DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
THE UNIVERSITY OF TEXAS AT ARLINGTON

ARCHITECTURAL DESIGN SPECIFICATION
CSE 4316: SENIOR DESIGN I
SPRING 2019

EYETYPE
EYE-TRACKER SOFTWARE

MATTHEW CRUM
TONYTAM DINH
RAGHAD SAFAULDEEN
MARCUS MALDONADO
SHAILI BALAMPAKI

Revision History

Revision	Date	Author(s)	Description
0.1	03.27.2019	MC, TD, RS, MM, SB	document creation
0.2	04.05.2019	MC, TD, RS, MM, SB	complete draft of ADS
0.3	04.06.2019	MC	ADS diagram
1.0	04.06.2019	RS, MC	improvements on ADS
1.1	04.07.2019	MC, TD, RS, MM, SB	presentation and final ADS

CONTENTS

List of Figures

List of Tables

1 INTRODUCTION

The EyeType team is creating an unprecedented state-of-the-art eye tracking software package. The software will map coordinates of the gaze vector of a userâs eye, sent from eye-tracking hardware, to a monitor. Users will be able to communicate through the GUI, surf the web, and control their environment with just the use of their eyes. The purpose of the software is to provide an easy-to-use, affordable, and sustainable eye-tracking system that can be customized to the needs of anyone. The eye-tracker software will be made to work with the hardware being created by another team, The EyeT Guys. However, it will be an independent software package that will be able to work with other similar hardware systems.

Initially, the software will be tailored specifically to work with the hardware teamâs system for Marcie. After that is complete, the independent software package, or the combined system with the hardware, may be made available commercially: the individual software package to companies with eye-tracking hardware, or the system directly to people afflicted with similar disabilities as Marcie.

The main function of the software will be its communicative functions. With the provided eye tracker, the user should be able to comfortably navigate the user interface to generate text. The user will then have an additional text-to-speech option after the text has been produced. Features of this specific function should also include: predictive text, auto-correct, dwell-time calibration, text clear, word clear, and keyboard re-size. The user will also have the ability to begin calibration with simple eye gestures. Additionally, the software should allow the user to control the cursor in a regular desktop environment and access the systemâs web browser.

The main input to the software will be pupil coordinates provided by the eye-tracking hardware to interact with the screen. There will also be input from the camera, which will be used to initiate eye gesture calibration. The output of the system will include the visuals of the GUI, audio from the speaker for the text-to-speech functionality, and the signal from the IR emitter.

2 System Overview

The software will receive input from the gaze vector of the user's eye. The input will then be interpreted in the Control Layer as either screen coordinates or an eye gesture. The user then can interact with the GUI screens in the Presentation Layer, or choose to use basic OS functionality with a mouse pointer in the Desktop Layer. If the keyboard GUI is being used, the Presentation Layer interacts with the Communication Layer with text as input, and then the text gets translated to speech and is output through the Transmission Layer as audio from the speakers. Finally, if the IR remote GUI is being used, the IR Module will interpret the input and send the output IR signal through the Transmission Layer to the IR emitter.

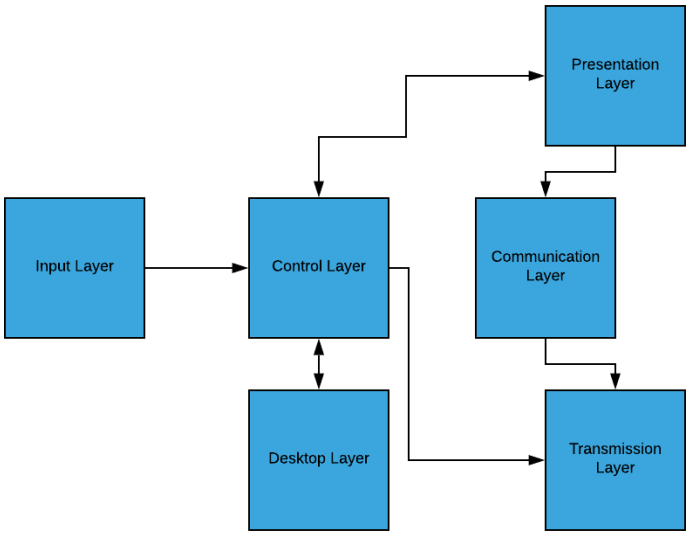

Figure 1: System Overview Diagram

2.1 Control Layer Description

The control layer is the heart of the architectural design of the application. The layer utilizes various subsystems to monitor and control the flow of data within the larger system in order for the other major layers to work properly together. It consists of the subsystems: Input Controller, Calibration, Gesture Processing, Main Program, Pause/Start Module, and IR Module. The Input Controller serves as the middleman between the Input Layer and the Control Layer, receiving the pupil data and sending it on to other subsystems in the Control Layer. The Calibration subsystem serves to fine tune the eye tracking experience and ensures that the user has accurate control of the GUI. The Gesture Processing subsystem receives data from the Input Controller to track the dynamic pupil ellipse for gesture recognition. The Main Program is the entry point of the Control layer and directs all related functions. The Pause/Start module serves give the user control over when he/she wants to use dwell time as a registered click. The IR module serves as the delegate between the Control layer and the Transmission Layer.

