

HUMAN BODY GESTURE CONTROLLED GAMES PORTAL FOR IMPLEMENTATION OF SOCIAL CAUSES

Submitted in partial fulfilment of the requirements for
the degree of

**BACHELOR OF ENGINEERING
In Electronics**

By

Bhairavi Chavan (07)

Sandesh Pal (29)

Jayesh Mahajan (38)

Shubham Metkar (40)

Supervisor:

Mrs. Jayamala Adsul



Department of Electronics Engineering

Vivekanand Education Society's Institute of Technology

University of Mumbai

Academic Year: 2021-22

CERTIFICATE

This is to certify that the project entitled “**Human body gesture-controlled games portal for implementation of social causes**” is a bonafide work of **Bhairavi Chavan (D16B-07), Sandesh Pal (D16B-29), Jayesh Mahajan (D16B-38), Shubham Metkar (D16B-40)** submitted to the V.E.S. Institute of Technology in partial fulfilment of the requirement for the award of the **Bachelor of Engineering in Electronics**.

Supervisor

Mrs. Jayamala Adsul

External Examiner

Head of Department

Mrs. Kavita Tewari

Principal

Dr. (Mrs.) J. M. Nair

PROJECT REPORT APPROVAL FOR B.E.

Project report entitled “**Human body gesture-controlled games portal for implementation of social causes**” by **Bhairavi Chavan (D16B-07)**, **Sandesh Pal (D16B-29)**, **Jayesh Mahajan (D16B-38)**, **Shubham Metkar (D16B-40)** is approved for the degree of **Bachelor of Engineering in Electronics** from University of Mumbai during the academic year **2021-2022**.

Examiners:

1. _____

2. _____

DECLARATION

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Bhairavi Chavan
(D16B - 07)

Sandesh Pal
(D16B -29)

Jayesh Mahajan
(D16B - 38)

Shubham Metkar
(D16B - 40)

Date:

ACKNOWLEDGEMENT

We are thankful to our college Vivekanand Education Society's Institute of Technology for considering our project and extending help at all the stages needed during our work of collecting information regarding our project.

It gives us immense pleasure to express our deep and sincere gratitude to **Mrs. Jayamala Adsul** (Project Supervisor) for his kind help and valuable advice during the development of project synopsis and for his guidance and suggestions.

We are deeply indebted to Head of the Electronics Department **Mrs. Kavita Tiwari**, and **Principal Dr. (Mrs.) J. M. Nair** for giving us this valuable opportunity to do this project. We express our hearty thanks to them for their assistance without which it would have been difficult in finishing this project and project review successfully.

We convey our deep sense of gratitude to all the teaching and non-teaching staff for their constant encouragement, support, and selfless help throughout the project work. It is our great pleasure to acknowledge the help and suggestions, which we received from the Department Electronics Engineering.

ABSTRACT

Gaming has always held the requirement of peripheral devices such as keyboards, mouse, joystick, etc. But with computer technology evolving every single day, human-body gestures are found to be a potential replacement for the primary accessory of Human-Computer Interaction (HCI).

In this project, an idea of a human-body gesture-controlled gaming desktop application has been proposed. With a simple web camera, body gestures have been detected and image processing techniques have been performed with the help of mediapipe - a prebuilt computer vision neural network, which enables a user to directly interact with the gaming console and with the use of OpenCV, a library designed to solve computer vision problems and which provides many useful programming functions used for features like face, edge, object detection, etc. So, the input images can be processed and communicated with the gaming commands virtually.

The activity of the said user is recorded and stored to MongoDB, an open-source No-SQL document-based database program. With this desktop application, a user can play any of the various games offered on the portal.

Furthermore, the time spent by the user on the application is stored. Corresponding to this time, the amount of money will be transferred to the social organizations.

INDEX

1. INTRODUCTION.....	01
2. LITERATURE REVIEW	03
3. PROPOSED SYSTEM	06
3.1 Methodology	07
3.2 Block Diagram	08
3.3 Working	09
4. HARDWARE / SOFTWARE SCHEME	10
4.1 Hardware Requirements.....	11
4.2 Software Requirements	11
A. For Game Development	11
B. For Automation	11
C. For Authentication.....	15
D. For Centralized Database Management	15
E. For Desktop application development	16
5. RESULTS	18
5.1 Developed Games	19
5.2 App Development	24
6. ADVANTAGES, APPLICATIONS & FUTURE SCOPE	25
6.1 Advantages.....	26
6.2 Applications	26
6.3 Future Scope	27
7. CONCLUSION	28
8. REFERENCES.....	30

LIST OF TABLES & FIGURES

Fig 3.1 Block Diagram	08
Fig 4.1 Mediapipe Framework Architecture	13
Fig 4.2 Authentication using Auth0	15
Fig 4.3 MongoDB (Centralized Database Management)	16
Fig 5.1 Quiz game	19
Fig 5.2 Bubble game	20
Fig 5.3 Red Light Green Light	21
Fig 5.4 Car Racing game	23
Fig 5.5 Flappy game	24
Fig 5.6 Application GUI	24

CHAPTER 1: INTRODUCTION

Chapter 1: Introduction

Gaming facilities have become part of people's lives and they now act as a source of entertainment and refreshment for individuals of all ages. Computer technology, too, over the decades, has developed immensely. As a result, people have been trying to evolve and develop ways to interact with devices, other than the usual peripherals like keyboards, mouse, joysticks, etc. Thus, Human-Computer Interaction (HCI) came into existence. Today, HCI is an integral part of our lives through everyday examples like our touch screen mobile phones, Google Now, Apple's Siri, etc. for voice commands which allow hand-free controls. Using this technology, in the gaming industry, we see many recently developed 3D games and applications. For these games, there have also been developments in controlling devices. Kinect, Leap Motion controllers, Wiimote and CyberGlove are all popular motion capture technologies. For example, the Nintendo Wiimote consists of a wireless remote, which is the primary controller for the console. So, the age-old controlling devices like keyboards and mouse are now being replaced by ones with more simple and direct control, that is by body gesture recognition. This technology gives the user a sense of simplicity and convenience and even the simplest game becomes interesting due to the massive change in the mechanism of the game.

CHAPTER 2:

LITERATURE REVIEW

Chapter 2: Literature Review

A review of relevant literature was conducted in order to gauge the current level of progress in the fields of computer vision. The current methodologies and research are as follows.

