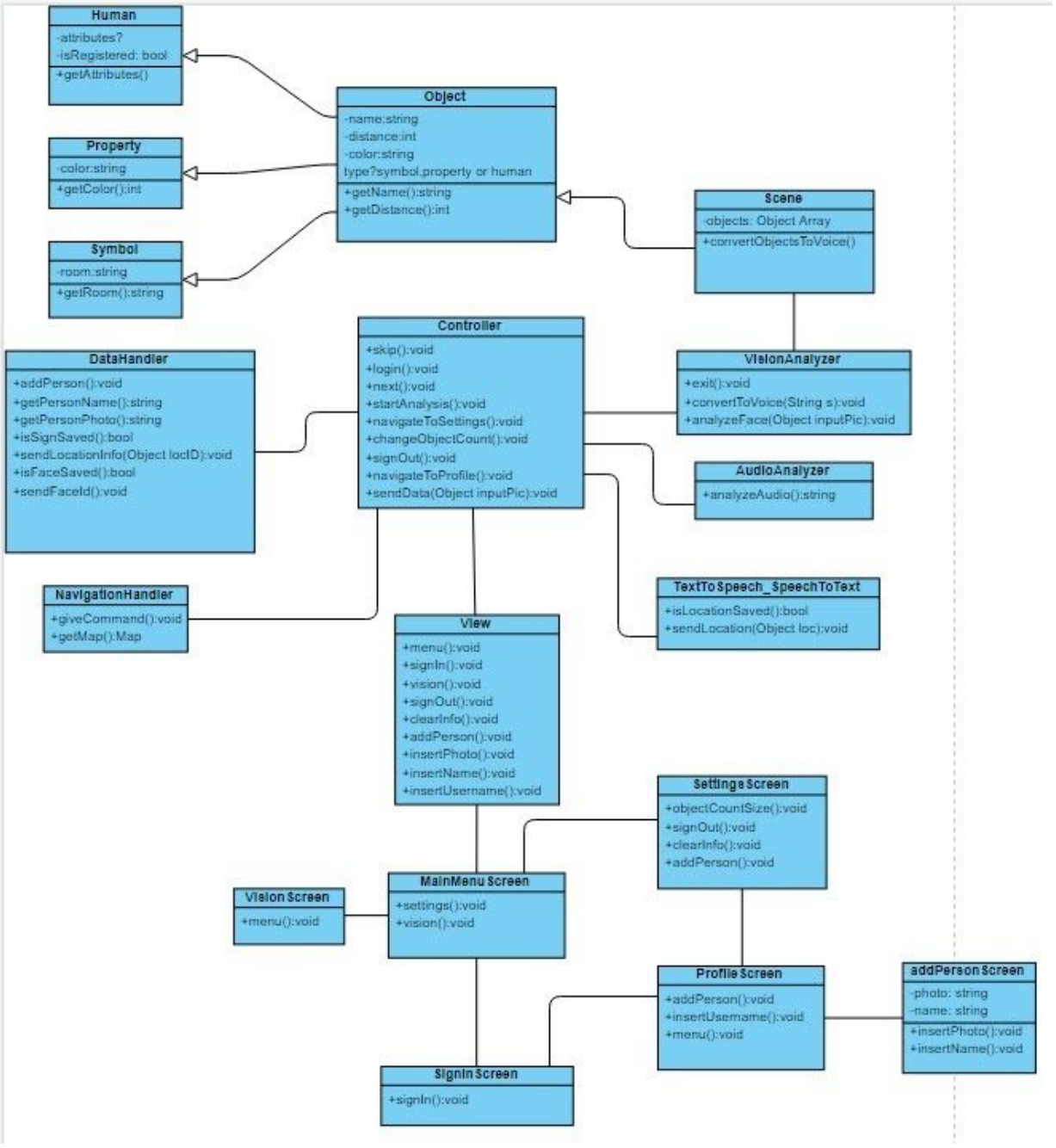


Figure 2: Class Diagram

### 3.5.4. Dynamic Models

#### 3.5.4.1. Main Functionality Sequence Diagram

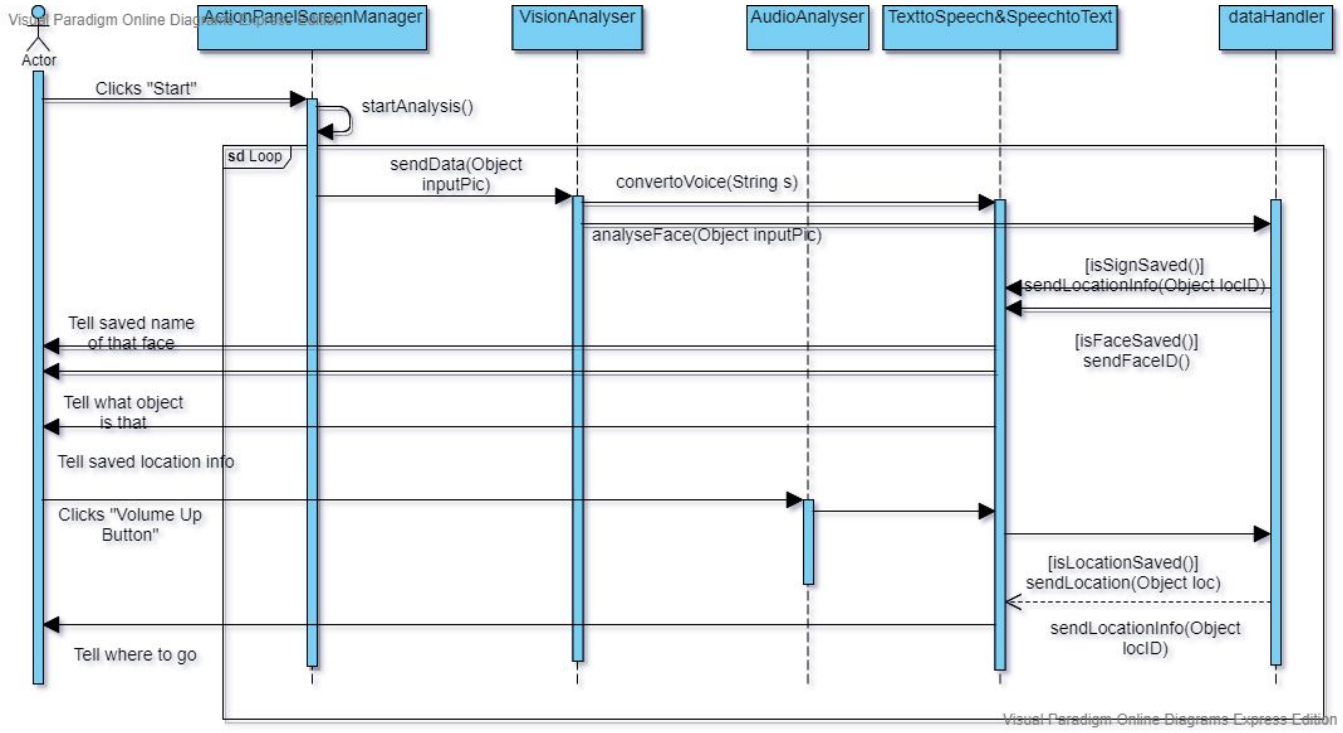


Figure 3: Main Functionality Sequence Diagram

When a user clicks the start button in the MainMenuScreen, EyeSight automatically starts analysing the camera inputs. The inputs are sent to the VisionAnalyser to be analysed. The outputs of the VisionAnalyser will be sent to the TexttoSpeech class where the input will be synthesised by the TexttoSpeech class then it will notify the user accordingly. In addition to this, the outputs of VisionAnalyser are also sent to the dataHandler to look for location signs that we have defined, or faces. If there is data saved beforehand, dataHandler will notify the TexttoSpeech class so that EyeSight can inform the user. If the user presses the volume up button, it will trigger the microphone to listen to the user's location information. The input is analysed by audioAnalyser and the output from audio analyser will be passed to TexttoSpeech class and it will pass the data to the dataHandler to be searched. After this, EyeSight will understand where the user is and where the user wants to go, therefore the necessary directions will be given to the user.