2.2 Input Layer Description

This layer is one of the layers that interacts with the user directly. It takes input from the user in the form of screen coordinates and eye gestures. Since the only kind of input our customer (an ALS patient) can give us is through eye gestures and dwelling on a certain area or a button, we will not have any other kind of input. Hence, the subsystems of this layer are screen coordinates and eye gestures.

2.3 Communication Layer Description

The Communication Layer is responsible for handling text input made by the user. Its subsystem are interconnected to optimize correctness and usability for the user. The following subsystems are: input text, text-to-speech, auto correct, and predictive text. Input text records the keyboard interface data. Text-to-speech translates the text to audio data. Auto correct fixes commonly misspelled words. Predictive text gives suggestions of relevant words and phrases.

2.4 Desktop Layer Description

The Desktop layer functions as the interface between the user and normal operating system control. It consists of the the Mouse Pointer and Dwell Click subsystems. The Mouse Pointer subsystem lies at the heart of the Desktop layer and allows the user to use the data from the control layer to operate the stock OS mouse pointer. The Dwell Click subsystem enables the OS to recognize dwell time as a standard click in the desktop environment. These systems work together in the Desktop layer to allow the user access to the computer outside of the eye tracking GUI.

2.5 Transmission Layer Description

The Transmission Layer's main priority is to transfer data generated by the program to it's physical counterpart. The two main subsystem this represents is the Audio Output and IR Code Processor. The former relays Speech-To-Text data to speakers and the latter communicates infrared control data to a infrared transmitter.

2.6 Presentation Layer Description

The Presentation layer includes many subsystems such as Camera Feed, Main Menu GUI, Keyboard GUI, IR Remote GUI, and GUI Interaction. Team EyeType attempting to trade-off those subsystems against others because they will be shown on the patient screen and from those subsystems the patient can choose to go the next in more details. For example, when the patient select the IR Remote GUI it will show the details of the remote that control the light, TV, and other useful staff that the patient will needs. The light control is to switch the light on and off. The TV control will be similar to the regular TV control. It is include many button such as the power button voice level and alike. Next are the details about each subsystem.

