Technische Hochschule Augsburg

Treat On You

An Air Hockey Game Using Hand Gestures

INTERACTION ENGINEERING

Prof. Dr. Michael Kipp
Hochschule Augsburg,
Winter Semester 2023/24

Ifrah Asif
Rachal Tu

Table of Contents

CONTENTS

ABSTRACT

Treat On You is an interactive Air Hockey game using hand gestures. A monitor screen is used as a board for playing the game. The goal is to hit the puck back and forth without letting it go into the opponent's goal. Whoever goals reaches 3; loses the game and has to pay for the treat. The implementation environment demonstrates the board game on the monitor screen and uses a phone camera to detect hand gestures. We developed two different hand-gesture interaction prototypes and compared these methods in a user study. This user study allowed us to analyze the usability of the two methods and the preferred type of interaction for users.

To summarize, our research demonstrates how users can interact with a virtual puck using their hands and which hand gesture type could be more efficient. In our comparison of different hand interactions, we discovered that users prefer the 90-degree method because it is easier and comfortable, and makes them feel more controlled.

MOTIVATION

We were inspired by the idea of “Card Roulette” tradition which is picking out a random credit card from a deck of cards by the waiter to determine who would pay the bill among friends. We wanted to transform this traditional idea into a fun game concept that friends could play with each other using their credit cards.

Then we implemented this idea in the form of Air Hockey game which is played between 2 players with the use of mallets and a puck. In Air Hockey game, users must use mallets controlled by their hands to play with the puck/ball. For our approach, we attempted to map these controls to more related movements in order to make the interaction more fun and challenging. So, the main motivation of Treat On You is to make traditional Air Hockey game more entertaining and challengeable with novelistic interactions.