### 3.5.4.2. GUI Interaction Sequence Diagram

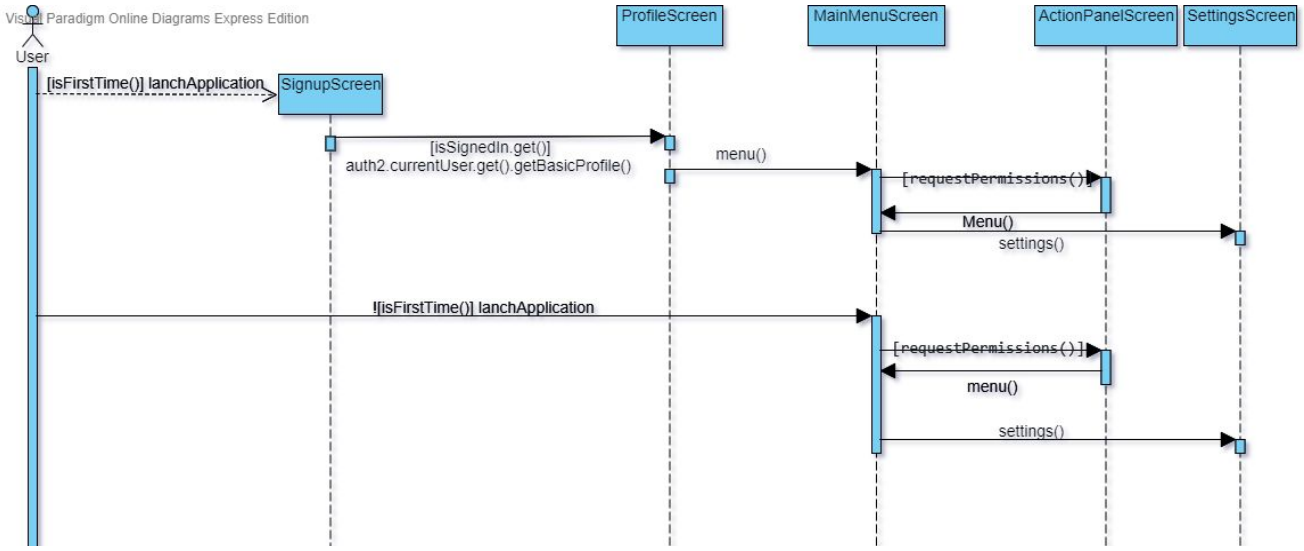


Figure 4: User Interface Interaction Sequence Diagram

The user interface sequence diagram (Figure 5) is given in above. If the user launches EyeSight for the first time, S/he has to sign up with their google account so that we can create the user's profile. Next, the user is navigated to the ProfileScreen for adding friends or family members, or add location information to the application. In this screen there is some information about how to install the direction stickers to the house and then they can start to add location. Adding friends/family information is not mandatory but we recommend it to be done. Next screen is the MainMenuScreen where The user can start the application or go to the settings. Since EyeSight is an android application, in the first usage EyeSight requests some permissions for reaching camera and microphone and then it will become functional. The user can return to the MainMenuScreen from ActionPanelScreen. From MainMenuScreen, the user can also go to the SettingsScreen.