3 Subsystem Definitions & Data Flow

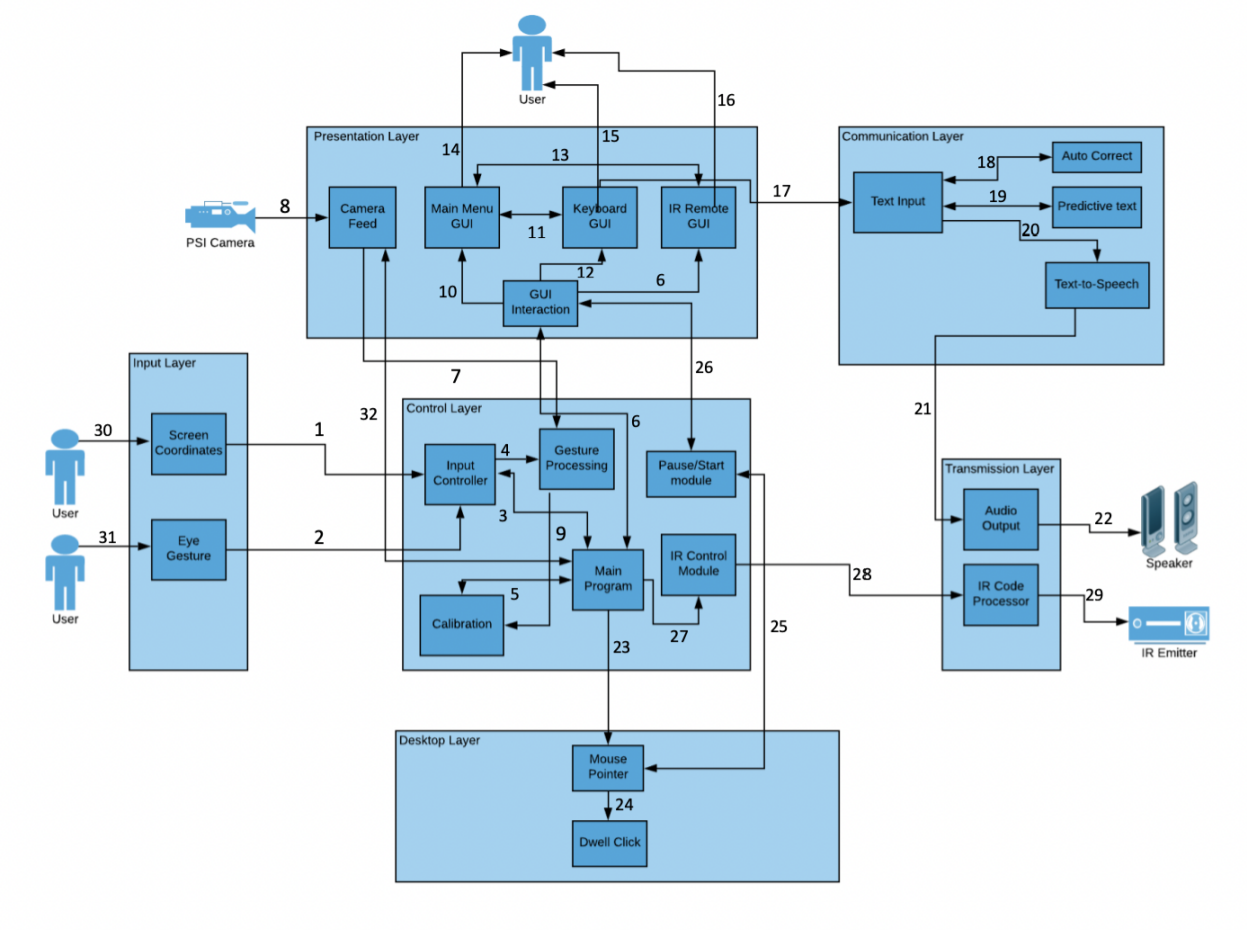

Figure 2: Data Flow Diagram

4 Control Layer

The control layer is the heart of the architectural design of the application. The layer utilizes various subsystems to monitor and control the flow of data within the larger system in order for the other major layers to work properly together. It consists of the subsystems: Input Controller, Calibration, Gesture Processing, Main Program, Pause/Start Module, and IR Module. The Input Controller serves as the middleman between the Input Layer and the Control Layer, receiving the pupil data and sending it on to other subsystems in the Control Layer. The Calibration subsystem serves to fine tune the eye tracking experience and ensures that the user has accurate control of the GUI. The Gesture Processing subsystem receives data from the Input Controller to track the dynamic pupil ellipse for gesture recognition. The Main Program is the entry point of the Control layer and directs all related functions. The Pause/Start module serves give the user control over when he/she wants to use dwell time as a registered click. The IR module serves as the delegate between the Control layer and the Transmission Layer.

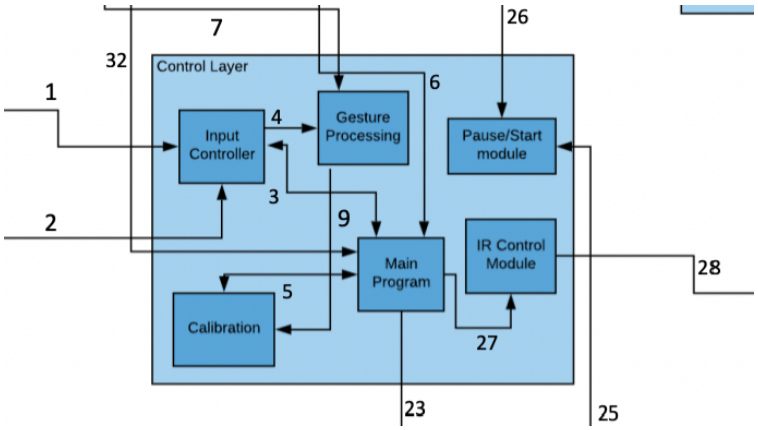

Figure 3: Control Layer Diagram

4.1 Input Controller

The input controller subsystem takes the screen coordinates from the user and translates the inputs. It may need to coordinate with the main program and calibration subsystems to achieve optimal accuracy and precision.

4.1.1 Assumptions

The camera feed is correctly positioned on the user's eye and the input controller is receiving screen coordinates.

4.1.2 Responsibilities

The input controller subsystem handles the main data flow between the screen coordinate and the main program. It also gives output data for gesture processing and handles offset data from the main program and calibration to optimize accuracy and precision.

4.1.3 Subsystem Interfaces

Table 2: Input Controller interfaces

ID	Description	Inputs	Outputs
#03	Sends coordinates to Main Program	User Coordinates	Mapping Data
#04	Send pupil location to gesture processor	Eye Ellipse	Gesture Data

4.2 Calibration

The calibration subsystem exists to guarantee the the input controller is accurately mapping the coordinates to the main program.

4.2.1 Assumptions

The subsystem is receiving eye coordinates and has a viewable eye ellipse from the camera feed.

