

GNG 1103  
FALL 2024  
November 24, 2024

# **DELIVERABLE H**

## **PROTOTYPE III AND CUSTOMER FEEDBACK**

Team DISMISS

GABRIELLE CHÉNIER  
RUSAFI KAMAL  
DARRIEN CHEN  
HANNAH KNIPE  
QUAN LUU

# Table of contents

---

Table of contents.....	2
Introduction.....	3
Prototype.....	4
Analysis.....	5
Prototyping test plan.....	6
Test results.....	8
Target specifications and bill of materials.....	11
Bill Of Materials.....	11
Updated Target Specifications.....	14
Conclusion.....	17

# Introduction

---

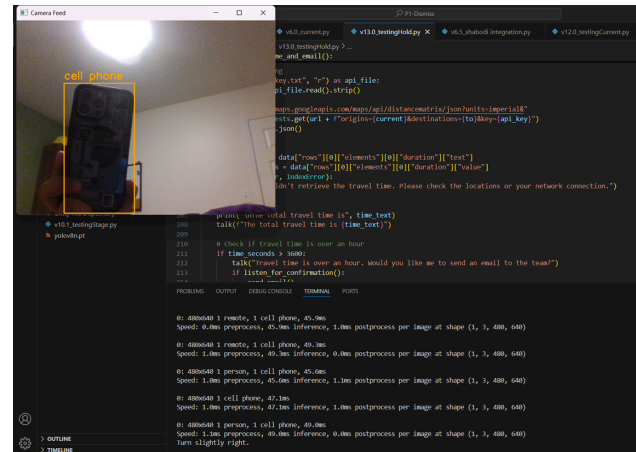
Visually impaired individuals have always faced challenges in navigating their environments and maintaining independence. Our smart glasses would address these concerns with features like object recognition, face detection and voice assistance. Those glasses would connect to an application to customize many features of the glasses. What sets our product apart is the use of Shabodi's APIs and offers customizable features such as customizable voice commands.

As part of the previous deliverables, we came up with the first and second prototype. The first one being a general design of what the final application could look like and how users could interact with it. This prototype allowed us to create a high-fidelity model. For the second prototype, we created three separate codes: voice assistance, object recognition and face detection with the help of different APIs. Finally, we'll create the last prototype, which is a combined program of all the codes of the previous prototype. We will then analyze all of the critical components, then we will define a test plan and gather all the feedback and comments received.

# Prototype

As planned, for this prototype, we have decided to combine all the focused subsystems we have been independently testing into one code. For Shabodi, we were finally able to generate tokens through their API. That let us finally start the actual implementation of the bandwidth API, which we previously planned to implement in our final product. Overall, this prototype aims to be an exact representation of what we wanted in our product, combined with the necessary APIs and changes which we had to implement midway. Below is a picture of how every result looks like in our product.

On the right we have our newly defined function which provides audio cues to the users to make sure they are warned of what is in front of them. This is just one of the iterations we have implemented in this prototype.



```
def invocation(access_token):
    conn = http.client.HTTPConnection("192.168.3.18", 7999)
    payload = json.dumps({
        "device": {
            "deviceId": 11
        },
        "maxBitRate": 400,
        "direction": "uplink",
        "duration": 10000
    })
    headers = {
        'Content-type': 'application/json',
        'Authorization': f'Bearer {access_token}'
    }
    conn.request("POST", "/qos/v1/bandwidth", payload, headers)
    res = conn.getresponse()
    data = res.read()
    print(data.decode("utf-8"))
    conn.close()
```

The figure on the left is the invocation function from Shabodi which lets us modify the maximum bandwidth available to our program/device.

The code on the right is the main code which keeps the whole program running. It gives a perfect overview of all the functions which have been combined together to work synchronously and collaboratively. It is all incorporated into Omega voice assistance which gives audible instructions to the user.

```
def run_omega():
    command = take_command()
    if command:
        print(f"Command received: {command}") # Only prints once for logging
        if 'time' in command:
            time = datetime.datetime.now().strftime('%I:%M %p')
            talk(f'current time is {time}')
        elif 'navigation' in command or 'route' in command:
            navigation_overview()
        elif 'play' in command:
            song = command.replace('play', '').strip()
            talk(f'Playing {song}')
            pywhatkit.playonyt(song)
        elif 'tell me about' in command:
            look_for = command.replace('tell me about', '').strip()
            if look_for:
                try:
                    info = wikipedia.summary(look_for, sentences=1)
                    talk(info)
                except wikipedia.exceptions.DisambiguationError:
                    talk("The topic is ambiguous. Please be more specific.")
                except wikipedia.exceptions.PageError:
                    talk("Sorry, I couldn't find information on that topic.")
                except Exception:
                    talk("An error occurred while searching for the topic.")
            else:
                talk("Please specify a topic to tell you about.")
        elif 'joke' in command:
            talk(pyjokes.get_joke())
        elif 'open camera' in command:
            open_camera()
        elif 'check destination' in command:
            check_travel_time_and_email()
        elif 'stop' in command:
            talk("Goodbye!")
            return False
        else:
            talk(f'Searching {command}')
```