### 3.5.4.3. Activity Diagram

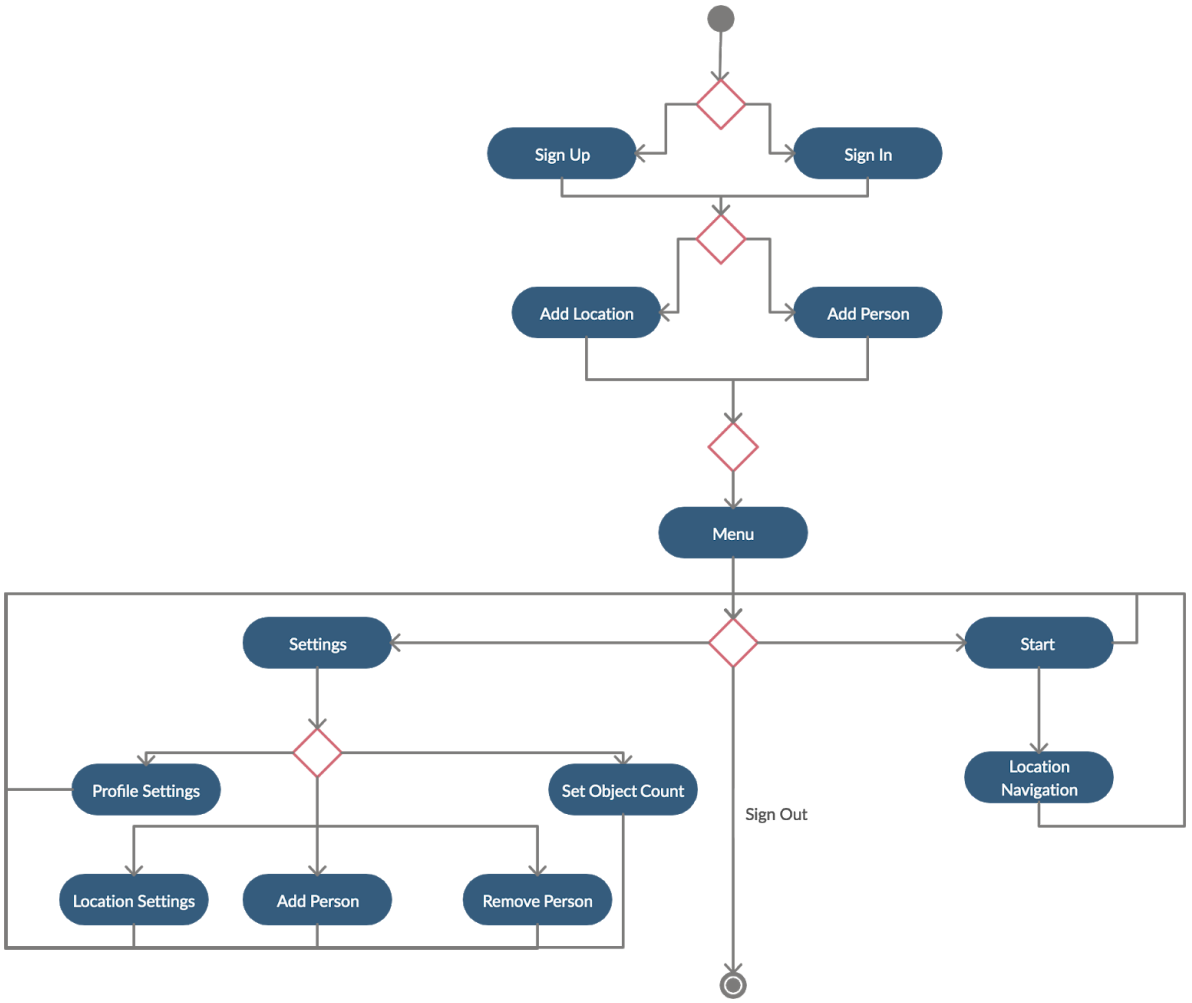


Figure 5: Activity Diagram

When the user first opens the app, the companion first sign up or sign in. Then the com

### 3.5.5. User Interface

The interaction between The user and the application will be “press and drag” method. Namely, when the application screen is fully loaded, the user can touch the screen and while pressing, they should drag their finger. If the finger is on a button, EyeSight will tell the users which button their finger is on. If they interact with this button, they will just release their finger on that button.

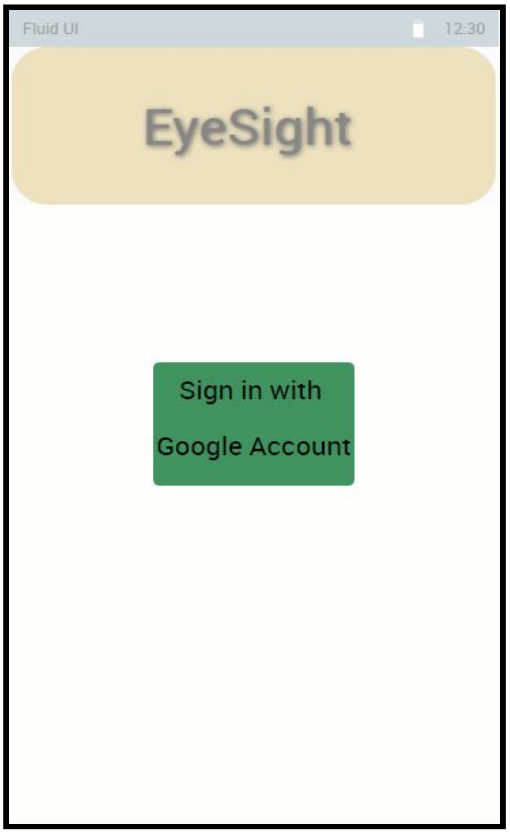


Figure 6: Sign In Screen

This is the first screen that will welcome the user for the first launch. We request users to sign in to create their profile.