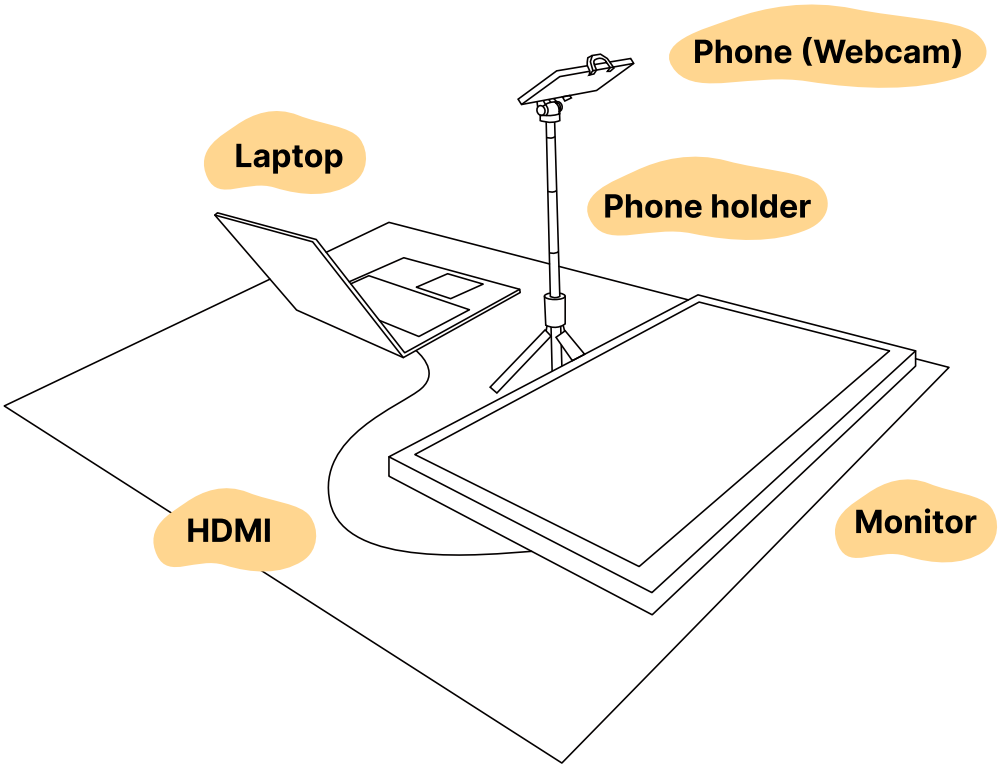
Idea 1:
Card Roulette

Idea 2:
Air Hockey

CONCEPT

1. ENVIRONMENT SET-UP

We are using our phone camera as our webcam to detect hand gestures For wireless connection, we used the Iriun app which needs to be run on both mobile and laptop at the same while using the same wifi. For Playing the game, we used a monitor screen as our table. The monitor is further connected to the laptop through an HDMI cable where our code is executed and reflected on the monitor screen.



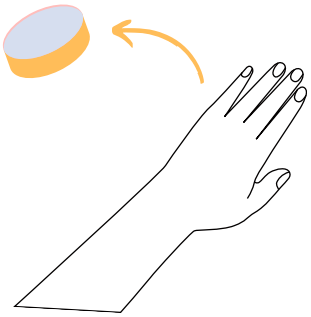
2. INTERACTIONS

We developed two prototypes ‘Gesture Group A’ and ‘Gesture Group B’ with different hand gestures for hitting, slicing, and squashing the ball. We then compared and evaluated each group's gestures with each other to determine which gestures users preferred more.

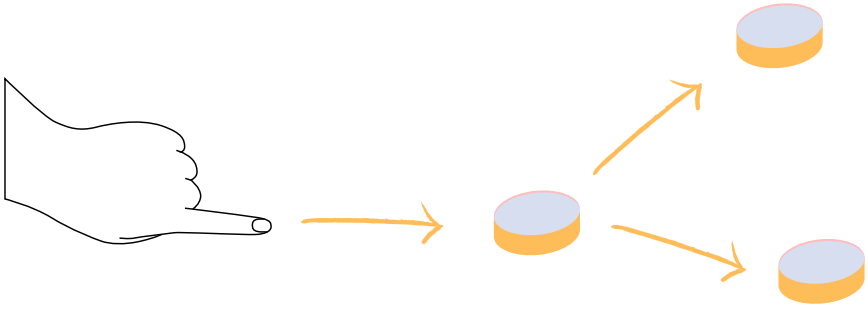
GESTURE GROUP A :

For the **Gesture Group A**, these techniques are as follows:

1. HITTING THE PUCK:



2. SLICING THE PUCK:



Slicing the puck would create clones of the puck.

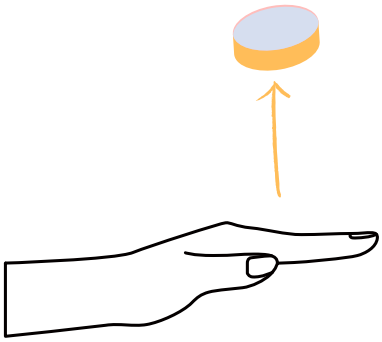
3. SQUASHING THE PUCK:

Squashing the Puck would squash and destroy the puck.

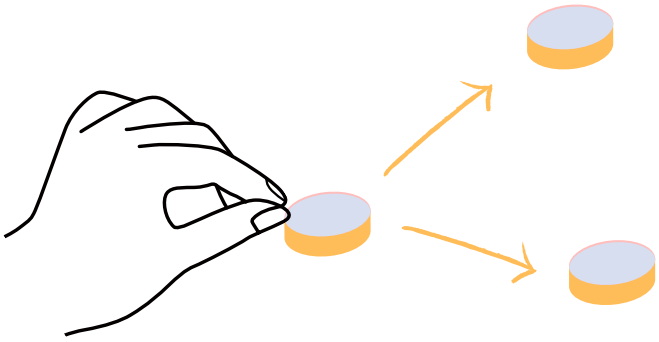
GESTURE GROUP B :

For the **Gesture Group B**, these techniques are as follows:

1. HITTING THE PUCK:

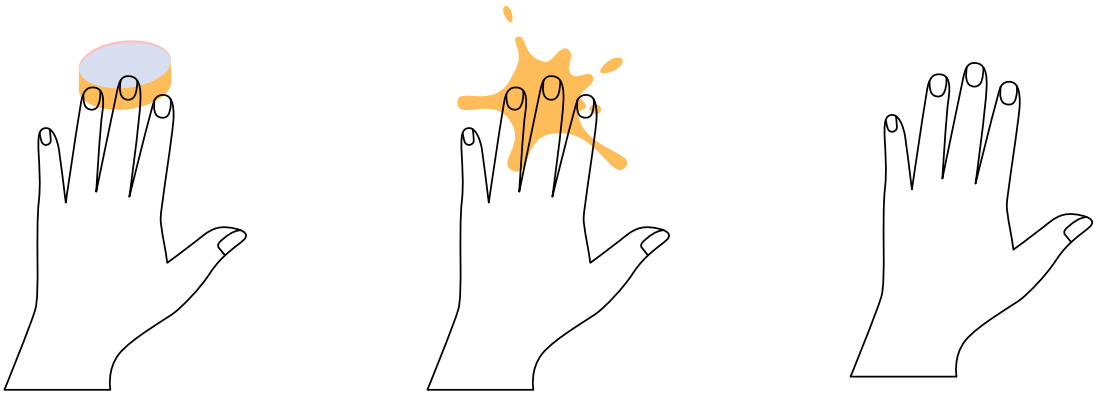


2. SLICING THE PUCK:



Slicing the puck would create clones of the puck.

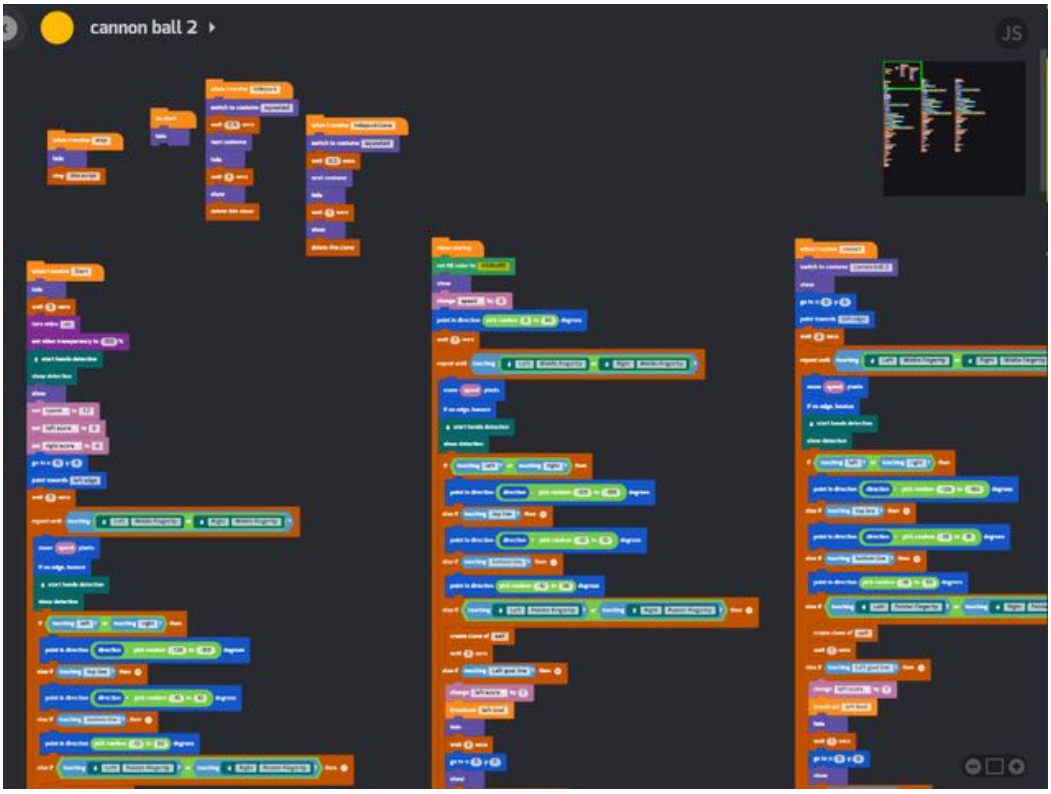
3. SQUASHING THE PUCK:



IMPLEMENTATION

Since we were both designers, so we tried to use a non-technical way of programming. We implemented this using a platform called ‘Tynker’ which uses block-based visual programming. This platform is intended to be used by beginners or junior students to help create logic-building and learn programming concepts. This application is similar to Scratch however, it provides extensive functionalities like AR blocks, Object Detection, and Pose Detection Blocks, etc. It can be used to create VR games as well.