# Analysis

---

With the prior prototypes, we were able to identify several ways to improve our final prototype. With our first prototype, the model of the application, we were able to determine how users would engage with the model and how to improve it. Overall, we received positive feedback on the organization and functionality of the application. With our second prototype, we created three separate codes: voice assistance, object recognition and face detection. This prototype enabled us to completely understand how our final product could benefit its users. As part of the deliverable, we will explain how both those prototypes allowed us to create our third prototype, a combined code of all the three different features of prototype 2.

In our previous prototype, we discovered a problem with our approach to this project, that is we were not being specific with our solutions. For instance, our project was not contributing to the functional design criteria in any way. For example, our object detection function was detecting objects but the users were not being benefited with it if they did not know where to go or what to do with those results. That's when we came up with a different approach to this problem which helps the user by giving direction-specific audio cues by which the visually impaired individuals will be benefitted, as well as contributing to the functional design of our project.

This prototype not only combines every feature but also is a result of iterations from our past prototypes which includes adding a better location API for ease of navigation, Shabodi's dynamic bandwidth API which reacts to the surroundings and adjusts the bandwidth, and even a better object and person detection feature which is better than our previous prototype.

# Prototyping test plan

---

Test	Objective	Description of test methods and materials needed	Description of results to be recorded	Estimated test duration	Stopping Criteria
1	We would like to test how well all the subsystems function together if they were merged into one single code. This will allow us to fix any software issues and make any adjustments required to improve the finished product.	Since the code may turn out to be software-heavy and require more processing power overall, we need access to a powerful computer. To test object recognition and face detection, we will use the faces of our team members and a variety of objects.	We will document how well each subsystem has adapted to the final code. One way to accomplish this would be to only use the voice assistance as a connection to the object recognition and face detection.	2 hours	We will continue until the test passes 80 percent of the time.
2	Seeing how the program would react to different conditions in the real world	Doing multiple tests in different locations with different users to see how the location API would work along with the object detection API.	We will document the overall accuracy of the program. We will make the users give it an accuracy rating out of 100 by determining if the prototype can be useful to us.	3 hours	We will keep on repeating this test until we reach a final value of above 75%.
3	Testing the final usability of the whole code	We will ask various individuals to evaluate this prototype and provide us with feedback so we	To determine our usability rating, we will ask the users to score the product out of 10, which we will then	2 hours	Realistically, we will do 5 tests but if it does not meet our criteria of over 70% we will continue until we've met

		can use them for our future development plan.	average out.		our criterion.
--	--	--	--------------	--	----------------

