ENHANCING USABILITY AND EXPERIENCE OF USER WHILE VIDEO STREAMING USING HAND GESTURES

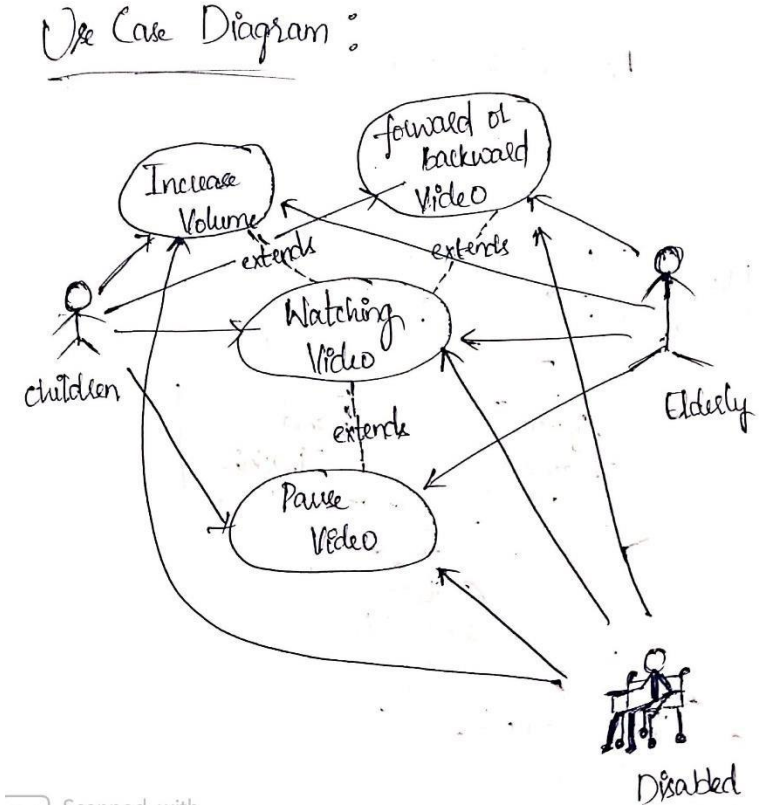
Abstract:

Current state of video streaming and watching Tv's is comfortable, but in this project, we aim to revolutionize the comfortability of the viewer. Using hand gestures has always been a part of communicating in our daily lives, we would like to use the same mode of communication to change the way a viewer interacts with the computer to watch movies or stream videos. In this project we aim to make the usability even higher and also, we will help some especially abled sections to have equal entertainment and involvement as normal people. We would like to make this experience available to everyone at an affordable rate, so keeping that in mind we plan on creating a cost-effective product. We hope that in the end we have a product, which will increase the usability and enhance the user experience in consuming video content. We will be using Arduino Uno (micro-controller) and some sensors. This project will change the current interaction method.

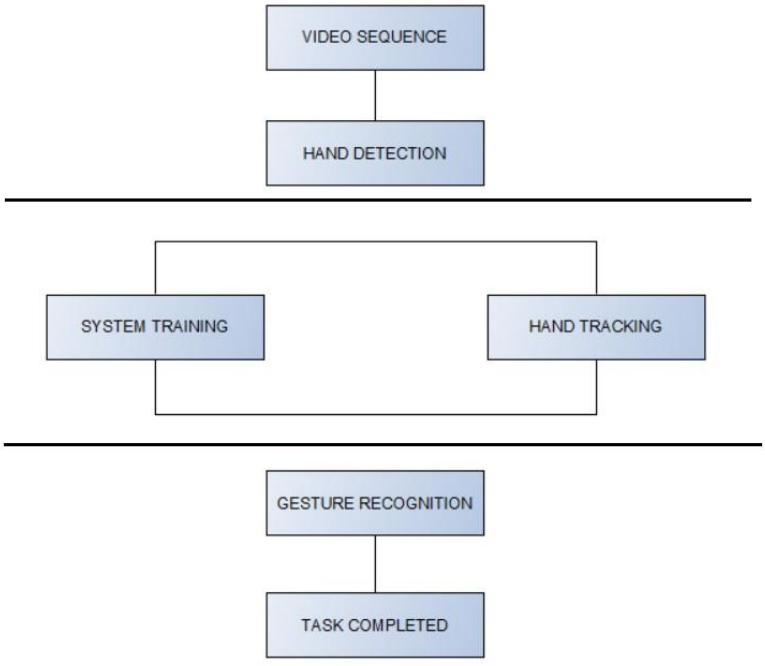
Objectives of the project:

1. Revolutionize the video streaming experience
    - High usability
    - Better interaction
    - User comfort
2. Help differently abled sections enjoy the movie watching experience better.
    - Physically disabled
    - Old age people
3. Provide cost – effective product
    - Low maintenance cost
    - Cheap purchase price
    - Durability 4. Learn new technologies.
    - Arduino Programming
    - Working with sensors
    - 3-D printing

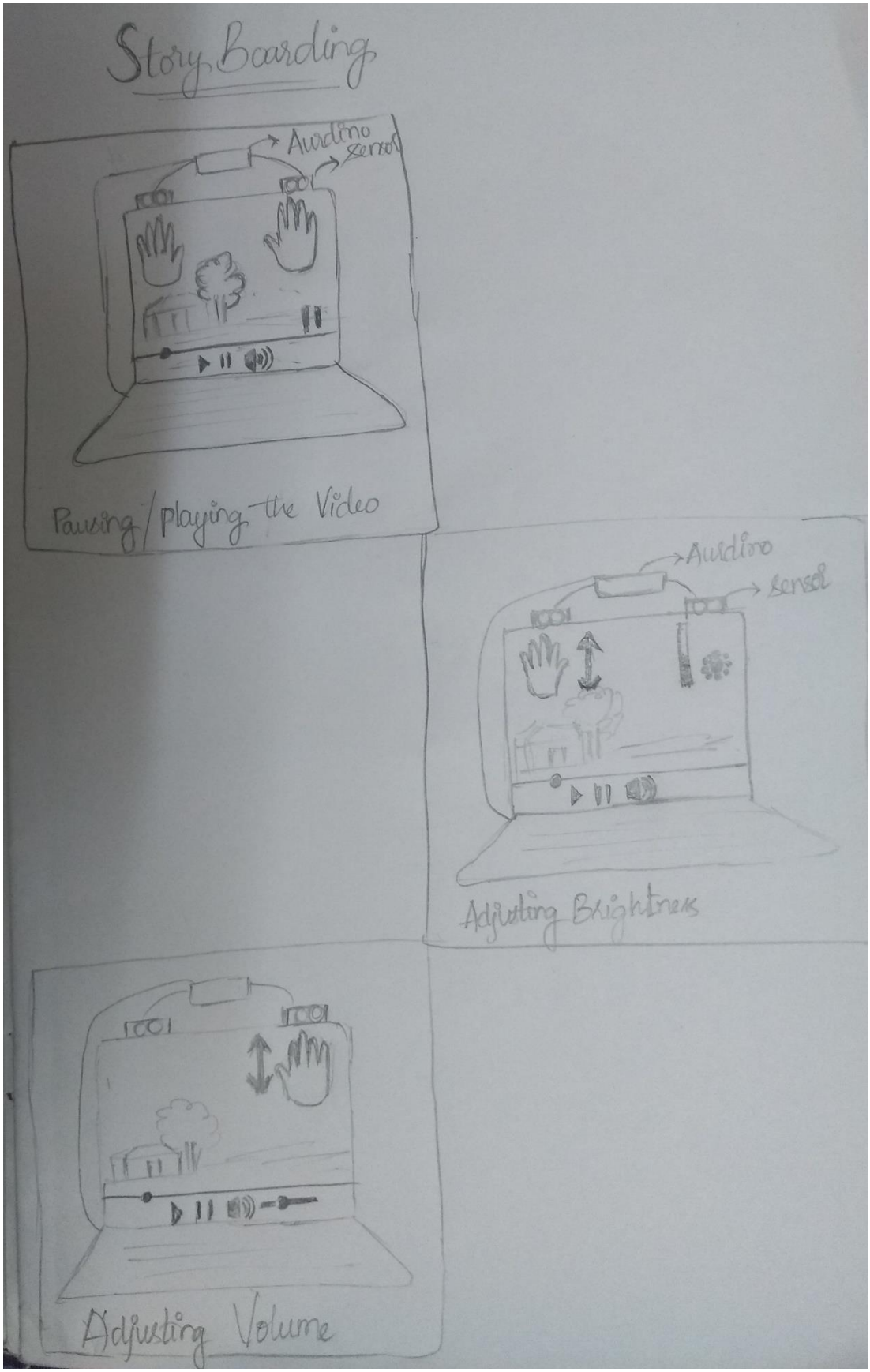
Use Case Diagram:



Hierarchical Task Analysis:



Story Boarding:

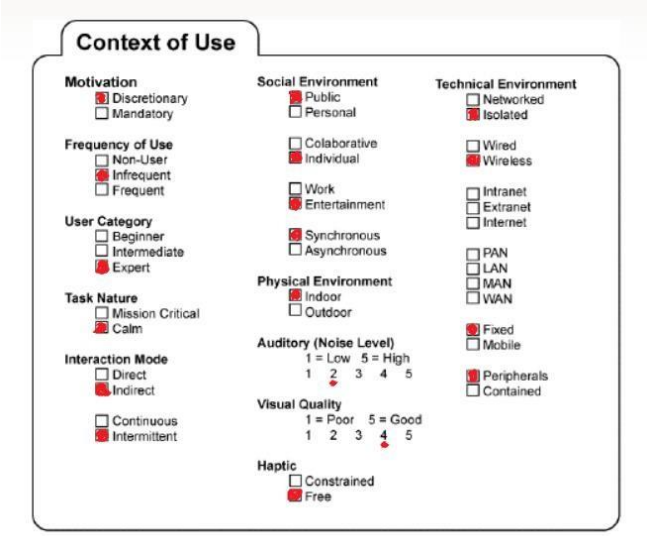


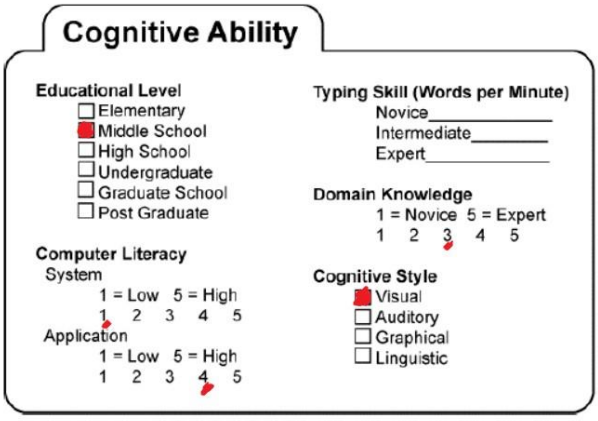
Stakeholder Profiles:

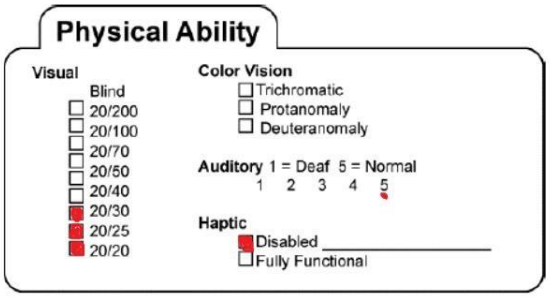
The three target clients are

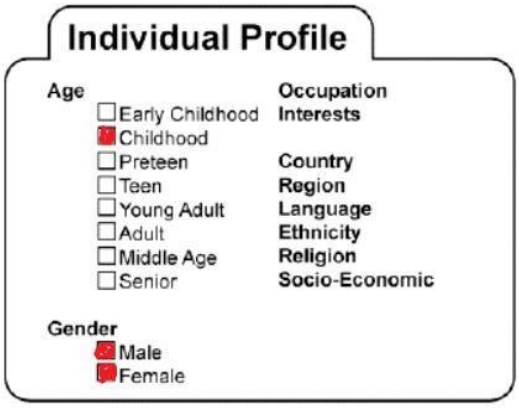
- Children Elderly
- Disabled

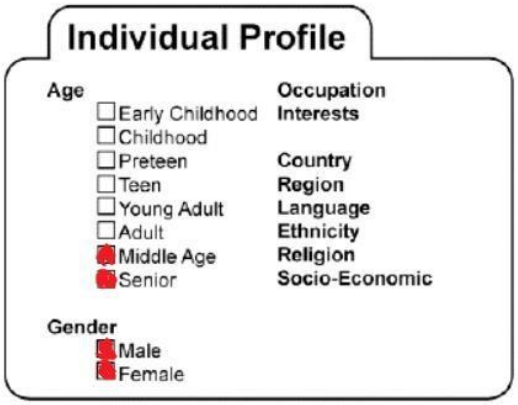
Elicitation technique used for observing for collecting requirements is Questionnaire (Google Form). This project will be very useful, children enjoy by controlling the system with hand gestures. Elderly can handle the without any efforts, Disabled people also can handle the system without moving, just use hand for task accomplishment.











Important Tasks:

- Aurdino Coding – Tough Task
- Python Coding – Moderate Task
- Hardware Assembling – Moderate Task
- 3-D Printing – Moderate Task

The two major tasks involved in our project are :-

1. Increasing / decreasing volume
2. Pausing a video

In design alternatives, we will compare the times taken by traditional methods and by using our product. Then we will compare the times and create a comparision table.

★ MHP :- (Model human processor)

1. Pausing a video :-

→ Traditional method :-

The user sits before a computer screen playing the movie. Whenever he/she wants to pause, s/he must press the space bar. Lets calculate the time between stimulus and response.

→ $T_p + T_c + T_m$

T_p → Time for perception (realizing they want to pause)
= 100ms

T_c → Time for recognition (realizing they have to press 'space' + finding where the 'space' is?)
= 70ms + 70ms

T_m → Time for action (pressing the 'space' key)
= 70ms

= 100 + 70 + 70 = ~~240ms~~ 310ms

→ Using our product:-

The user sits before the computer screen and watches a movie, whenever he/she wants to pause the video, s/he must raise both the hands above. Lets calculate the time between stimulus and response.

→ $T_p + T_c + T_m$

T_p → Time for perception (Realizing they want to pause)

= 100ms

T_c ⇒ Time for recognition (realizing they have to raise both their hands)

= 70ms

T_m → Time for action (rasing both hands)

= 70 ms

= 100 + 70 + 70 = 240 ms.

2. Increasing / decreasing volume:-

→ traditional method:-

The user sits before a computer screen watching a movie. whenever he/she wants to calculate the time between stimulus and response.

→ $T_p + T_e + T_m$

T_p → (realizing they want to want to inc/dec the volume)

= 100 ms

T_c → Time for recognition (realizing to press up/down arrow + finding where the arrow key is ?)

= 70ms + 70 ms = 140ms

T_m → Time for action (putting hands on keyboard + pressing the up/down key)

= 70 + 70ms = 140ms

= 100 + 140ms + 140ms

= 380 ms

→ <u>using our product :</u>

The user sits before the computer screen and watches a movie, whenever he/she wants to pursue the video, s/he must raise both the ~~hands above~~ left hand above. Lets calculate the time between stimulus and response

→ $T_p + T_c + T_m$

T_p → Time for perception (realizing they want to to inc/dec the volume)

= 100ms

T_c → Time for recognition (realising to move left hand up)

= 70ms

T_m → Time for action (moving the left hand up)

= 70ms

= 100 + 70 + 70 = 240ms

Key-Stroke Level Model (KLM)

(i) KLM for pausing the video:

General/traditional Method:

Description	operators
Move hand to keyboard	H
Searching space bar	M
clicking the space bar	K

Time, $t = H + M + K$

$= 0.4 + 1.35 + 0.2$

$t = 1.95$ secs

Using our product:

Description	operator
Raising hands towards sensors	H
Adjusting hands	M

Time, $t = H + M$

$= 0.4 + 1.35$

$t = 1.75$ secs

(ii) KLM for volume changing:

General/traditional Method:

Description	operators
Move hand to keyboard	H
Searching volume button	M
Increase or Decrease Volume	M
click the volume button	K

Time, t = H + M + M + K

= 0.4 + 2.7 + 0.2

t = 3.3 secs

Using our product:

Description	Operators
Raising hand towards sensor	H
increase & decrease volume	M
Adjusting hands	M

Time, t = H + M + M

= 0.4 + 2.7

t = 3.1 secs

→ GOMS - CMN

1. Pausing the video :-

Goal : PAUSE THE VIDEO

GOAL : RAISE LEFT HAND

operator : Think to raise left hand	M
operator : move left hand above	H
operator : Hold hand in air	H

GOAL : RAISE RIGHT HAND

operator : Think to raise right hand	M
operator : move right hand above	H
operator : Hold hand in air	H

* Selection rule for subgoals

The subgoals can be performed in arbitrary order

Considering an average user the timing required are :-

$$(M + H + H) * 2 = (1.35 + 0.40 + 0.40) * 2$$

$$= 2.15 \times 2$$

$$= 4.30 \text{ seconds.}$$

2. Increasing / decreasing the volume :-

Goal :- INC/DEC THE VOLUME

[Select GOAL: Increase the volume

operator :- deciding to change volume M

operator :- raising the left hand H

operator :- moving hand back nH

verify increase of volume M

GOAL : DECREASE THE VOLUME

operator :- deciding to change volume M

operator :- raising the left hand H

operator :- moving hand towards sensor nH

verify decrease of volume M]

* Selection rule :-

if (user wants to increase volume)

{ Goal 1 };

else if (user wants to decrease volume)

{ Goal 2 };