1. Hand Gesture Controlled Gaming Application by Tanay Thakar, Rohit Saroj, Prof Vidya Bharde

Human Computer Interaction deals with how users interact with their systems and how the user experience and the user interfaces can be optimized to improve user satisfaction. The proposed system is vision based, which uses image processing techniques and inputs from a webcam. The input channel used is the hand, which is an object that is recognised by the interface through Open Computer Vision library and it's relative position is mapped along the way.

The system can be broken down into three main components:

1. Hand Pivot Position Region:

This is the region that will be used as a landmark or reference to track the hand movement gesture.

2. Hand Position Detection:

The position of the hand is tracked by the center of the landmark region & the hand itself. The direction to which the hand is moved will be detected by calculating the position of the hand center by the position of the landmark region's center.

3. Cursor Control:

Once the hand gestures are recognized, it will be a simple matter of mapping different hand gestures to specific functions.

2. **Hand Gesture Recognition System for Image Process (IP) Gaming**
by Ashwini Shivatare, Poonam wagh, Mayuri Pisal, Varsha Khedkar
Prof. Mrs. Vidya Kurtadikar

This paper talks about how using a camera can be used for Detection hand gestures and can be applied to any game control. They are using cameras as a detecting device as well as input device for Augmented Reality System. Their proposed system helps reduce the burden on experts to look into few regular activities. Their system has a single player playing a game through hand gestures using a web camera. The motions are detected using a web camera and passed to the system for processing. The system consists of OpenCV and Java Application, using these Hand Detection and Background subtraction is done. This processed image is checked in which area it has occurred and its corresponding key event is called for action to be performed in the game. These steps are repeated till the player is playing the game.

3. **A Review on Hand Gesture Recognition System by Jayesh**
S. Sonkusare Nilkanth. B. Chopade, Ravindra Sor, Sunil.L. Tade

The authors of this paper explained the different methods of hand gesture recognition.

1. **Pixel to Pixel Comparison:** This method involves the pixel by pixel comparison of the frame captured and with the image database. This method is not so accurate but easy to implement.
2. **Edges Method:** The objective of this method is to find out the highest gradient in an image and this is found by applying the threshold in the gradients. Threshold will remove the low magnitude gradients.
3. **Using Orientation Histogram:** Orientation histogram is dependent on the feature vectors, results in formation of histogram based on the edges of the image. First the system is trained with the images. As images are captured by the webcam as the input these images are converted to the gray scale and from this the histogram is created which is used as the training pattern.

CHAPTER 3:

PROPOSED SYSTEM

Chapter 3: Proposed System

3.1 Methodology

The technology or method which uses any kind of sensors to read, analyze and interpret gesture movements resulting in some computational commands is known as gesture recognition. Gesture recognition requires carrying out certain processes in order to produce meaningful outcomes for any video stream. This gesture recognition follows a generalized series of steps.

- a. The first and foremost step is image acquisition where the user shows gestures to the webcam which takes pictures in an infinite loop until the user terminates the program and is readily followed by a subprocess of image filtering which reduces noise in the image.
- b. The second process of image segmentation involves the determination of the pixels that belong to the human gesture through various processes such as Canny Edge detection, Background Subtraction, Calibration and Thresholding.
- c. Once this process is over then an outline that joins all the pixels on the outside of the recognized gesture image is determined which is called the contour of the image.
- d. The selection of valid gestures is then carried out onto the largest contour found. Then the calculation of a few geometric features generated from the contour position will be fed as inputs to the gesture recognition algorithm.

The framework with which various gesture recognition and applications are going to be carried out in our project is mediapipe. **MediaPipe** is a Framework that is built for machine learning pipelines to process time-series data like video, audio, etc. Desktop/Server, Android, iOS, and embedded devices like Raspberry Pi are a few of the platforms to which mediapipe supports its solutions and APIs hence it is a cross-platform framework. This framework holds some dependencies such as OpenCV for video processing, FFMPEG to manage audio data.

Out of numerous solutions provided by mediapipe, few of the solutions such as the face detection model, hand landmark model and pose model are extensively used.

Face detection: Mediapipe gives a real-time face recognition system, which is a rapid face detection system with 6 landmarks and multi-face support.

Hand Landmark: Regression or direct coordinate prediction, is said to perform unambiguous keypoint localization of 21 3D hand-knuckle coordinates within the observed hand areas. Even with faintly covered hands and self-occlusions, the model develops a consistent internal hand posture representation.

Pose detection: Pose detection uses a two-step detector-tracker ML pipeline which has well functioned for hands and face mesh detections. The pipeline initially locates the person/pose region-of-interest (ROI) inside the frame using a detector. Using the ROI-cropped frame as input, the tracker then determines the posture landmarks and segmentation mask inside that ROI.