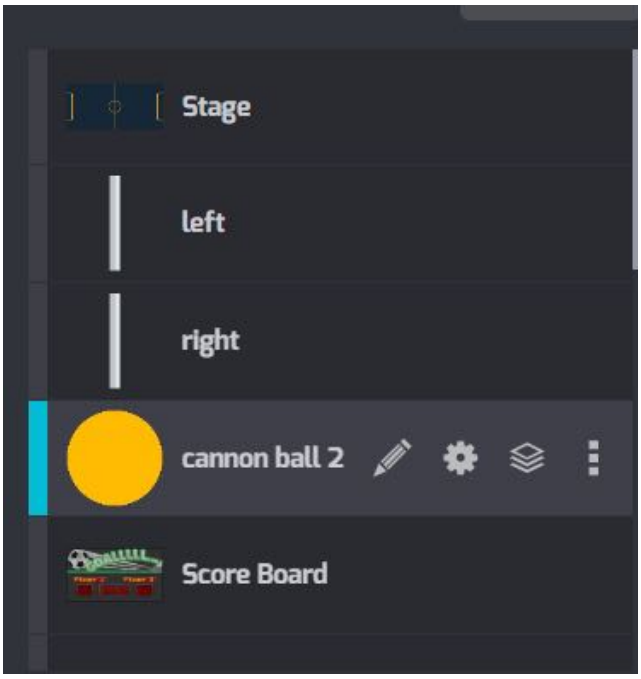
For our prototype, we utilized Pose Detection Blocks which uses PoseNet model at the backend. This model provides 16 hand landmarks from fingertip to wrist for both left and right hands.