4.2.2 Responsibilities

The subsystem will ask the user to look at specific target points within the screen. After comparing what user is looking at to the target point, the calibration subsystem will communicate to the main program the offset of the actual point to the theoretical point.

4.2.3 Subsystem Interfaces

Table 3: Calibration interfaces

ID	Description	Inputs	Outputs
#05	Corrects Coordinate Mapping	Translation Data	Coordinate offset
#09	Corrects Gesture Mapping	Gesture Data	Gesture Offset

4.3 Gesture Processing

This is the subsystem under control layer that handles all the gestures. It processes the gestures and calibrates it into useful commands for our software. It gets input from camera feed and input controller through eye gestures.

4.3.1 Assumptions

It processes the gestures correctly.

4.3.2 Responsibilities

The responsibilities of this subsystem is to take input from the camera feed and translate the gestures correctly into the system for calibration. Another responsibility is to take eye gestures from the input controller and process them.

4.3.3 Subsystem Interfaces

Table 4: Gesture Processing interfaces

ID	Description	Inputs	Outputs
#04	process eye gestures	eye gestures from input controller	ready to calibrate
#07	process eye gestures	gestures from camera feed	ready to calibrate

4.4 Main Program

This subsystem is the center of the control layer. It gets input from camera feed, calibration, input controller and GUI interaction. It analyzes the input from each of the subsystem and process it to the IR control module. Data also flows from the main program to the mouse pointer in the desktop layer.

4.4.1 Assumptions

Data from all the above mentioned subsystems are processed successfully.

4.4.2 Responsibilities

This subsystem allows our software to pass the data to GUI interaction which in turn sends it to the Main Menu GUI enabling the user interface of our software. This is then directly connected to the user. Also, the data provided to the IR Control Module is in turn transferred to IR Code Processor of the Transmission Layer.

4.4.3 Subsystem Interfaces

Table 5: Main Program interfaces

ID	Description	Inputs	Outputs
#32	data processing from camera feed	camera feed	data for IR control module
#05	analyzing calibration	N/A	mouse pointer co-ordination
#03	processing input from input controller	N/A	data for IR module
#06	enabling GUI interaction	N/A	data for main menu GUI

4.5 Pause/Start Module

This subsystem allows the user to enable or disable the dwell time click.

4.5.1 Assumptions

It is assumed that the input layer is functioning properly.

4.5.2 Responsibilities

The module is responsible for giving the user control over dwell time click functionality, giving way for the ability to switch this capability on or off.

4.5.3 Subsystem Interfaces

Table 6: Pause/Start Module interfaces

ID	Description	Inputs	Outputs
#26	Interaction with the GUI and pause/start button	user interaction	enables mouse pointer
#25	enabling and disabling mouse pointer	user input	related movement of the mouse pointer

4.6 IR Control Module

This subsystem works with the main program and IR Remote GUI of the presentation layer in order to generate data for an IR emitter.

4.6.1 Assumptions

Codes for specific environmental infrared devices are already known to the user.

4.6.2 Responsibilities

The user will input specific codes for infrared devices within the IR Remote GUI of the presentation layer. This will specify which device the IR will control. The IR Control Module will then generate the necessary data for the particular device to pass to the IR Code Processor. This will allow for the main program to control a connected IR emitter.

4.6.3 Subsystem Interfaces

Table 7: IR Control Module interfaces

ID	Description	Inputs	Outputs
#28	Description of the interface/bus	IR command	Generated IR Data

5 Input Layer

This layer is one of the layers that interacts with the user directly. It takes input from the user in the form of screen coordinates and eye gestures. Since the only kind of input our customer (an ALS patient) can give us is through eye gestures and dwelling on a certain area or a button, we will not have any other kind of input. Hence, the subsystems of this layer are screen coordinates and eye gestures.

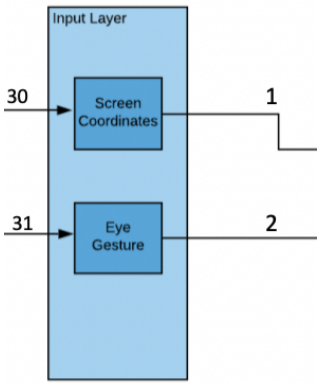

Figure 4: Input Layer Diagram

5.1 Screen Coordinates

This subsystem is one of the categories of input layer. The movement of the user's eye on the screen can be captured as coordinates and mapped into certain instructions. Different type of coordinates can be assigned to different commands.

5.1.1 Assumptions

The coordinates are mapped correctly into the screen.

5.1.2 Responsibilities

This subsystem will plot the coordinates of the gaze vector into the screen so that our software can convert them into specific commands.