3.2 Block Diagram

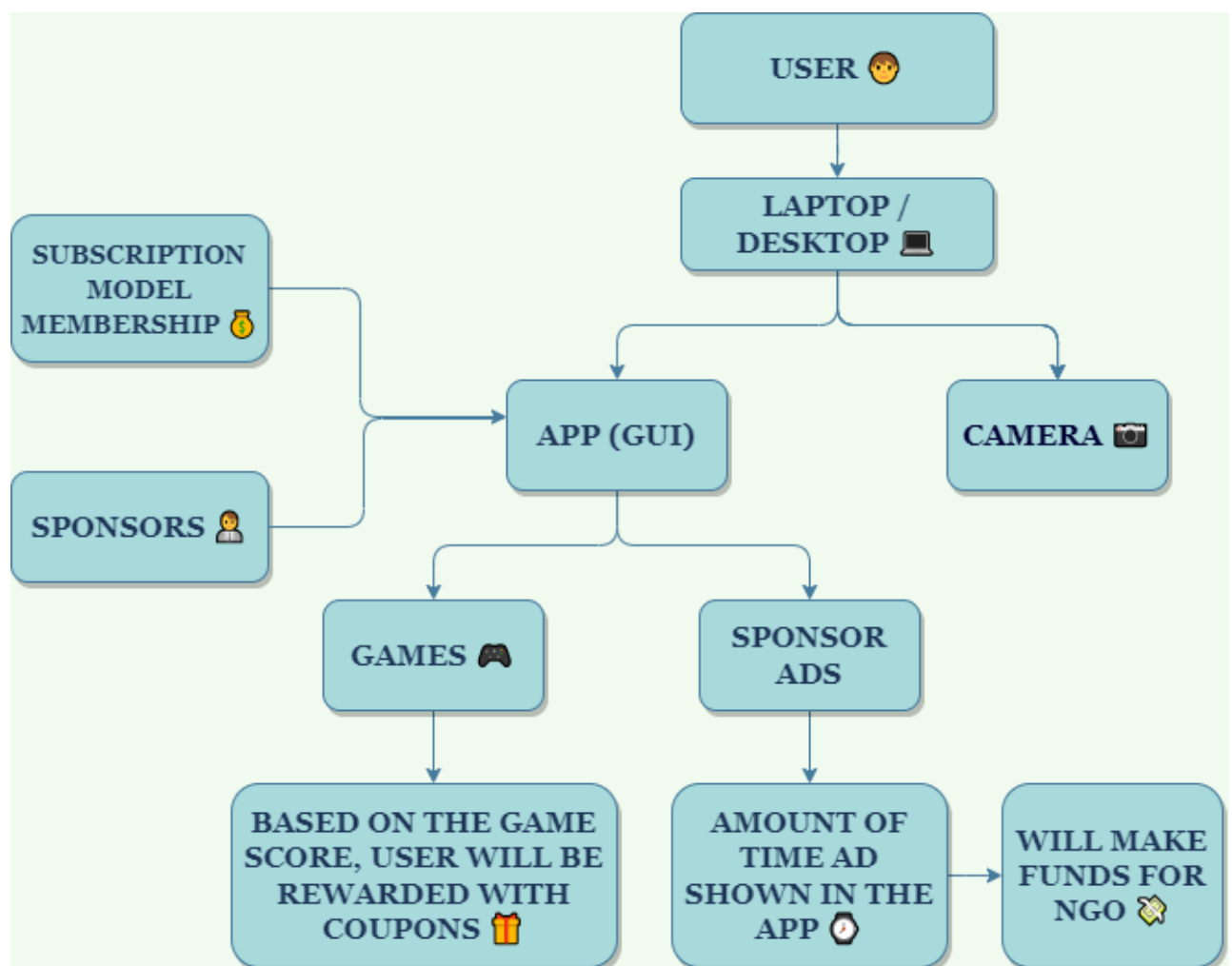


Figure 3.1 Block Diagram

3.3 Working

- The objective of this proposed system is to develop an application for numerous games into a single portal. For this, we use Electronjs, an open-source cross-platform framework that is used to build desktop applications that use Chromium and Node.js. Electronjs is used because game files are written using the python language and for designing desktop application, technologies such as HTML, CSS and Javascript are used. Electron makes it easy to integrate these technologies.
- Whenever a user opens the desktop application the user has been authenticated first. Authentication is developed using Auth0 which provides secure and easy authentication as well as authorization service. There are multiple social identity provider options available for users to directly log in using their google, linkedIn, github, etc. accounts along with conventional sign-up and sign-in methods with username and password. After successful login, the user sees the main dashboard of the app which contains a navigation bar and different games that users can play.
- The navigation bar provides us with different options such as the membership option to take the membership in order to play some games which are reserved for members, then user rewards in which the user can see all his rewards, then user profile and a logout button.
- The application contains different games that use different body parts to play games using gestures that can be played by anyone of any age. The games are developed using OpenCV, python and using libraries such as pygame, cvzone and pyautogui. Every game has instructions on how to play each game using hand gestures respectively along with the play game button.

CHAPTER 4:

HARDWARE / SOFTWARE SCHEME

Chapter 4: Hardware / Software Scheme

4.1 Hardware Requirements

The proposed system uses various software schemes, with one very simple and basic hardware requirement, that is, a computer or laptop with webcam access. This is the only hardware required for this system.

4.2 Software Requirements

A. For Game Development:

i) Python: We are using Python as a coding language for the development of our games. Python is a computer programming language often used to build websites and software, automate tasks, and conduct data analysis. Python is a general-purpose language, meaning it can be used to create a variety of different programs and isn't specialized for any specific problems.

ii) Pygame: Pygame is a cross-platform set of Python modules which is used to create video games. It consists of computer graphics and sound libraries designed to be used with the Python programming language.

B. For Automation:

i) OpenCV: OpenCV (Open-Source Computer Vision Library) aims at real-time computer vision. The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image of an entire scene, find similar images from an image database, remove red eyes from images taken using flash, follow eye movements, recognize scenery and establish markers to overlay it with augmented reality, etc.

ii) PyAutoGUI: PyAutoGUI is a cross-platform GUI automation module that works on Python 2 & 3. You can control the mouse and keyboard as well as perform basic image recognition to automate tasks on your computer. PyAutoGUI lets our Python scripts control the mouse and keyboard to automate interactions with other applications. PyAutoGUI has several features like:

- Moving the mouse and clicking in the windows of other applications.
- Sending keystrokes to applications (for example, to fill out forms).
- Take screenshots, and given an image (for example, of a button or checkbox), and find it on the screen.
- Locate an application's window, and move, resize, maximize, minimize, or close it (Windows-only, currently).
- Display alert and message boxes.

iii) Mediapipe: Mediapipe offers ready-to-use yet customizable Python solutions as a prebuilt Python package. It provides a variety of ML Solutions such as face detection, hand detection, Object Detection, Motion Tracking, etc. MediaPipe is a Framework that is built for machine learning pipelines to process time-series data like video, audio, etc. Desktop/Server, Android, iOS, and embedded devices like Raspberry Pi are a few of the platforms to which mediapipe supports its solutions and APIs hence it is a cross-platform framework. This framework holds some dependencies such as OpenCV for video processing, FFMPEG to manage audio data.