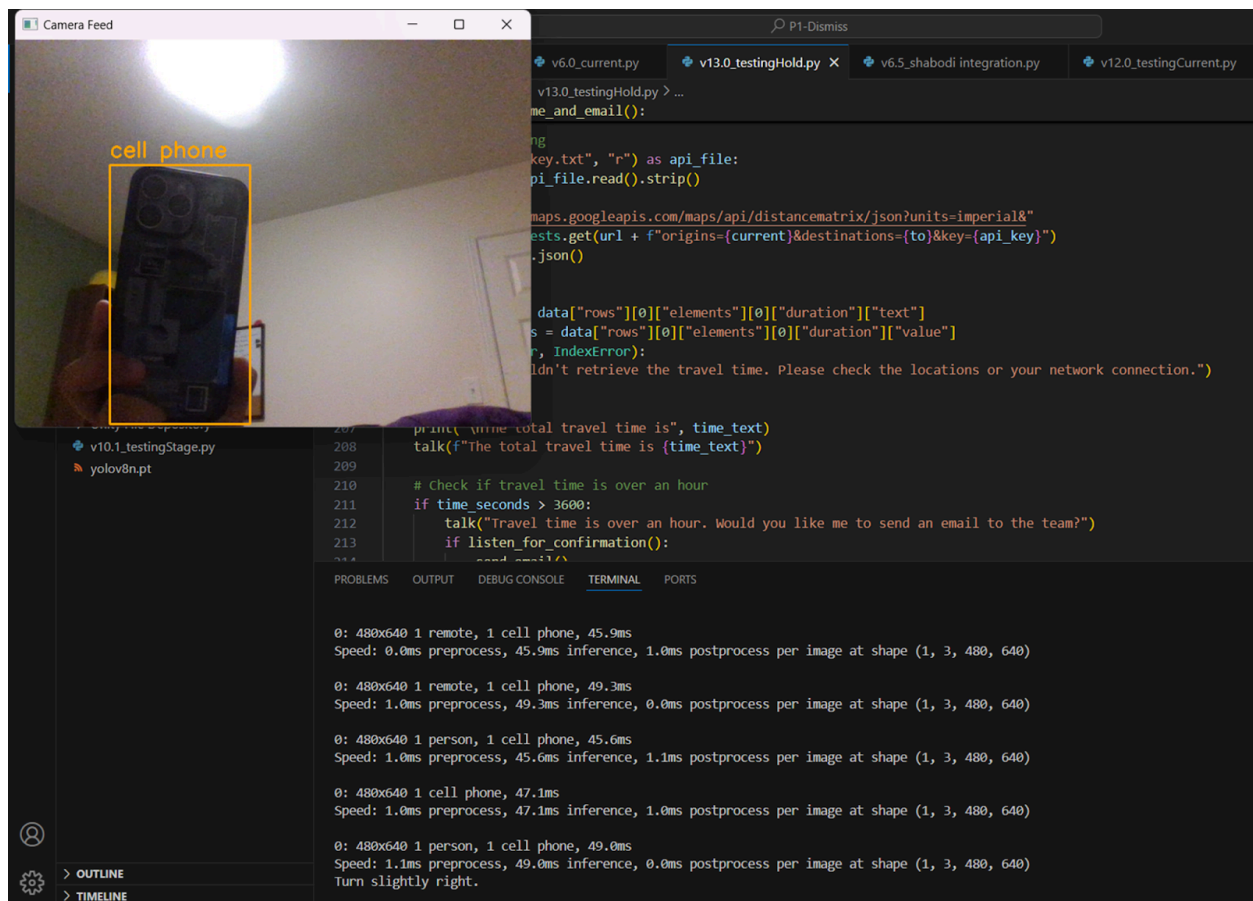
# Test results

## Test results 1

To test the object recognition and face detection features of this prototype, we used several objects and faces. Throughout the whole entire testing stage, we were able to obtain a high accuracy regarding the object recognition. Despite certain difficulties, like occasional errors in recognizing objects, it still performed admirable.

Again, we obtained a high accuracy rate in face detection. The program was able to detect nearly all of the faces shown. We did encounter some problems under different environments. For instance, the program occasionally struggled detecting faces in low lighting. The photo shown below is just one example of an object recognized by our program.

We also ran into an issue with cluttered backgrounds. If there were too many items in the background, our software would have difficulty distinguishing which item is which. For some potential cautions in the future, there could be bias labeling in objects. As if the software has not labeled the specific object it will not be able to distinguish it.





## Test results 2

Users	Pros	Cons	Additional comment or feedback
User 1	“It was impressive to see how the program got its location automatically and basically gave me all the navigation details to my destination. It really is hands-free!”	“Sometimes the audio cues would be delayed by a few seconds but it does the job.”	“There’s definitely room for improvement but overall the program works well.”
User 2	“I liked how the navigation also accounts for any obstacles which come in front of me by telling me to ‘turn right’ or ‘turn left’.”	“I really wish it gave me more updates regarding the route options, which seems to be limited.”	N/A
User 3	“The program was not only able to detect and recognize any objects in front of me but also help me navigate my surroundings, by warning	“The program sadly couldn’t adapt to my change in environment. I went from a well lit room to a darker one and it struggled to detect my face in the second environment.”	“Having a button to turn it off instantly could be a good option.”
User 4	“I was able to access other features such as emailing my team for not making it on time for my meeting. Overall it seemed awesome on how much this small program can do!”	N/A	N/A

### Test results 3

Users	Pros	Cons	Additional comment or feedback
<b>User 1</b>	“The voice assistance feature is very impressives and works well with the other features, like object recognition and face detection.”	“I sometimes had a bit of trouble having the voice assistance correctly understand me.”	N/A
<b>User 2</b>	“I found it impressive how every feature could be reached with the voice assistance. It could be a very useful feature for your main user.”	“The device couldn’t correctly identify every object shown. For example, it identified a phone as a water bottle.”	“This could be a game-changer for accessibility!”
<b>User 3</b>	“The voice assistance is super innovative and works well with the other features	“I just wished that you didn’t have to wait between each time you talk to the voice assistance feature”	N/A
<b>User 4</b>	“The object recognition detected and recognized all of the objects I showed. It also easily detected my face.”	“The voice assistance couldn't catch what I was saying or asking at some times.”	N/A

Overall, every test was successful. We received great feedback from potential users and were able to recognize several key aspects where our prototype needed improvement. Some possible points for improvement could be a better adaptability to different environments or a better microphone for the voice assistance feature. Users often felt as if the voice assistance feature couldn’t properly understand everything that was said or that the object recognition feature occasionally couldn’t properly identify objects shown.