5.1.3 Eye Gesture interfaces

Table 8: Screen Coordinates interfaces

ID	Description	Inputs	Outputs
#30	mapping gaze vector into coordinates	coordinates from the user	specific commands
#1	mapping screen coordinates into commands	screen coordinates	instructions for the input controller

5.2 Eye Gesture

This subsystem is one of the categories of input layer. The movement of the user's eye on the screen can be captured as eye gestures and converted into different commands.

5.2.1 Assumptions

Each eye gesture will be input correctly into the system.

5.2.2 Responsibilities

This subsystem will track the eye gestures of the user and classify them into different inputs.

5.2.3 Subsystem Interfaces

Table 9: Eye Gesture interfaces

ID	Description	Inputs	Outputs
#31	mapping eye movements into classified eye gestures	eye movement	eye gestures
#2	analyzing the eye gesture	eye gesture	commands for the input controller

6 Communication Layer

The Communication Layer is responsible for handling text input made by the user. Its subsystem are interconnected to optimize correctness and usability for the user. The following subsystems are: input text, text-to-speech, auto correct, and predictive text. Input text records the keyboard interface data. Text-to-speech translates the text to audio data. Auto correct fixes commonly misspelled words. Predictive text gives suggestions of relevant words and phrases.

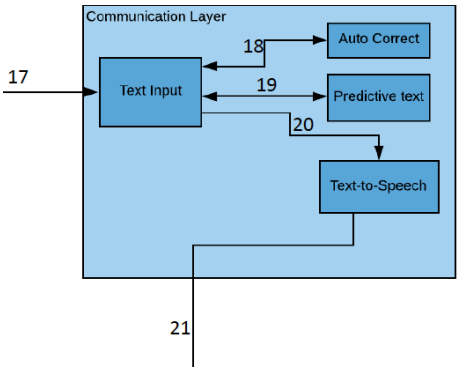

Figure 5: Communication Layer Diagram

6.1 Text Input

This subsystem records the text input from the keyboard interface. There will be persistent bi-directional interaction between this subsystem and other subsystems within this layer.

6.1.1 Assumptions

The input text can be modified and changed by other subsystems.

6.1.2 Responsibilities

This subsystem records the input text created alongside the keyboard interface. It will update the current text based on user input and modify the text if other subsystem requires it.

6.1.3 Subsystem Interfaces

Table 10: Text Input interfaces

ID	Description	Inputs	Outputs
#20	Input text to audio conversion	Interface Input	Speech Data
#18	Input text is compared to a dictionary	Auto Correct Data	Updated Text
#19	Input text suggestions	Predictive Data	Updated Text

6.2 Predictive Text

This subsystem will present users with commonly used words and phrases based on the current text input. The user will have the ability to choose the presented options if it fits their needs.

6.2.1 Assumptions

This subsystem assumes that text is partially inputted in order to suggest common words and phrases.

6.2.2 Responsibilities

After reading in current text input of the user, the subsystem will list commonly used words and phrases. If the user chooses one of the presented options, the subsystem will update the current text input.

6.2.3 Subsystem Interfaces

Table 11: Predictive Text interfaces

ID	Description	Inputs	Outputs
#19	Bi-Directional Text Transfer	Input Text	Updated Text

6.3 Auto Correct

After reading in current text input of the user, the subsystem will compare the input with a dictionary. It will then update the current input text if it contains a commonly misspelled word.

6.3.1 Assumptions

This subsystem assumes that a word is already completely typed and separated by white space before auto correcting.

6.3.2 Responsibilities

The subsystem will be constantly reading the input text. After a word is completed, it will compare the word to a dictionary. If the word is found to be incorrect, then it will update the current input with the correct word.

6.3.3 Subsystem Interfaces

Table 12: Auto Correct interfaces

ID	Description	Inputs	Outputs
#18	Bi-Directional Text Transfer	Input Text	Updated Text

6.4 Text-To-Speech

Upon user request, the following text data in the input text subsystem will be transferred to audio data.

6.4.1 Assumptions

This assumes that the text inputted from the user is completed, and that the user intends for the completed text to be recited.

6.4.2 Responsibilities

Completed text input will be converted to audio data. The audio data will be transferred to an audio output such as a speaker.

6.4.3 SUBSYSTEM INTERFACES

Table 13: Text-to-Speech interfaces

ID	Description	Inputs	Outputs
#20	Text input to Audio Output	Input Text	Audio Output

7 Desktop Layer

The Desktop layer functions as the interface between the user and normal operating system control. It consists of the the Mouse Pointer and Dwell Click subsystems. The Mouse Pointer subsystem lies at the heart of the Desktop layer and allows the user to use the data from the control layer to operate the stock OS mouse pointer. The Dwell Click subsystem enables the OS to recognize dwell time as a standard click in the desktop environment. These systems work together in the Desktop layer to allow the user access to the computer outside of the eye tracking GUI.