Architecture of Mediapipe:

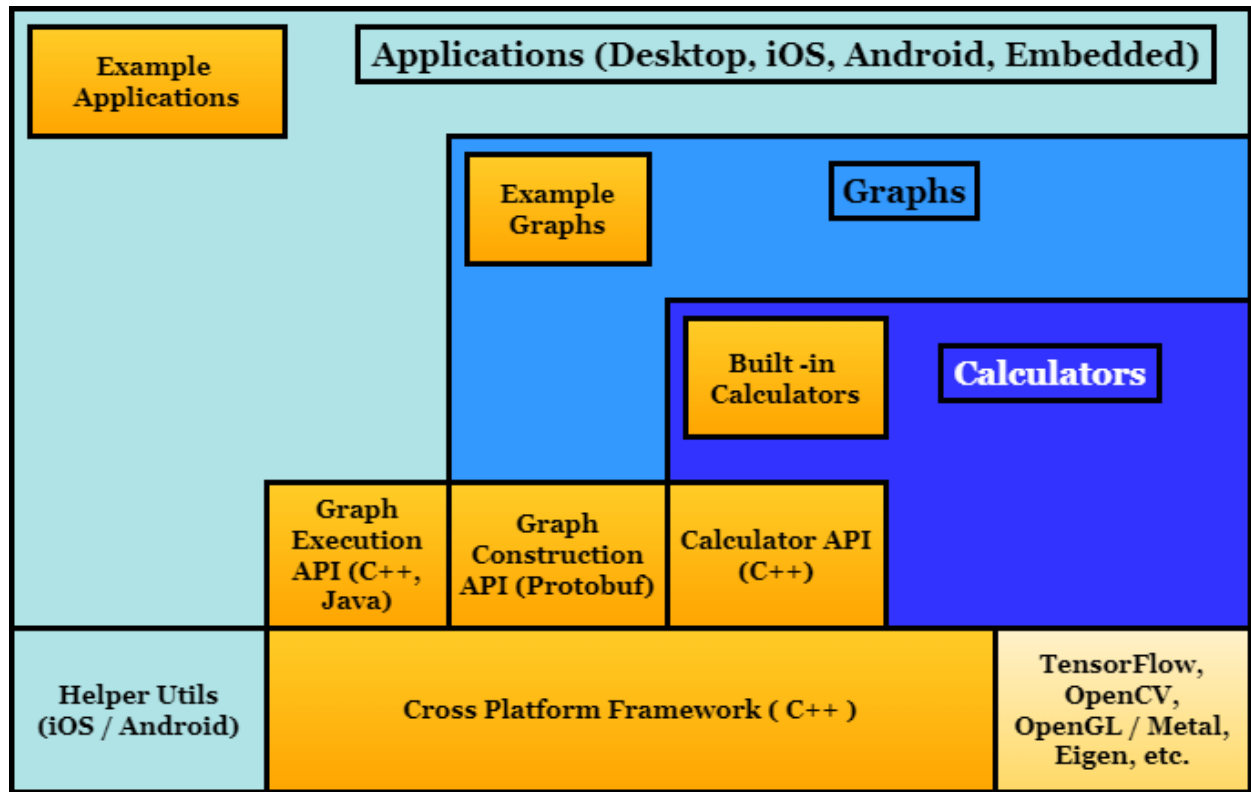


Fig. 4.1 Mediapipe Framework Architecture [13]

a. **Packets:** It is the basic data flow unit of the framework. It consists of a timestamp in the numeric format and a shared pointer to a payload that is not mutable. Packets are value classes and can be copied cheaply. Each copy shares ownership of the payload, with reference counting semantics and here timestamp is owned by every copy [15]. Through sending and receiving of the packets communication of the calculators takes place. At each input timestamp, a single packet is sent along each input stream. Any kind of data can be contained, such as a single frame of video or a single integer detection count.

b. **Graph:** It defines packet flow paths between nodes and inside this entire mediapipe processing takes place. Any number of inputs and outputs is allowed with bidirectional data flow support. GraphConfig is specified as functionality that describes the topology and functionality of a MediaPipe graph. Modularization of CalculatorGraphConfig into submodules takes place with the help of a subgraph. The third term under the graph is a cycle. Mediapipe needs acyclic nature graphs and cycles are assumed as errors.

- c. **Nodes:** These are present at those places where most of the graph's work takes place. The number of input and output ports is determined by each node's interface.
- d. **Stream:** It is defined as a connection that transports a sequence of packets between two nodes.
- e. **Side Packets:** A single packet is transmitted by a side packet link between nodes. Comparing streams with side packets, the persistent flow of information is dynamic whereas inside packets some part of information remains unchanged.
- f. **Calculator:** Each calculator is a graph node. The ports in a calculator are where data packets such as video frames / audio segments arrive and exit. The packet payload type that will transit the port is declared when a calculator is initialized. The Framework implements Open, Process, and Close procedures in the calculators every time a graph runs. The calculator is started by running the open function; the process is repeated each time a packet is entered. Pre-processing calculators, Inference calculators, Post-processing calculators and Utility calculators are some of the types of mediapipe calculators [13].
- g. **Scheduling mechanics:** Calculators are used for processing data in a mediapipe graph. A Minimum of one scheduler queue exists for each graph. One executor is allocated to each scheduler queue. A queue is assigned to nodes in a static manner (and therefore to an executor). There is only one queue by default and its executor is a thread pool with a thread count decided by the system's capabilities.
- h. **Synchronization:** The mediapipe framework is accountable for supplying input synchronization for nodes, which is one of its main purposes. A timestamp's essential role in terms of framework mechanics is to serve as a synchronization key.
- i. **Input policies:** The process of synchronization is managed locally across each node, according to the node's input policy [15].

C) For Authentication:

Auth0 is a flexible, drop-in solution to add authentication and authorization services to our applications. Auth0 uses the OpenID Connect (OIDC) Protocol and OAuth 2.0 Authorization Framework to authenticate users and get their authorization to access protected resources.

With Auth0, we can easily support different flows in our own applications and APIs. So the login windows that you see for any web application, that authentication can be implemented using Auth0. In our system we have implemented the authentication through various platforms like Google, Twitter, GitHub etc.