# Target specifications and bill of materials

---

## Bill Of Materials

PROTOTYPE 3 - Bill of Materials					
Item #	Item Name	Quantity	Purpose and Description	Price	Amount
1	Figma	1	This software will help us create prototype images of outputs our software will give to the user based on the object recognized. We will create scenario-based images for example if this object is recognized, our user will see this	\$0.00	\$0.00
2	Visual Studio Code	1	To code all the APIs, libraries, and Figma. Into one final product.	\$0.00	\$0.00
Total product cost					\$23.00

DESIGN DAY- Bill of Materials					
Item #	Item Name	Quantity	Purpose and Description	Price	Amount
1	VR headset	1	To run our code and visualize the output	\$0.00	\$0.00
2	Glasses	1	A prop to display during design day to give an idea of what the HUD would look and feel like.	\$0.00	\$0.00

3	Trifold	2	To present our project	\$0.00	\$0.00
4	Karaoke machine	1	Has a speaker and microphone: To enhance volume in the crowd	\$0.00	\$0.00
5	Laptop 1	1	To always display an advertisement video and attract attention	\$0.00	\$0.00
6	Laptop 2	1	To always display an advertisement video and attract attention	\$0.00	\$0.00
7	Extension cord	1	To connect to the TV	\$0.00	\$0.00
8	Candy (caramel) Bowl	1	To attract attention and reward audience members who ask about us	\$2.00	\$2.00
9	Markers	1 pack	To create our trifold	\$3.75	\$3.75
10	Bristol Board	3	To create our trifold	\$1.00	\$3.00
11	Figma Pro membership	1	To convert our first prototype into a functional website	\$23.00	\$23.00
<b>Total product cost</b>					<b>\$31.75</b>

<b>BUDGET</b>	
<b>PROT 1</b>	<b>\$0.00</b>
<b>PROT 2</b>	<b>\$5.00</b>

<b>PROT 3</b>	<b>\$0.00</b>
<b>Design Day</b>	<b>\$31.75</b>
<b>Total</b>	<b>\$36.75</b>

## Updated Target Specifications

ID	Design Criteria	Original Values	Updated Values	Justification
1	Face Detection Reliability	Pass, 1m	Pass, 1.5m	Testing showed the need for an extended range to detect faces from farther distances.
2	Speech Recognition	Pass	Pass with noise	Enhanced to work effectively in noisy environments, addressing user feedback.
3	Object Recognition Reliability	>80%	>90%	Users highlighted the need for higher accuracy when recognizing objects.
4	Software and Device Compatibility	Many	Universal	Updated to ensure seamless compatibility across various devices and platforms.
5	Audio Instructions	Yes	Responsive	More accurate and responsive
6	Low Latency	<20ms	<15ms	Reduced latency to improve responsiveness in high-pressure scenarios.

7	Processing Power	1.6-2.5 GHz	>2.5 GHz	Enhanced to handle more complex tasks and faster computations.
8	Network Connection	5G	5G with fallback	Added fallback to ensure connectivity even in areas with weaker signals.
9	Display Resolution	1080p	1440p	Improved clarity for users requiring detailed visuals.
10	Obstacle Detection	Yes	Enhanced range	Updated to detect obstacles at a farther distance for improved navigation.
11	Object Recognition	Yes	Improved AI model	Refined the recognition model for better performance and wider object categories.
12	Facial Detection	Yes	Multi-face detection	Allows detection of multiple faces simultaneously, improving usability.
13	Plans Routes	Yes	Optimized routes	Made the finding route system working.
14	Knows Current Location	Yes	Enhanced precision	Increased GPS accuracy to within closer radius for precise localization.

15	Voice Control	Yes	Multi-voice	Enhanced to recognize multiple users or voices with greater accuracy.
16	Functional Microphone	<100dB	<95dB	Improved sensitivity to pick up quieter sounds and commands.
17	High Camera Quality	>12MP	>16MP	Increased resolution to capture clearer images for analysis.
18	Audio Warnings	Yes	More Accuracy	Added the option for users to customize audio alerts based on preferences.
19	Informs User of System Errors	Yes	Detailed Notification	Updated to provide clearer and more detailed error messages.



# Conclusion

---

As part of the previous deliverables we came up with the first and second prototype and received feedback from potential users. In this deliverable, we developed our final prototype, a combined code of all the codes from the previous prototype, analyzed all of the key components, devised our test plan, and gathered feedback and comments from potential users. In terms of future work, we are going to create a functional website and connect it to this prototype. The figma document. The first prototype would serve as the basis for this website for our Omega assistance AI. Creating a functional application for our Omega voice assistance and the other features will complete our product and make it accessible.