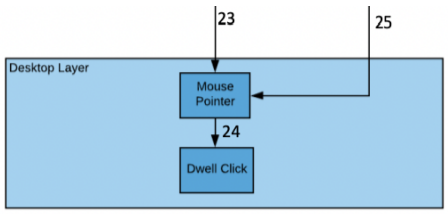

Figure 6: Desktop Layer Diagram

7.1 Mouse Pointer

This layer utilizes the pupil data translated in the control layer to operate the stock OS mouse pointer.

7.1.1 Assumptions

The raw pupil data received in the control layer has been properly translated.

7.1.2 Responsibilities

This layer is responsible for using the translated pupil data to control the OS mouse pointer.

7.1.3 Subsystem Interfaces

Table 14: Mouse Pointer interfaces

ID	Description	Inputs	Outputs
#23	Main Program activates Mouse Pointer	Main Program	N/A
#24	Dwell-click functionality can be toggled	N/A	Dwell Click
#25	Pause or start dwell-click functionality	Enable/Disable	Standard click

7.2 Dwell Click

This layer serves to enable the OS to recognize dwell time as a standard mouse click.

7.2.1 Assumptions

The Mouse Control layer is properly functioning.

7.2.2 RESPONSIBILITIES

Working in conjunction with the Mouse Pointer layer, this layer translates dwell time to a standard mouse click.

7.2.3 SUBSYSTEM INTERFACES

Table 15: Dwell Click interfaces

ID	Description	Inputs	Outputs
#24	Receives click to enable/disable dwell	Mouse Click	N/A

8 Transmission Layer

The Transmission Layer's main priority is to transfer data generated by the program to it's physical counterpart. The two main subsystem this represents is the Audio Output and IR Code Processor. The former relays Speech-To-Text data to speakers and the latter communicates infrared control data to a infrared transmitter.

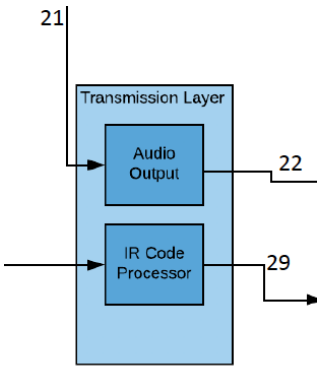

Figure 7: Transmission Layer Diagram

8.1 Audio Output

This subsystem relays the Speech-To-Text data to physical speakers so that the input text can be dictated.

8.1.1 Assumptions

The system has a connected output speaker system. The input received has already been converted to audio data.

8.1.2 Responsibilities

The subsystem will receive audio data from the Communication Layer's text-to-speech subsystem. The subsystem will then transmit the audio data to a physical speaker connected to the system's audio port.

8.1.3 Subsystem Interfaces

Table 16: Audio Output interfaces

ID	Description	Inputs	Outputs
#22	Audio data transmission output	Audio Data	Speaker Sound

8.2 IR Code Processor

This subsystem relays the infrared control data to infrared emitter so that different environmental systems can be controlled.

8.2.1 Assumptions

The system has a connected infrared emitter. The input received has already been converted to data the IR transmitter can handle.

8.2.2 RESPONSIBILITIES

The subsystem will receive IR data from the Control Layer's IR control module subsystem. The subsystem will then transmit the IR data to a physical infrared transmitter connected to the system.

8.2.3 SUBSYSTEM INTERFACES

Table 17: IR Code Processor interfaces

ID	Description	Inputs	Outputs
#29	IR data transmission output	IR Data	Infrared Signal

9 Presentation Layer

The Presentation layer includes many subsystems such as Camera Feed, Main Menu GUI, Keyboard GUI, IR Remote GUI, and GUI Interaction. Team EyeType attempting to trade-off those subsystems against others because they will be shown on the patient screen and from those subsystems the patient can choose to go the next in more details. For example, when the patient select the IR Remote GUI it will show the details of the remote that control the light, TV, and other useful staff that the patient will needs. The light control is to switch the light on and off. The TV control will be similar to the regular TV control. It is include many button such as the power button voice level and alike. Next are the details about each subsystem.