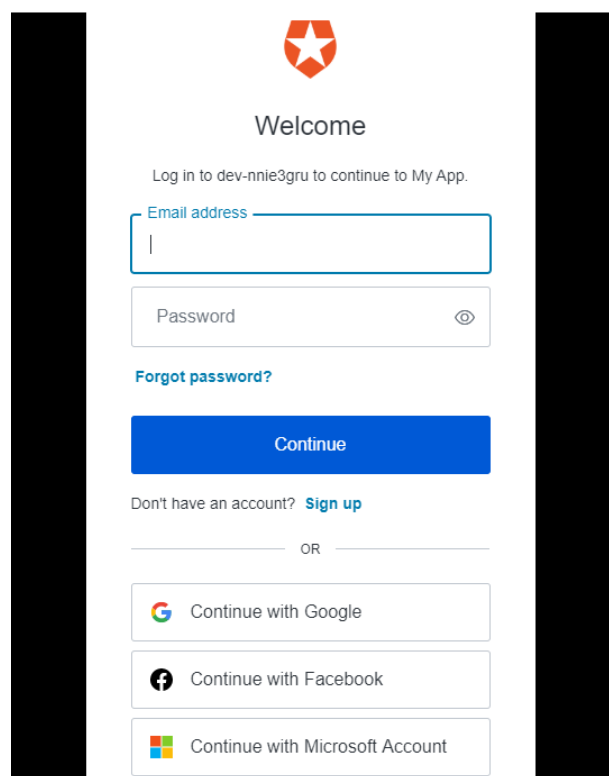


Fig 4.2 Authentication using Auth0

D) For Centralized Database Management:

MongoDB is a cross-platform, document-oriented database that provides high performance, high availability, and easy scalability. MongoDB works on the concept of collection and document.

MongoDB is used as a database service which is an open source cross-platform NoSQL database program. As it is a document-oriented database and it stores data in JSON-like documents along with optional schemas. This data is then made accessible by turning it into APIs which can be consumed by any client-side application to get regular analytics/metrics/reports on dashboards.

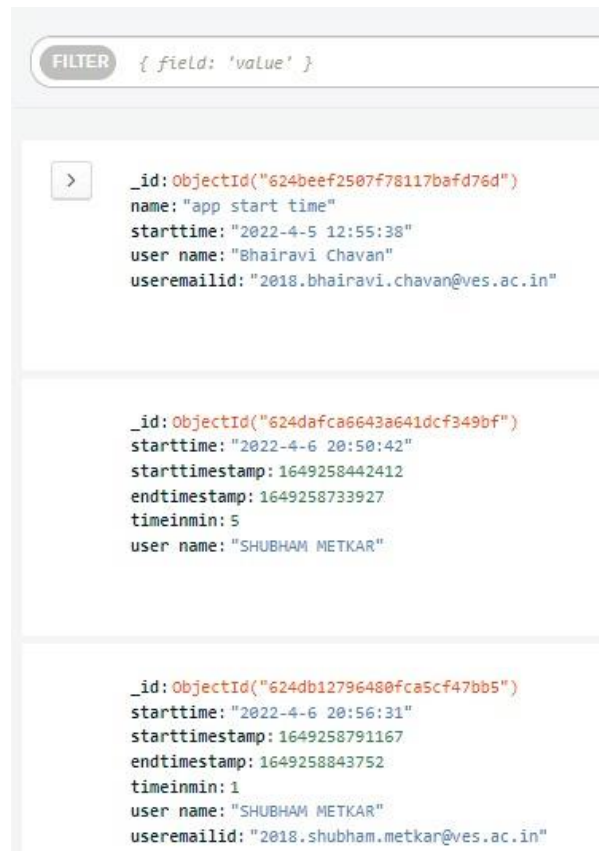


Fig 4.3 MongoDB (centralized database management)

E) For Desktop application development:

i) **HTML**: HTML stands for **Hyper Text Markup Language**, which is the most widely used language on the web to develop web pages. So, we use HTML to simply "mark-up" a text document with tags that tell a Web browser how to structure it to display.

ii) **CSS**: **Cascading Style Sheets**, which is a simple design language intended to simplify the process of making web pages presentable and we can control the colour of the text, the style of fonts, the spacing between paragraphs etc.

iii) **Bootstrap**: Bootstrap is a sleek, intuitive, and powerful, front-end framework for fast and easier web development. It uses HTML, CSS and JavaScript and comes with the feature of global CSS settings and fundamental HTML elements.

iv) **Javascript**: JavaScript is a lightweight, interpreted programming language which is designed for creating network-centric applications. It is integrated with Java and HTML. It is open and cross-platform. Node JS is built on Chrome's Javascript runtime for building fast and scalable network applications, which will be useful for electron.

CHAPTER 5:

RESULTS

Chapter 5: Results

5.1 Developed Games:

Quiz game:

This game tests the knowledge and intelligence of the user through a quiz. All the questions are multiple-choice questions in which four options will be given from which the user has to choose the correct answer. To select the answer, the user has to decrease the distance between the tip of the index finger and the tip of the middle finger, keeping both fingers behind the option they want to select. Users can also track how much the quiz is completed and at the end, they will see a summary of the quiz and the score according to the correct answers given.

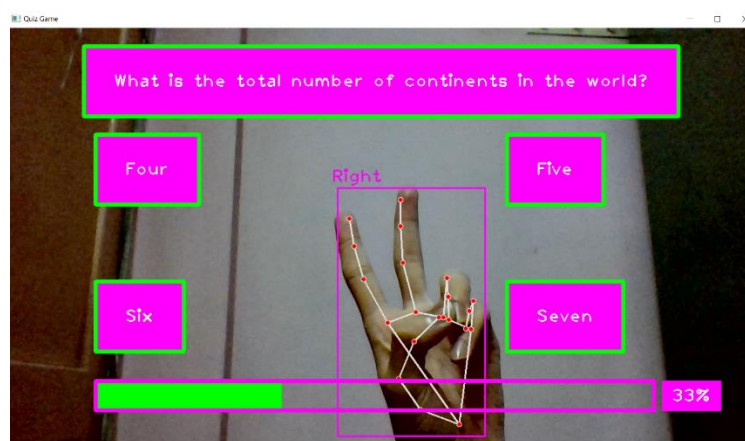


Fig 5.1 Quiz Game

Bubble game:

A bubble game involves popping as many bubbles as possible appearing on the screen in a specified amount of time. As shown in figure 6, as soon as the game starts the countdown timer will start decreasing and bubbles will appear randomly on the screen one after another when the user pops them. To pop the bubble, the user has to use the tip of the index finger on which bubble will be displayed. When the bubble shown on the index fingertip and the in-game bubble overlaps it will be popped. When the timer becomes zero, the final score will be displayed on the screen. This game tests and can help in improving the spotting level and speed of the user.