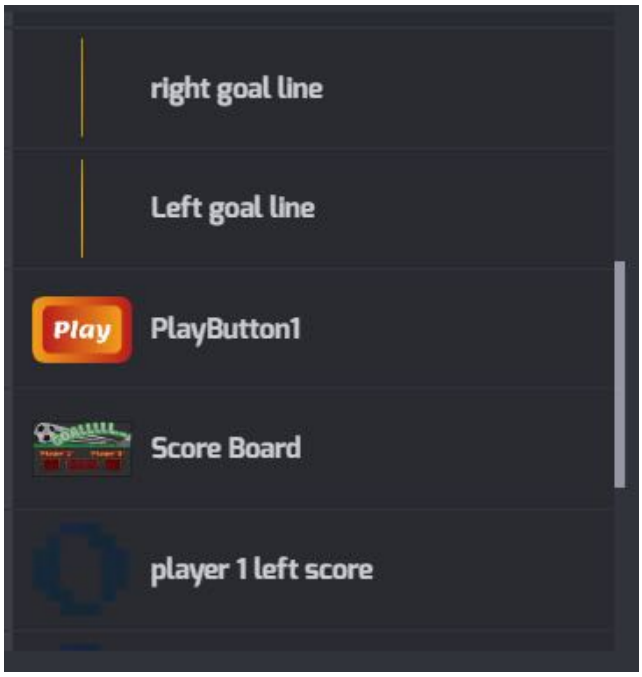


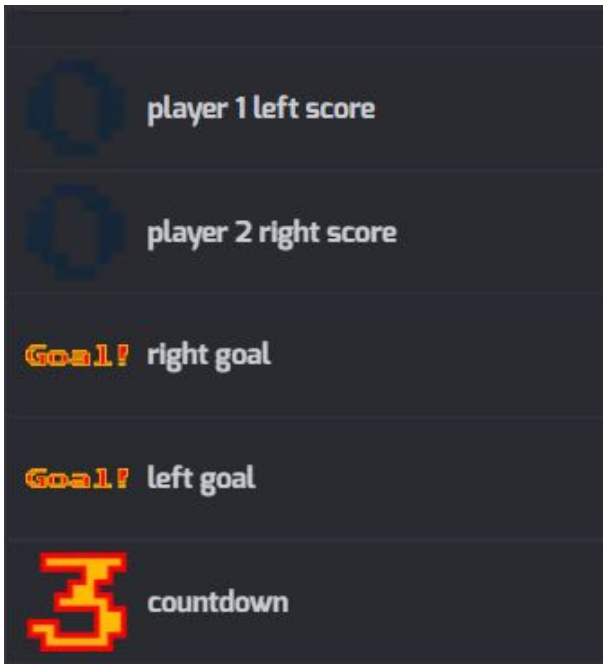
These blocks are then automatically converted into it’s own programming language similar to JS at the backend.

```
this.onMessageReceived("hidepuck", function () {
    this.looks.setCostume("squashed");
    this.wait(0.3);
    this.looks.nextCostume();
    this.looks.hide();
    this.wait(1);
    this.looks.show();
    this.deleteActor();
});

this.onStart(function() {
    this.looks.hide();
});

this.onClone(function () {
    this.canvas.setFillColor("#b3bb00");
    this.looks.show();
    speed += 8;
    this.motion.pointDirection(Math.randomRange(0, 90));
    this.wait(1);
    while (!this.sensing.isTouching((blockAIHandLandmark("Left", 12) || blockAIHandLandmark("Right", 12)))) {
        this.motion.move(this.speed);
        this.motion.bounceOnEdge();
        blockAIStartHands();
        blockAIShowDetection();
        if ((this.sensing.isTouching("left") || this.sensing.isTouching("right"))) {
            this.motion.pointDirection((this.motion.getDirection() - Math.randomRange(-120, -180)));
        } else if (this.sensing.isTouching("top line")) {
            this.motion.pointDirection((this.motion.getDirection() + Math.randomRange(-10, 10)));
        } else if (this.sensing.isTouching("bottom line")) {
            this.motion.pointDirection(Math.randomRange(-10, 50));
        } else if ((this.sensing.isTouching(blockAIHandLandmark("Left", 8)) || this.sensing.isTouching(blockAIHandLandmark("Right", 8)))) {
            this.clone("self");
            this.wait(1);
        } else if (this.sensing.isTouching("Left goal line")) {
            left_score++;
            this.broadcast("left Goal");
            this.looks.hide();
            this.wait(1);
            this.motion.moveTo(0, 0);
            this.looks.show();
        } else if (this.sensing.isTouching("right goal line")) {
            right_score++;
            this.broadcast("right Goal");
            this.looks.hide();
            this.wait(1);
            this.motion.moveTo(0, 0);
```

In Tynker, there are actors instead of components or modules. We created separate actors for their separate functions. Our main actor was the puck where all the detection was executed, another actor was the scoreboard Every time the puck touched the left or right goal, the intended player would get a point. The game score was set to 3.





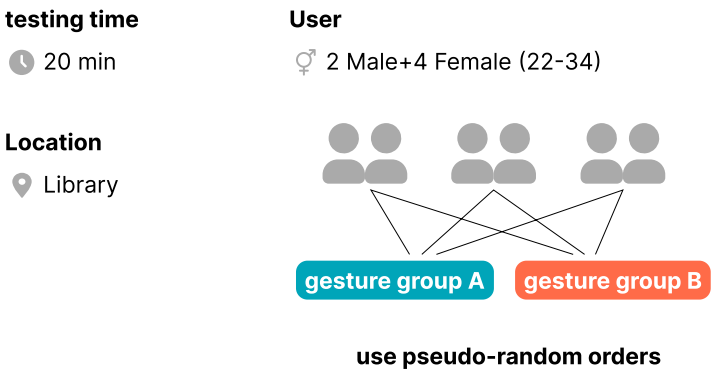


EVAULATION

The goals of our evaluation were to find out about users' preferences for the appropriate interaction gestures for playing the game. we tested two different prototypes with different gestures for each interaction for a within-subject test. Additionally, we had 3 hypotheses as:

- Playing with credit cards is more engaging than playing with hands.
- Gesture Group B is easier to play than Gesture Group A.
- It is fun to play air hockey with gestures.