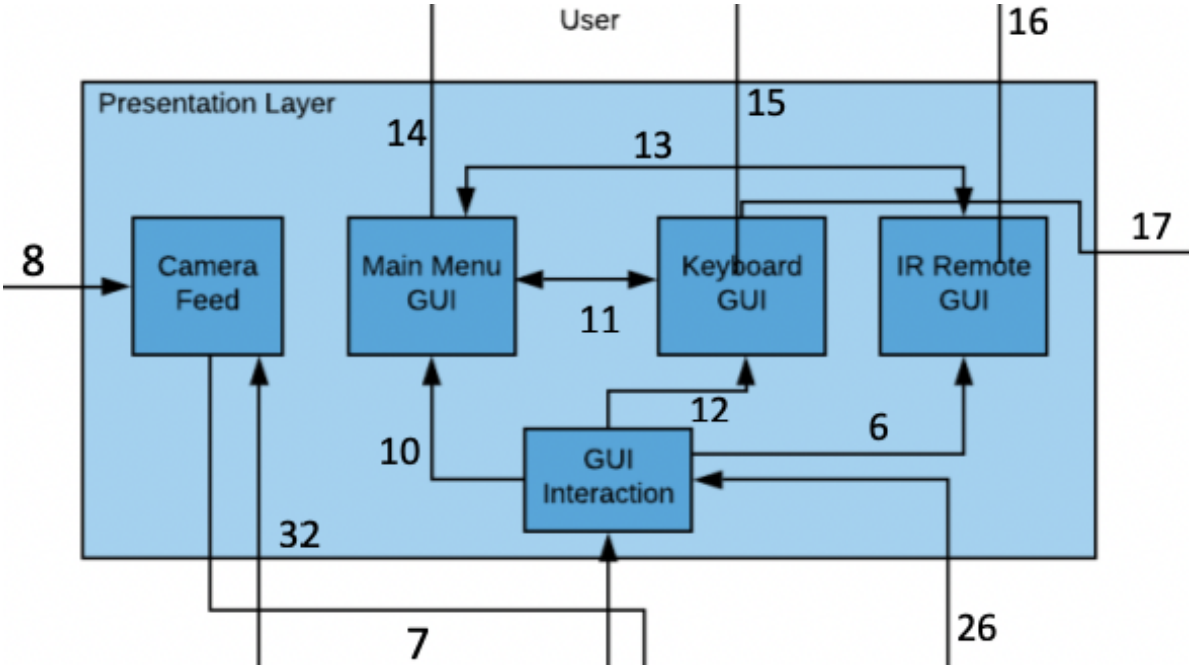

Figure 8: Presentation Layer Diagram

9.1 Camera Feed

The Camera Feed is to show to the user on his screen what the camera is getting as and input. In general it should shows the eyes of the patient on the screen of the computer. This subsystem should communicate with Control Layer and exactly with Main program and Gesture Processing subsystems. Also getting input from PSI camera.

9.1.1 Assumptions

Team EyeType assumes that camera feed will get the data from Control Layer and give the output to the display screen. We assume the Camera Feed will shows the patient's eyes and what they are looking at.

9.1.2 Responsibilities

The responsibility of Camera Feed subsystem is to getting an acceptable quality of data from PSI Camera and communicate with Main program and Gesture Processing subsystems. The other responsibility is to give nice view of user's eyes and shows where they are looking at as an output data will be shown on the display screen.

9.1.3 Subsystem Interfaces

Camera Feed subsystem is getting the incoming data from PSI Camera. Also communicate with Main program and Gesture Processing subsystems. Both subsystems are part of the Control Layer. And the outgoing data elements will be pass through Camera Feed interface then the data will be shown on the display screen.

Table 18: Camera Feed interfaces

ID	Description	Inputs	Outputs
#08	Camera Feed is getting data from PSI Camera	PSI Camera	N/A
#07	Communication with Control layer	Gesture Processing	Gesture Processing
#32	Communication with Control layer	Main program	Main program

9.2 Main Menu GUI

The Main Menu GUI subsystem is part of the Presentation layer. It is communicating with other subsystems inside the Presentation layer. It is getting the data flows from GUI Interaction, Keyboard GUI, and IR Remote GUI subsystems.

9.2.1 Assumptions

Team EyeType is assuming that the Main Menu GUI will be shown on the display screen. And we assume the Main Menu GUI will interact smoothly with Keyboard GUI and IR Remote GUI subsystems, and getting sufficient quality of data from the GUI Interaction subsystem.

9.2.2 Responsibilities

The Main Menu GUI's responsibility is to show the patient what to choose from the menu. Its other responsibilities is to getting data flow from the GUI Interaction subsystem and then display the result to the user.

9.2.3 Subsystem Interfaces

As other subsystems, Main Menu GUI has incoming and outgoing data. The incoming data is come from the GUI Interaction subsystem which is part of the Presentation layer. And for the outgoing data is the data that will be displayed to the patient's display screen. Also there are more connection between the subsystems in the same layer. So, Main Menu GUI is communicate with Keyboard GUI and IR Remote GUI subsystems.