Fig 5.2 Bubble Game

Red Light Green Light game:

This game involves whole-body tracking which is implemented using the mediapipe pose detection model. The game will start only when the whole body is detected. The game interface is shown in figure 7. The user has to run during the green light and needs to pause in whichever position the user will be during the red light. The progress of the users will be measured with the help of steps taken. If the height between the right hip and right ankle is decreased then it is considered a step was taken by the user. If the user reaches 100 points then he will be the winner. If any movement is detected during the red light game will be over. This game indirectly helps the user to do the exercise as running is required to play the game and also attentiveness.

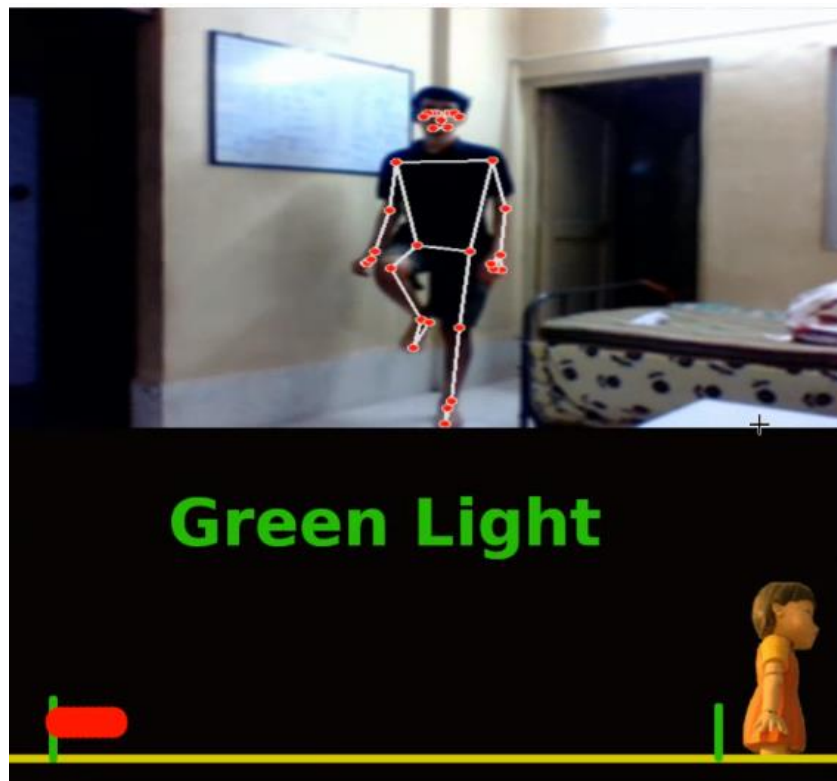
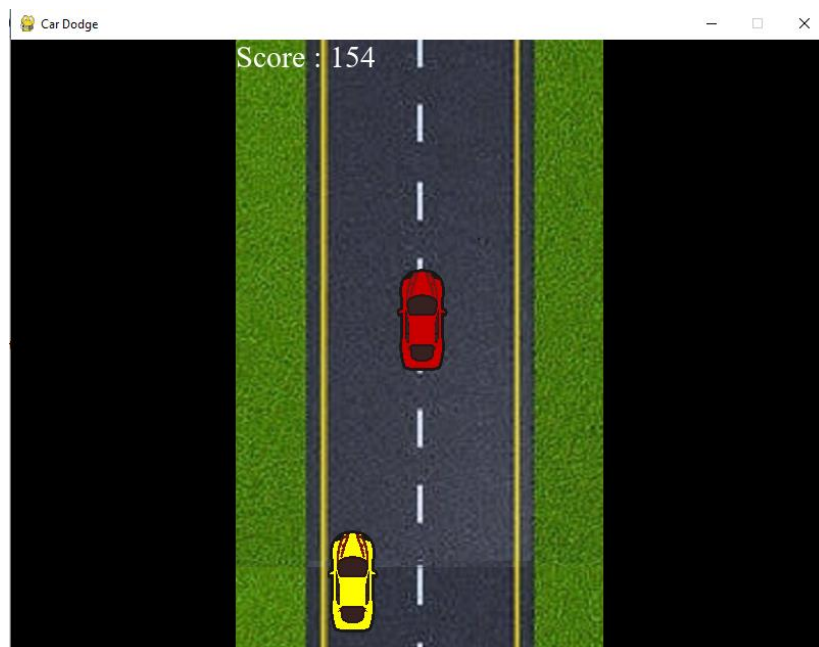
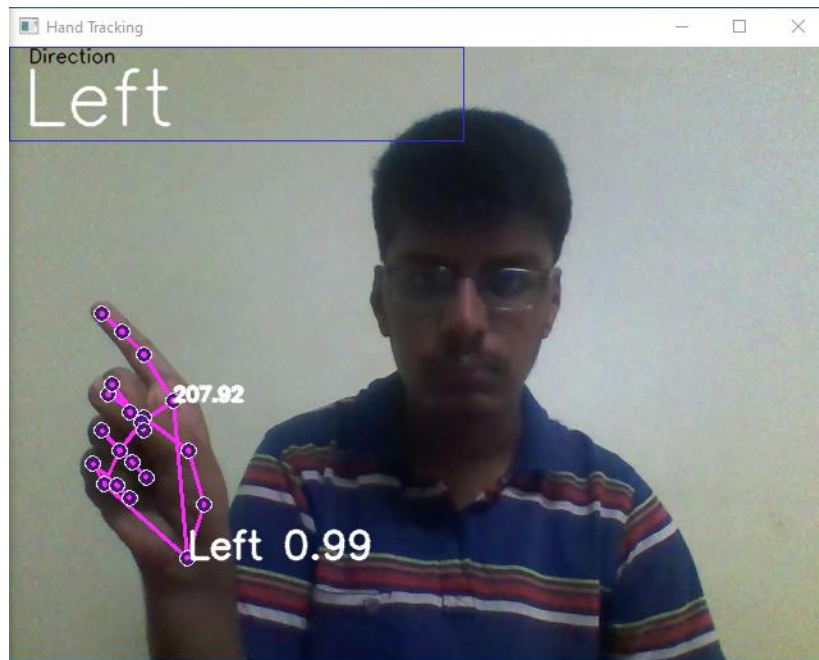


Fig 5.3 Red Light Green Light Game

Car Racing game:

The car racing game involves dodging a car coming in front of the user's car. The user has to control the car with his index finger. If the angle of the index finger is less than 180 degrees then the car will move in the right direction and vice versa. Whenever the car collides with the other car the game will be over. This game tests and improves the reflexes and focus of the user.