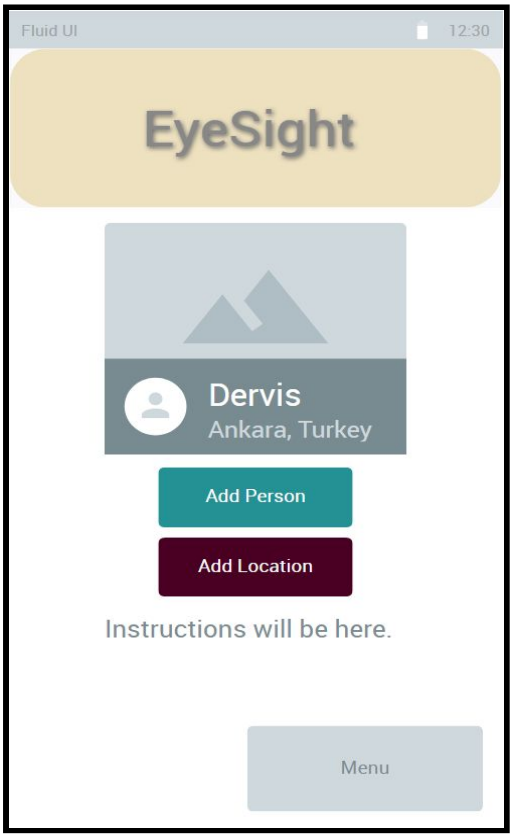


Figure 7: Profile Screen

This is the second screen when installing the application. In this screen, EyeSight requests users to add relatives or friends to be recognized by the app later on. There will be some instructions to install the location stickers and then The user can click the “add location” button to add location information. Then they can go to the menu.