Table 19: Main Menu interfaces

ID	Description	Inputs	Outputs
#14	Showing the output to the user	N/A	User
#13	Communicate with other subsystem in the same layer	IR Remote GUI	IR Remote GUI
#11	Communicate with other subsystem in the same layer	Keyboard GUI	Keyboard GUI
#10	Communicate with other subsystem in the same layer	GUI Interaction	GUI Interaction

9.3 KEYBOARD GUI

The Keyboard GUI subsystem is part of the Presentation layer. It includes different buttons that represent the letters. Also it will includes other useful buttons For example, Enter, Backspace, and more. Furthermore buttons with numbers 0-9 to write numbers. The sequence of the letters could be in qwerty or ABC order based on what the patient's selection.

9.3.1 ASSUMPTIONS

Team EyeType is assuming that the Keyboard GUI subsystem will be shown on the display screen. And we assume the Keyboard GUI subsystem will interact smoothly with Main Menu GUI subsystems, and getting sufficient quality of data from the GUI interaction subsystem. And giving the data flow to the text input subsystem that in Communication layer, and the data will be also displayed to the patient.

9.3.2 RESPONSIBILITIES

The Keyboard GUI's responsibility is to show the patient the type of keyboard that he/she wants. Its other responsibilities is to getting data flow from the GUI Interaction subsystem and then display the result to the user.

9.3.3 SUBSYSTEM INTERFACES

Keyboard GUI has incoming and outgoing data. The incoming data comes from the GUI Interaction subsystem which is part of the Presentation layer. And for the outgoing data is the data that will be displayed to the patient's display screen and other outgoing data to the text input subsystem in the Communication layer. Also there is connection between the subsystems in the same layer. So, Keyboard GUI subsystem is communicate Main Menu GUI subsystem.

Table 20: Keyboard interfaces

ID	Description	Inputs	Outputs
#15	Showing the data to the user	N/A	User
#17	Giving data to Communication layer	N/A	Text Input
#12	Getting data from subsystem in same layer	GUI Interaction	N/A
#11	Communicating with other subsystem in the same layer	Main Menu GUI	Main Menu GUI

9.4 IR Remote GUI

The IR Remote GUI subsystem is part of the Presentation layer. It is include different buttons with different functions. For example, there should be one button for power to turn the TV on and off. The voice level buttons, one to increase the voice level and other to decrease it. Also, some buttons with numbers 0-9 to choose channel, and other two buttons to just go to the next channel and go back to the previous one.

9.4.1 Assumptions

Team EyeType assumes the IR Remote GUI subsystem getting good quality of data as an input from GUI interaction subsystem. We assume that the IR Remote GUI will send a signal by the infrared waves to the TV.

9.4.2 Responsibilities

The responsibility for IR Remote GUI subsystem is to connect with the TV by passing an infrared waves to transfer the signal. Also getting input from GUI interaction subsystem and communicate with the Main Menu GUI subsystem in the Presentation layer.

9.4.3 Subsystem Interfaces

IR Remote GUI subsystem has incoming and outgoing data. The incoming data comes from the GUI Interaction subsystem which is part of the Presentation layer. And for the outgoing data is the data that will be passed to the TV by the infrared waves.

Table 21: IR Remote GUI interfaces

ID	Description	Inputs	Outputs
#16	Showing remote to the user	N/A	User
#06	Getting data from another subsystem in the same layer	GUI Interaction	N/A
#13	Communicate with other subsystem in the same layer	Main Menu GUI	Main Menu GUI

9.5 GUI Interaction

The GUI interaction subsystem is making connection between three layers. It is part of the Presentation layer and it is interacts with other subsystems in Control layer and Communication layer. Next sections will give more details about GUI Interaction subsystem.

9.5.1 Assumptions

Team EyeType assumes the GUI Interaction subsystem transmitting good quality of data to the Main Menu GUI, Keyboard GUI, and IR Remote GUI subsystems in Presentation layer. We assume that the GUI Interaction subsystem will exchange data from Main Program subsystem in the Control layer.

9.5.2 Responsibilities

The responsibility for GUI interaction subsystem is to communicate with Pause/Start Module subsystem in Communication layer. Also getting input from Main Program.

9.5.3 SUBSYSTEM INTERFACES

GUI interaction subsystem is like a ring between three layers. It is subsystem of Presentation layer. It has communication with Pause/Start Module subsystem in Communication layer and with Main Program subsystem in the Control layer. On the other hand, outgoing data to Main Menu GUI, Keyboard GUI, and IR Remote GUI subsystems in Presentation layer.

Table 22: GUI Interaction interfaces

ID	Description	Inputs	Outputs
#10	Giving data to other subsystem in the same layer	N/A	Main Menu GUI
#12	Giving data to other subsystem in the same layer	N/A	Keyboard GUI
#06	Giving data to other subsystem in the same layer	N/A	IR Remote GUI
#26	Communicate with Control layer	Pause/start module	Pause/start module
#6	Communicate with Control layer	Main program	Main program

REFERENCES