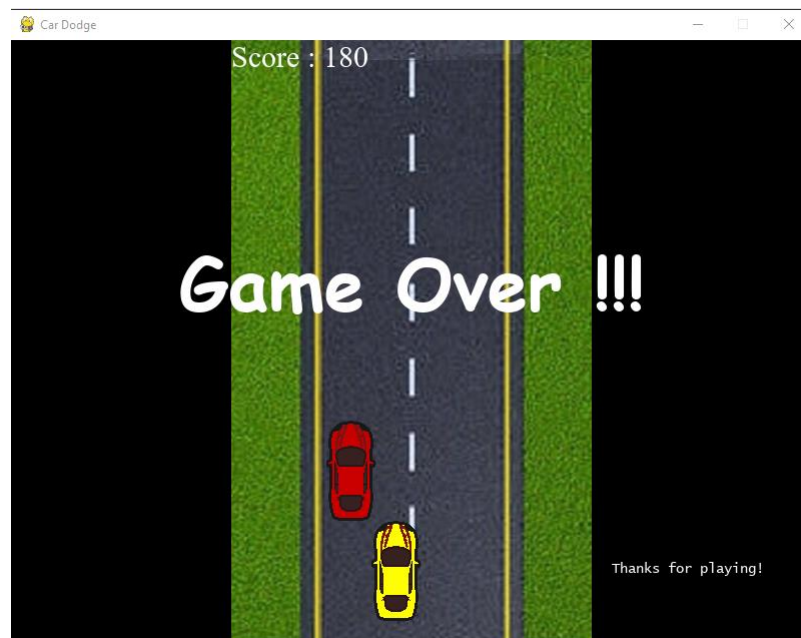


Fig 5.4 Car Racing game

Flappy Bird game:

The flappy game involves a bird dodging the pipes and moving the bird through the gap between the pipes coming one after another at a fixed distance. The bird is controlled with the help of the x and y coordinates of the nose which is obtained by using mediapipe face detection. The user has to move up and down in order to avoid the pipes coming. Whenever the bird collides with the pipe the game will be over and the score will be shown on the screen. In this game, the user has to move up and down just like squats so this game will also help in doing the exercise.

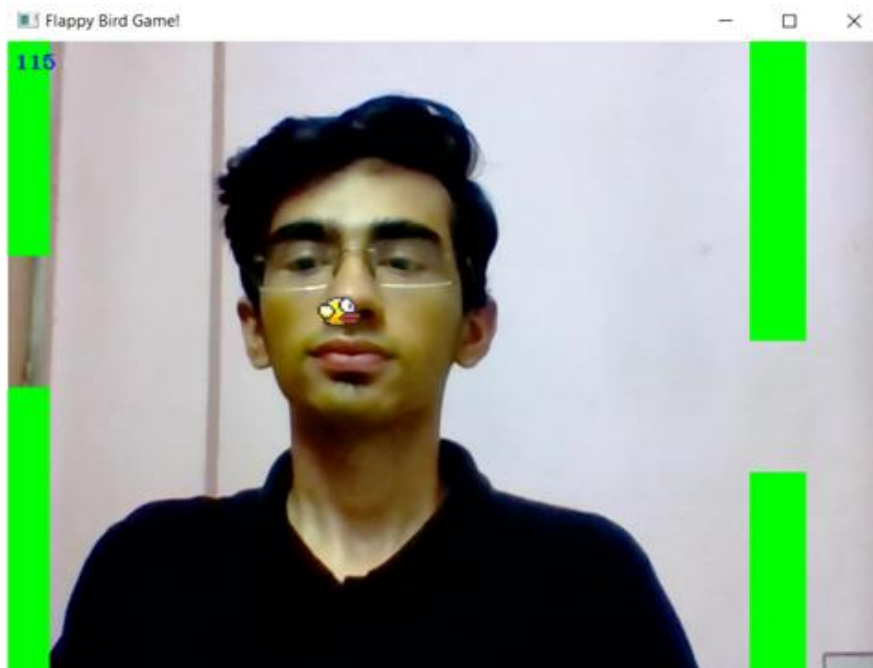


Fig 5.5: Flappy Bird Game

5.2 App Development

The application has been developed using Electronjs, an open-source cross-platform framework that is used to build desktop applications that use Chromium and Node.js.

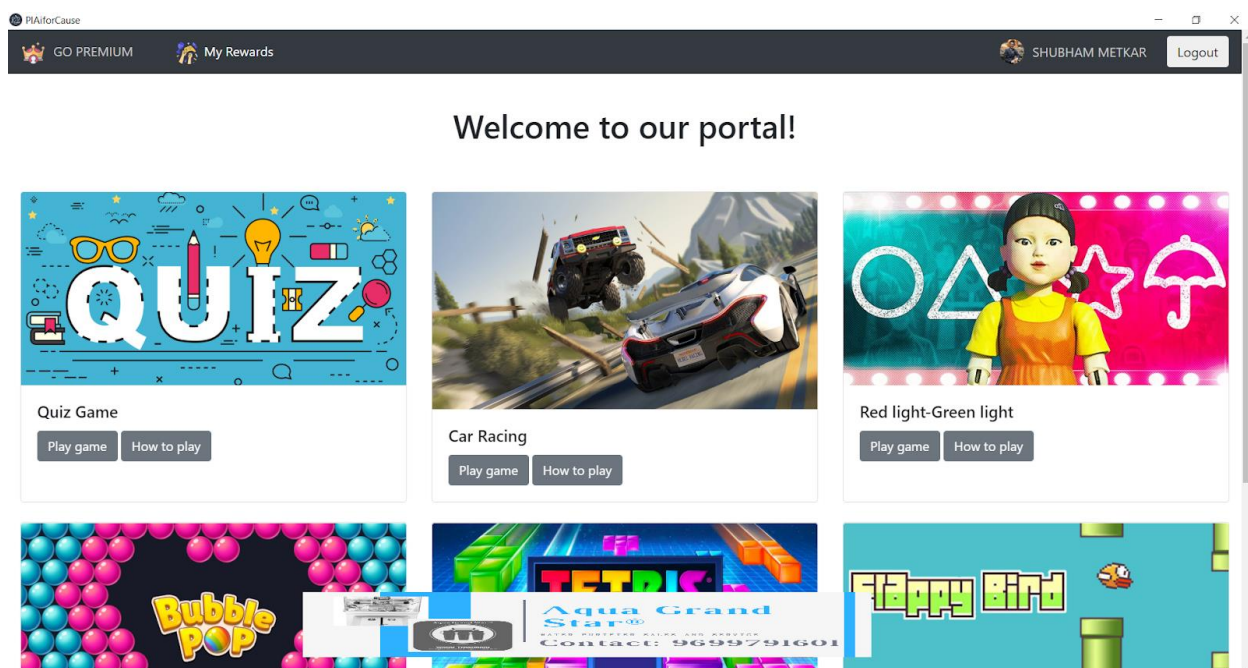


Fig 5.6 Application GUI

CHAPTER 6:

ADVANTAGES, APPLICATIONS & FUTURE SCOPE

Chapter 6: Advantages, Applications & Future Scope

6.1 Advantages

- With the help of this desktop, application users can have fitness exercises, improved reflexes, entertainment, etc.
- NGOs are empowered with funds and recognition and local or small businesses/organizations are provided with a platform to showcase their products and services.
- Users are able to install the entire application on their laptop/desktop and can open it with a single click. Also, these games are very user friendly so that people of any age group can easily play them.
- There are a lot of customizations available to enhance these games from the user's perspective. As all of the technology stacks such as libraries for game development, framework for desktop application building, database provider, etc. are all open-source therefore there is not any cost to build the entire application.

6.2 Applications

This desktop application itself serves as an application and is applicable in the following areas:

1. User: Users will enjoy playing the gesture games on the desktop application and would also receive rewards based on their active time. They will receive discount coupons on the products or services offered by sponsors.
2. Sponsor: Sponsors will be the local businesses that wish to sponsor the application and their products will be displayed through the platform.
3. NGO: The awareness will be created regarding the topic which is the NGOs purpose. NGOs will also be receiving funds which are from membership and sponsors based on the active time of users which will solve the issue that most NGOs face.

Also, Sponsors and NGOs can also see the statistics regarding active users, the display time of advertisements, and funds generated on the website.

6.3 Future Scope

- In future, more games can be added to the application which will use different techniques for gesture recognition so that users get to play a variety of games in one place. Advanced games like Subway Surf, Super Mario, Hill Climb Racing, etc. can be added to the application.
- Also, this project model can be scaled on a web application where users can hop into the application with a single click by allowing camera access on the website & with no need of downloading any resources.
- Also, a mobile application can be developed which can be easily available on the app store. We can reach more users through the mobile application as everyone has an android mobile device nowadays.
- We can also set up gaming kiosks at places such as malls, gardens, colleges, etc where public gathering takes place. Through these users can play through the setup which is placed at different places without any installation and requirements. Through this, we can also get instant feedback from the audience. We can engage more audience through the interaction and change the way of looking at the gaming.

CHAPTER 7:

CONCLUSION

Chapter 7: Conclusion

In this project, a desktop application is proposed that contains gesture-based games which use mediapipe with different features such as hand detection, face detection and pose. Different games such as quiz game, car dodge game, flappy bird game, etc. is developed which are easy to play. These games test and improve different skills of the users. Also through these games, exercising will become fun rather than just a boring task. With the help of gaming we are trying to do social cause which will be benefit our society. User is indirectly contributing for the noble cause while also having fun playing different games offered on the desktop application.

CHAPTER 8:

REFERENCES

Chapter 8: References

- [1] Harpreet Kaur and Jyoti Rani,"A Review: Study of Various Techniques of Hand Gesture Recognition", 1st IEEE International Conference on Power Electronics. Intelligent Control and Energy Systems (ICPEICES2016), IEEE 2016.
- [2] Jayesh S. Sonkusare, Nilkanth. B. Chopade, Ravindra Sor and Sunil.L. Tade, "A Review on Hand Gesture Recognition System", 2015 International Conference on Computing Communication Control and Automation, IEEE 2015.
- [3] Chao Peng, Jeffrey Hansberger, Vaidyanath Areyur Shanthakumar, Sarah Meacham, Victoria Blakley and Lizhou Cao, "A Case Study of User Experience on Hand-Gesture Video Games", 2018 IEEE Games, Entertainment, Media Conference (GEM).
- [4] Ashwini Shivatare, Poonam wagh, Mayuri Pisal,Varsha Khedkar and Prof. Mrs. Vidya Kurtadikar, "Hand Gesture Recognition System for Image Process(IP) Gaming", International Journal of Engineering Research and Technology (IJERT), Vol. 2 Issue 3, March - 2013.
- [5] Guan-Feng He, Jin-Woong Park, Sun-Kyung Kang and Sung-Tae Jung, "Development of Gesture Recognition-Based Serious Games", Proceedings of the IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI 2012) Hong Kong and Shenzhen, China, 2-7 Jan 2012.
- [6] Doe-Hyung Lee and Kwang-Seok Hong,"Game Interface using Hand Gesture Recognition".
- [7] Oana Saman and Loredana Stanciu,"Hand Mobility Recovery through Game Controlled by Gestures", SACI 2019 • IEEE 13th International Symposium on Applied Computational Intelligence and Informatics • May 29-31 • Timioara, Romania.
- [8] Tanay Thakar, Rohit Saroj and Prof Vidya Bharde3,"Hand Gesture Controlled Gaming Application", International Research Journal of Engineering and Technology (IRJET), Volume: 08 Issue: 04, IRJET, Apr 2021.
- [9] Zi Xian Justin Yeo,"Hand Recognition and Gesture Control Using a Laptop Web-camera".
- [10] Siddharth S., Rautaray and Anupam Agrawal,"Interaction with Virtual Game through Hand Gesture Recognition", 2011 International Conference on Multimedia, Signal Processing and Communication Technologies, IEEE 2011.
- [11] Alvin T.S. Chan, Hong Va Leong and Shui Hong Kong,"Real-time Tracking of Hand Gestures for Interactive Game Design", IEEE International Symposium on Industrial Electronics (ISIE 2009), July 5-8, 2009.

- [12] <https://google.github.io/mediapipe/>
- [13] <https://learnopencv.com/>
- [14] <https://blog.tensorflow.org/2021/08/3d-pose-detection-with-mediapipeblazepose-ghum-tfjs.html>
- [15] <https://arxiv.org/>