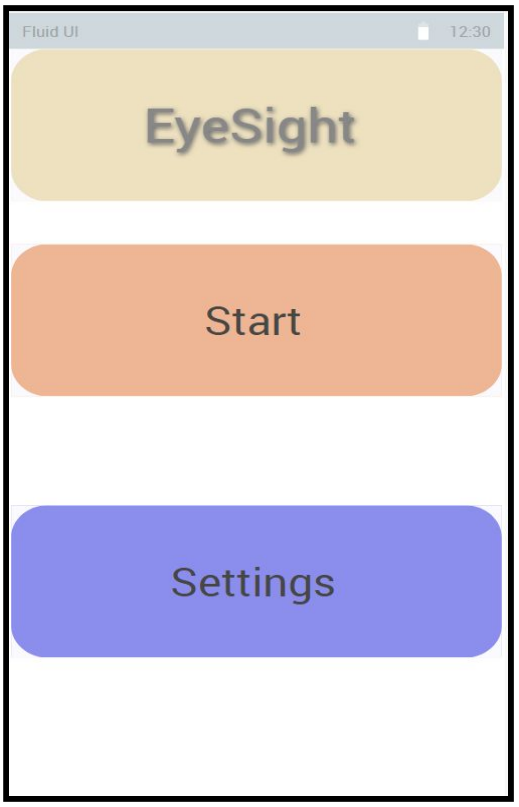


Figure 8: Main Menu Screen

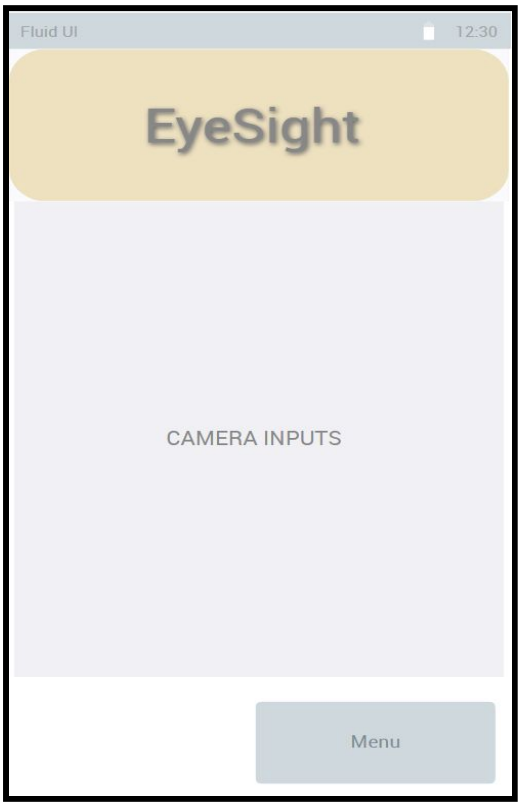


Figure 9: Main Function Screen

This is the main menu screen for EyeSight. We decided to keep this screen as simple as possible for the users who has visual impairment. The users can start the analysis by pressing the start button.

This screen is where main analysis and the interactions will happen. This screen will be open and process the camera inputs.


Figure 10: Main Function Screen : Destination Defined

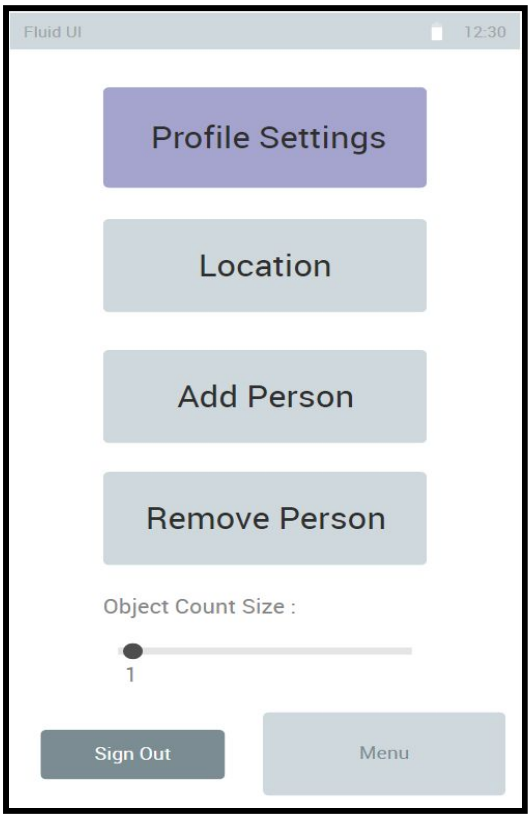


Figure 11: Settings Screen

This is the screen that shows the user's current location and where they want to go. While processing is continuing, if the user presses the volume up button and say where s/he wants to go, we visually show the locations on the screen to help understanding of others.

This page is the settings page. The users can view their profiles, modify location information, add person or they can remove person from saved ones. Also they can adjust the detected object count from the settings page. This means while processing the camera inputs there will be many objects detected by the EyeSight. Telling all of these objects to the user will create noise so we provide a simple slider to the users to adjust how much objects they want to hear.

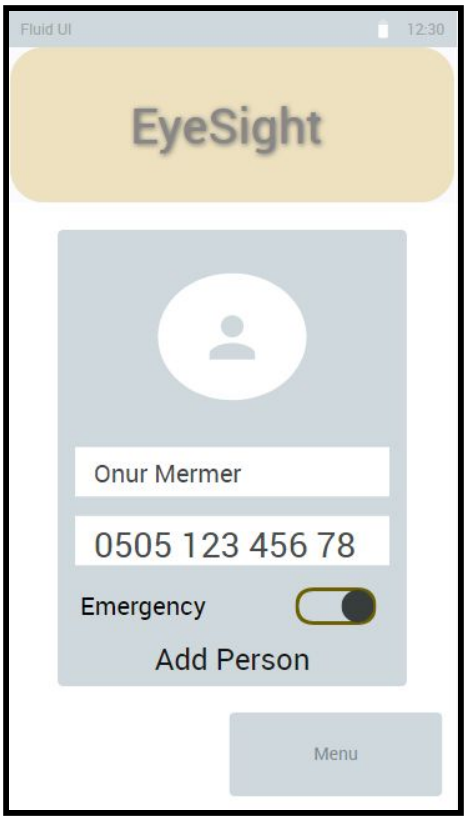


Figure 12: Add person Screen

This screen is for adding people. They enter their name, cell phone and their photos to be recognized later on. if The actual user wants that person to be in emergency contact, they just need to mark the related box and click the “add person” button.

