**Eye tracker**

Design of glasses:

1. Use of normal glasses and attaching an arm for the sensor support.
2. Designing or choosing a 3d model and printing the whole glasses with an arm support.

Methods for eye tracking:

1. Using 4 QTR-1A Reflectance Sensor placed at both sides, top and bottom of eye, we can determine the position of eyeball. But this would result in lack of visibility for the user.
   * <https://create.arduino.cc/projecthub/H0meMadeGarbage/eye-motion-tracking-using-infrared-sensor-227467>
2. Use of IR filter removed camera module to record eye movement. A pre trained model can be used to identify the position of the eye. But to do so we need to find a way to remove the pc requirement for running the model.

Finalization:

Method 2 is selected since it allows the user to see his surroundings much clearly.

Updates:

Base of eye tracking software is developed and looking forward for improvements. Refer the provided video for understanding the code:

<https://www.youtube.com/watch?v=-jFobb6ARc4&list=PLJ958Ls6nowUwRXHUcFwZy2CT0naMULR3>

(7/12/2021): Raspberry pi 4 is selected as microcontroller to remove the pc requirement for running the python model.

Improvements needed:

Tracker need to detect up and down movement of iris. Currently it detects only left, right and blink.

Update (7/12/2021): tracker is capable of detecting up and down movement of iris along with previous positions.

**Home Automation**

**Micro Controller:**

1. Arduino UNO
2. Raspberry Pi 4

Finalization: Raspberry Pi 4 have been selected as the microcontroller since it have better computational power and support complex models.

**Wheelchair**

**Node MCU**

**Simulations:**

Simulation can be done using protheus in raspberry pi 4.  
Refer the below video for more understanding on how to simulate camera detection using raspberry pi 4 in protheus:

<https://youtu.be/5uptP78_LAw>

**Eye Tracker Program Working:**

We use googles python library *mediapipe* for eye pupil detection. Different landmarks are tracked using this library. We select the landmarks around the eyes from *facemesh* (landmarks function for face). A horizontal and vertical line is drawn between the extreme sides and up and down of both eyes. The distance of these 2 lines are calculated using Euclidian Formulae.

*d = √ [(x22 – x11)2 + (y22 – y11)2]*

The position of the distance of the horizontal and vertical lines are used to estimate whether the eye blinked or not. When the ratio between horizontal and vertical line is greater than 4 the program detects the eye as blinked.

For position of pupils, the image of eye is gone through different filtration process and the binary image of the eye is obtained. From that image, the black pixels position indicate the position of pupil. In order to identify which zone the black pixels are, the horizontal line of eye is divided into 3 zones.

*hz = hd/3*

And vertical line of eye is divided into 2 zones.

*vz = vd/2*

Now the different zones are

Centre Zone = *[hz : hz + hz]*

Left Zone = *[0 : hz]*

Right Zone = *[hz + hz : hd]*

Upper Zone = *[vz : vd]*

Lower Zone = *[0 : vz]*

When the black pixels enters these zones we get the positons of the pupil.