time for inc / dec volume is M+H+H+M = 1.35+0.4+0.4+1.35

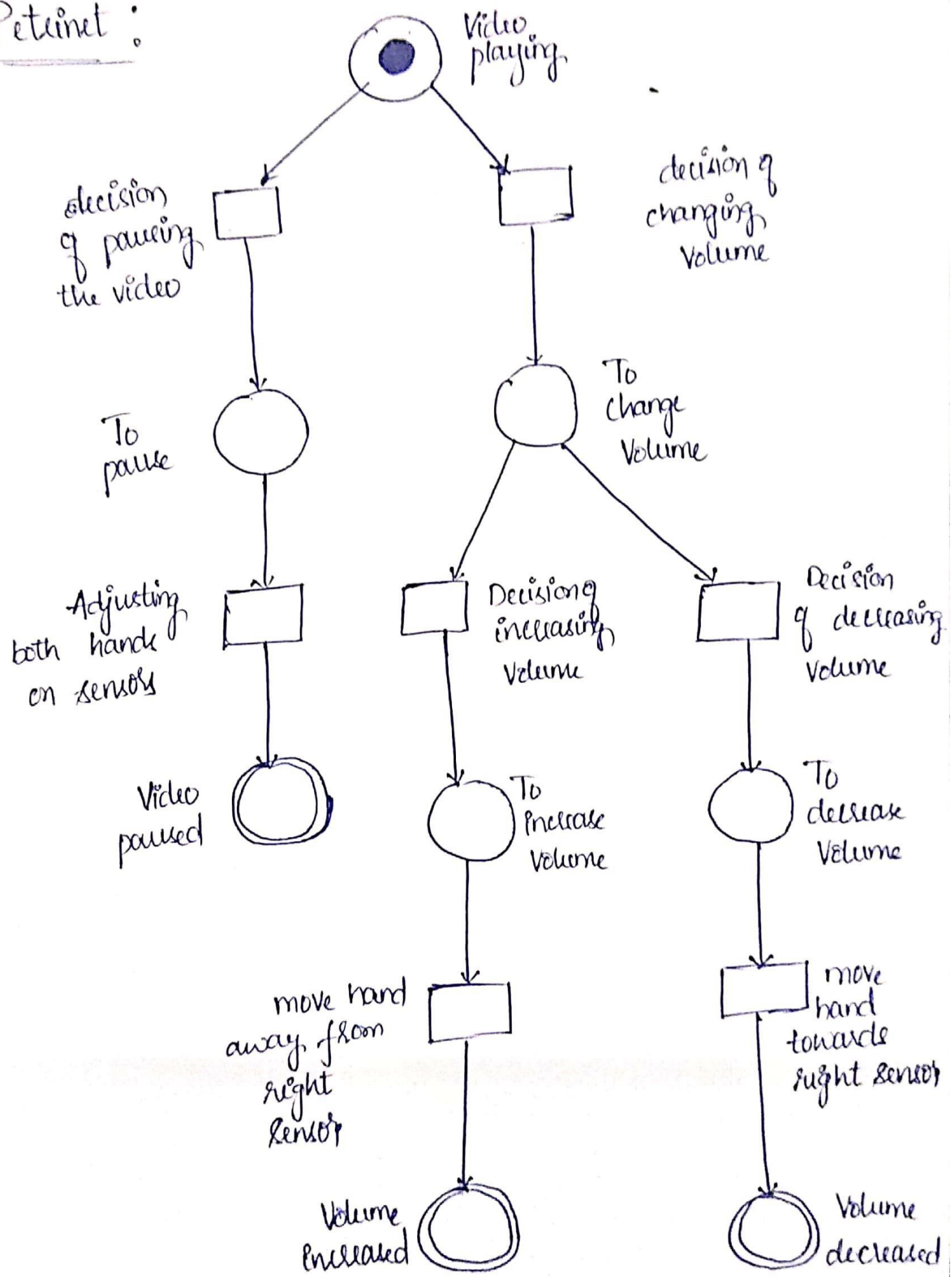
= 3.5 s

Empirical formula :- 2M+(n+1)H

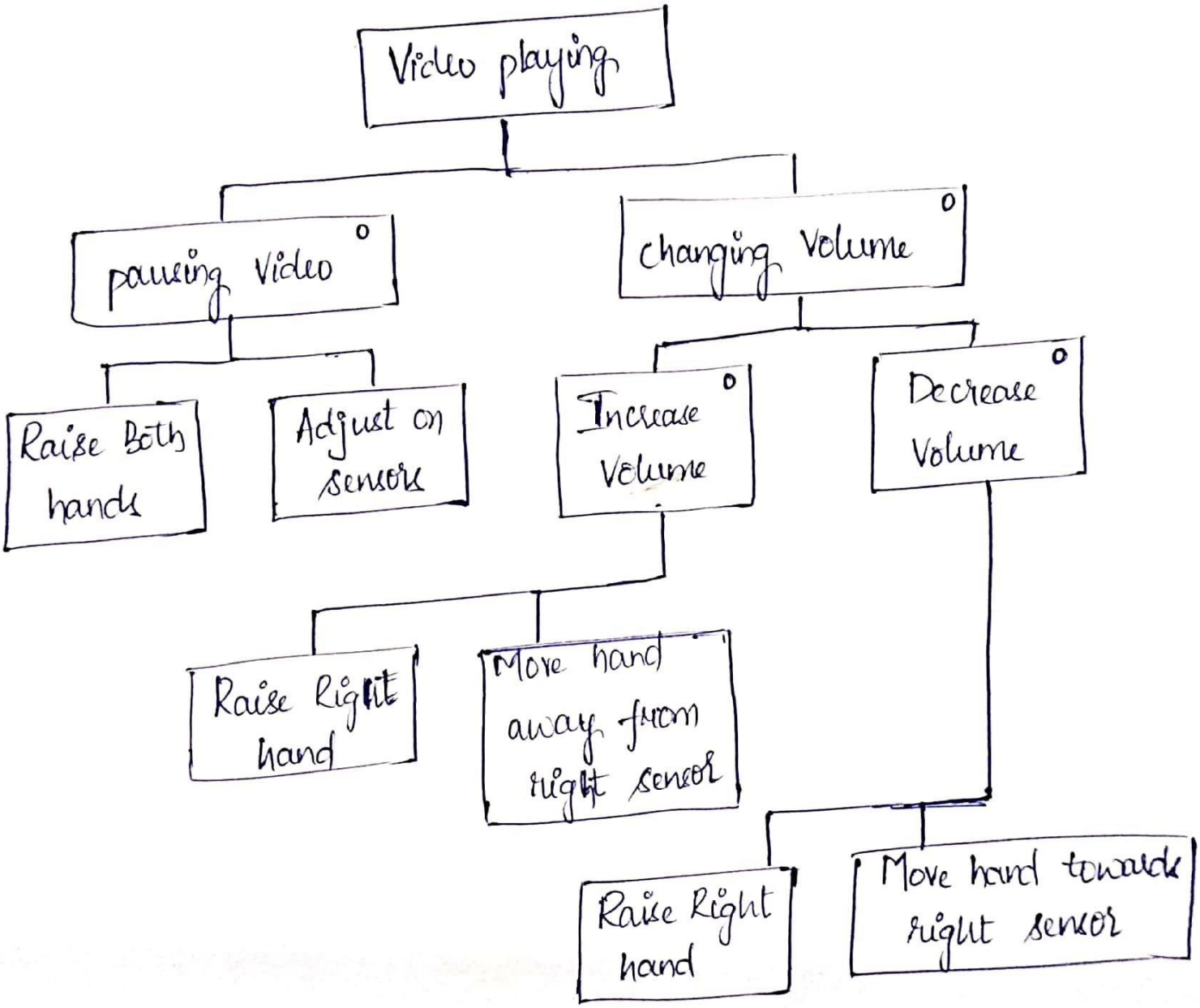
n → number of levels the sound needs to be changed.

→ State transition network:

Petrinet :


Video playing
decision of pausing the video
decision of changing Volume
To pause
To change Volume
Adjusting both hands on sensors
Decision of increasing Volume