# 4. Other Analysis Elements

## 4.1. Consideration of Various Factors

In order to ensure that EyeSight is comprehensive enough in its design, different factors that will affect our design process and our decisions are explained in this section.

### 4.1.1. Public Health

EyeSight is a smartphone application which targets people with visual impairment as users thus during the decision processes one of the top priorities should be ensuring safety of such users especially for the object detection and navigation features in order to prevent any physical harm. We should consider the condition of a person with visual impairment in order to not harm both physical and mental health. Such people have hard time during the day even for basic activities where another person who does not have visual impairment might easily handle the situation thus we should design the features to be easily usable for people with visual impairment to ensure they won't have hard time during usage of EyeSight.

### 4.1.2. Public Welfare

Since EyeSight itself aims to improve the well-being of people with visual impairment all the features should be designed with that consideration. EyeSight also should not disturb others in the environment with loud sounds.

### 4.1.3. Public Safety

We should consider the safety of private information of such users for the profile and family registration features. Private information should be accessible to only the user and companion.

### 4.1.4. Social Factors

We should consider any possible social interaction we may add for the users with visual impairment during the design. There are already few applications who target such users so in order to create the feeling of regular smartphone application usage, we should always consider the possibility of social interactions between users and helpers in every feature.

### 4.1.5. Global Factors

During decision making we should consider already existing popular applications and their design choices which may help the users since most of the people are familiar with such applications. For the registration and google profile are the most basic features which should resemble already existing popular designs so users may easily use EyeSight.

### 4.1.6. Economic Factors

In order to reach many people, EyeSight should have no cost for the users.

### 4.1.7. Cultural Factors

Most of the features we will add to EyeSight should be universally understandable but we probably would not have to consider each specific cultural effect since nearly all the people around the world who have visual impairment have similar experiences and similar solutions to their problems which we may consider as a cultural factor.

### 4.1.8. Environmental Factors

EyeSight should not create much noise then necessary to not disturb others in the environment.

Table 1: Considered Factors Table

| Factor | Effect Level (Out of 10) | Effect |
|---|---|---|
| Public Health | 10 | Possible physical harm during object detection, navigation and mental well being |
| Public Welfare | 7 | All of the features have consideration of welfare of people with visual impairment |
| Public Safety | 5 | Safety of private information |
| Social Factors | 4 | Possible social interaction between users and companions |
| Global Factors | 3 | Already existing designs and trends which are familiarized by public |
| Economic Factors | 1 | No cost for application |
| Cultural Factors | 1 | No culture specific constraints other than people with visual impairment |
| Environmental | 1 | No noise pollution |

## 4.2. Risks and Alternatives

Object detection is the main feature of the EyeSight which has the risk of not being accurate enough in real time. Most of the analysis will be done in real time in the smartphone and there would be times where there are many objects in the environment. We may implement it so that even if the application would not return an audio feedback it would recognize the object and we may recognize every single object in the environment but it might be too much processing for giving the information of just one or two objects. On the other side if we decide to recognize as many objects as we need to tell then we may face inaccuracies and waiting times for each new object that is going to be recognized. We should balance according to accuracy of recognition and the time it requires to recognize objects.

Navigation feature uses the symbols as specific pointers to recognize the path between rooms so that it can lead the user to the specific room from the other room. There is the risk that such a system with pointers may be problematic if the symbol is hard to detect. Another alternative was the usage of GPS location however we got rid of that idea since it would be not accurate as much as we want in small locations. As long as we choose a recognizable symbol, a system with pointers and prior decided paths might work better than a system with GPS in small locations.

Table 2: Risks Table

| Risk | Likelihood (Out of 10) | Effect on the project | B Plan Summary |
|---|---|---|---|
| Object detection accuracy and speed is not satisfactory | 5 | It's the main feature of the project thus it may cause project to fail since we cannot give information to user in a satisfactory accuracy and speed | We may balance the number of objects to be processed according to speed and accuracy of system |
| Navigation is not accurate or hard to use | 4 | It's another main feature of the project but we may still give information about environment thus it may require to reassess requirements of the project | We may use GPS or create a model of location to increase navigation accuracy |

## 4.3. Project Plan

Work division by module for implementation:

Table 3: Work Division Table

| | |
|---|---|
| Mustafa Azyoksul | User Interface |
| Onur Mermer | User Interface |
| A A M Jubaeid Hasan Chowdhury | Object detection and text-to-speech using Google Cloud |
| Derviş Mehmed Barutcu | Navigation with symbols |
| Cemil Şişman | Depth sensing |

For report writing all the members will be responsible for each report. Responsibility for individual members will vary depending on the report.

## 4.4. Ensuring Proper Teamwork

As team EyeSight, we value teamwork greatly and spend effort in making use of everyone's experiences as much as possible. From reports to implementation, we assigned the workload taking into consideration team members' previous work experience and tried to achieve the out team's highest potential. Everyone participates taking essential decisions and making high level design choices, then whoever is more experienced in that area takes over the rest of the work.This experience consideration, gives us more chance to implement new and improved features and decreases the total amount of time this project will take without changing the workload. Additionally, we divided the work in a way that everyone gets more or less the same workload. Doing this achieves team cohesion and guarantees everyone works with maximum motivation.

## 4.5. Ethics and Professional Responsibilities

EyeSight is planned to be a new eye for visually impaired people. Therefore, it is extremely important to minimize the margin of error and make the app work at a significant success rate. The users of the EyeSight will heavily depend on the app to fulfill some of their daily life routines. That's why it's our first priority to provide as good of a user experience as possible. It's also very important because any error from the app might lead to undesirable circumstances because visually impaired people are fragile to the obstacles. It's our professional responsibility to ensure that they don't get injured or get lost because of EyeSight.

## 4.6. New Knowledge and Learning Strategies

In this project, because our main job is processing images from video inputs, we first need to learn image processing techniques and image analysis from online sources. Udemy and Youtube will be our main sources and the bug fixes processes we will get help from StackOverFlow.

EyeSight will be an Android application, hence we need to learn Android Studio. For writing android applications, Java programming language is used, however the environment is different and requires more time to get used to and learn new functions. To learn more about Android language Java and Android Studio we will use online courses on Youtube. One of our members, Mustafa, has past experience on writing Android applications so with his guidance in this part, we can quickly overcome this.

Lastly, we will use Google's API to overcome some of the loads, therefore, implementing cloud connections and using them in our project we will get help from Google's online courses to learn this technique and use them in our project.

# 5. References


[1]: “Seeing AI App from Microsoft,” *Seeing AI App from Microsoft*. [Online]. Available: https://www.microsoft.com/en-us/ai/seeing-ai. [Accessed: 20-Mar-2020].

[2]: Braille Works, “5 of the Best Mobile Apps for the Blind or Visually Impaired,” *Braille Works*, 27-Jun-2019. [Online]. Available: https://brailleworks.com/5-top-mobile-apps-for-the-blind/. [Accessed: 21-Mar-2020].

[3]: Woodill, G., 2020. *9 Outdoor Navigation Apps For People With Visual Impairments | Float*. [online] Float. Available at: <https://gowithfloat.com/2016/10/9-outdoor-navigation-apps-people-visual-impairments/> [Accessed 23 March 2020].

[4]: “Download Android Studio and SDK tools ： Android Developers,” *Android Developers*. [Online]. Available: https://developer.android.com/studio. [Accessed: 23-Mar-2020].

[5]: “TensorFlow,” *TensorFlow*. [Online]. Available: https://www.tensorflow.org/. [Accessed: 22-Mar-2020].

[6]: “Cloud Text-to-Speech - Speech Synthesis | Google Cloud,” *Google*. [Online]. Available: https://cloud.google.com/text-to-speech. [Accessed: 20-Mar-2020].

[7]: “Cloud Speech-to-Text - Speech Recognition | Google Cloud,” *Google*. [Online]. Available: https://cloud.google.com/speech-to-text. [Accessed: 20-Mar-2020].