Decision of decreasing Volume
Video paused
To Increase Volume
To decrease Volume
move hand away from right sensor
move hand towards right sensor
Volume Increased
Volume decreased

Jackson Structured Diagram:



	Pausing		Inc/dec volume	
	traditional	product	traditional	product
KLM	1.95s	1.75s	8.3s	3.1s
MHP	310ms	240ms	380ms	240ms

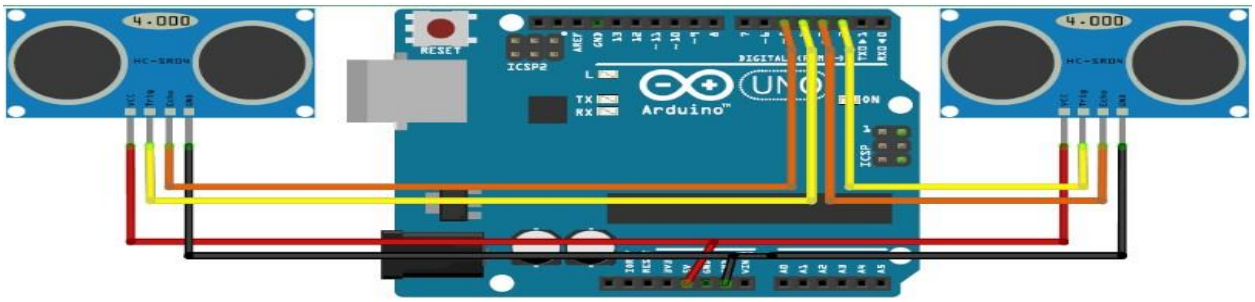
fig:- Comparision table

SNAPSHOTS OF THE PROJECT IMPLEMENTATION:

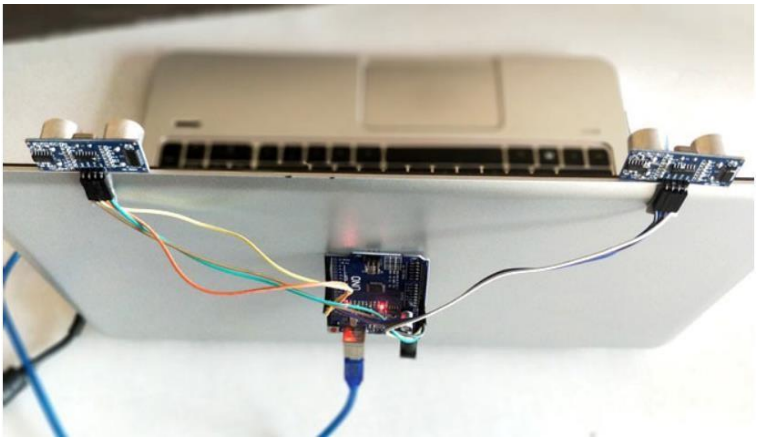
Basic circuit diagram:

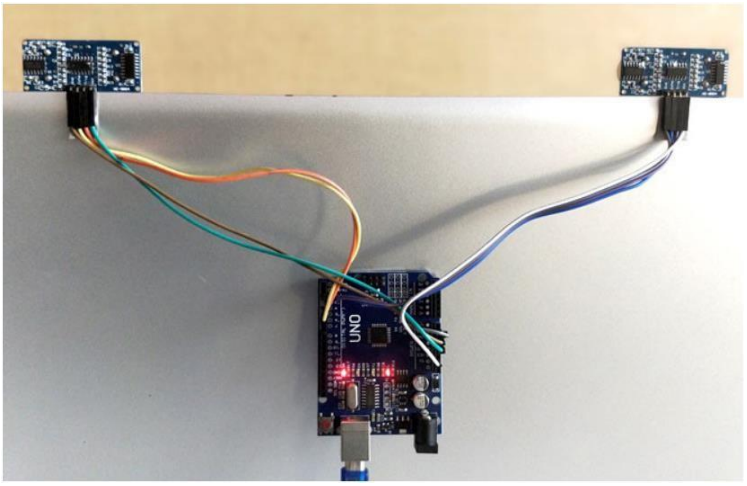
We connect two ultrasonic sensors to a hardcoded Arduino which will determine the distance between the hand and the sensor and control the keyboard accordingly.

In the picture below we can clearly see that input & output as well as GND pins are connected, this will allow the ultrasonic sensors to work without the use of an external battery.



Setting up the environment





In these pictures we can see that the sensors and the Arduino are placed in such a way that the Arduino is not visible to the user and also the Sensors are placed so that it is comfortable for the user to use his/her hands to use the product. This positioning of the components has been designed keeping in the mind the usability factors. You can also see that the Arduino is connected to the Machine (PC, Laptop) using an USB Connector, this is not only useful for communicating with the python module but also it allows the Arduino to function without the need of an external battery.

In working Snapshot:

This is a snapshot taken while testing **Test case 2.1**, here the hand is placed on the left sensor which controls the Audio aspect in our project, when the Hand is pulled away from the sensor, you can see an increase in volume in the screen.

Test Report:

TEST ID	MODULE NAME	TEST DESCRIPTION	ACTUAL RESULT	EXPECTED RESULT	TEST CASE
1.1	US -1,2	Pausing the video	Video paused	Video paused	Pass
1.2	US-1,2	Playing the video	Video started	Video started	Pass
2.1	US-1	Increasing the volume	Volume increased	Volume increased	Pass
2.2	US-1	Decreasing the volume	Volume decreased	Volume decreased	Pass
3.1	US-2	Forwarding the video	Video forwarded	Video forwarded	Pass
3.2	US-2	Rewinding the video	Video rewind	Video rewind	Pass

AS we can see all the Test cases have been passed! Thank you