1. USER STUDY



We conducted a user test with 3 teams of 2 participants each to test our prototype. Our test group consists of four women (age range between 23-30) and two men (age range between 25-32). The user study was carried out in the following order:

1. First, users were informed about the overall game concept and how to play the game.

2. Second, we explained to the users each of the gestures for both prototypes. We started with Gesture Group A for team 1 users and then followed a random pseudo order for the remaining 2 teams. As a first trial, users played the game a few types before becoming familiarized with the gestures.

3. After the users completed the first trial using one of the prototypes, we explained the other prototype gestures to them and they played the game again with those hand gestures interaction.

4. After playing the game using both prototypes, users were asked to complete a questionnaire by filling out an online form.

2. THE QUESTIONNAIRE

As previously stated, we created a questionnaire for users to complete after the test. In this questionnaire, we asked questions that aim to learn users' level of confidence in the interactions, their opinion, and their preferences. These questions are the following:

Do you have experience with air hockey? (yes/no)
It was a yes or no question to analyze the experience level of our test participants regarding the air hockey game.

Do you have prior experience with relevant technology? (2 choices+other)
It was an open-ended question to analyze the test participants' technical knowlege.

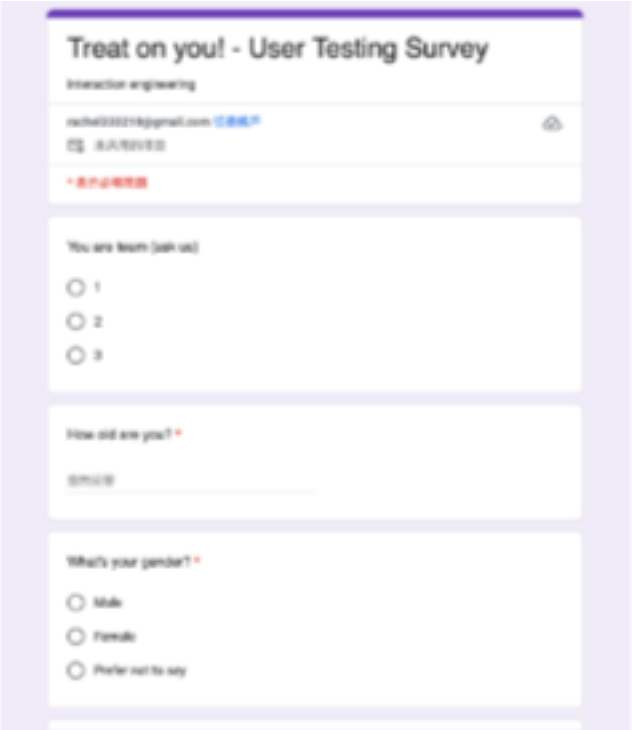
Do you prefer to play with card or without card? (1-5 likert scale)
We provided a 5-point Likert scale from one end to another: "Strongly Disagree" and "Strongly Agree".

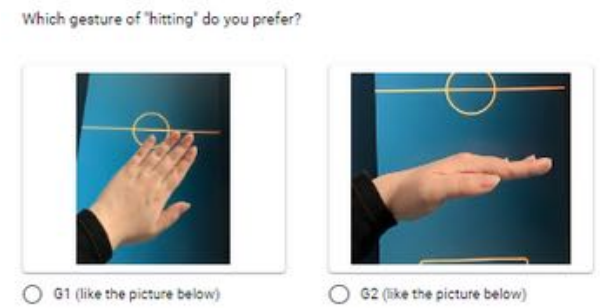
Which gesture of "hitting" do you prefer? (2 choices)
We asked the user to select their preferred gesture of interaction for hitting.

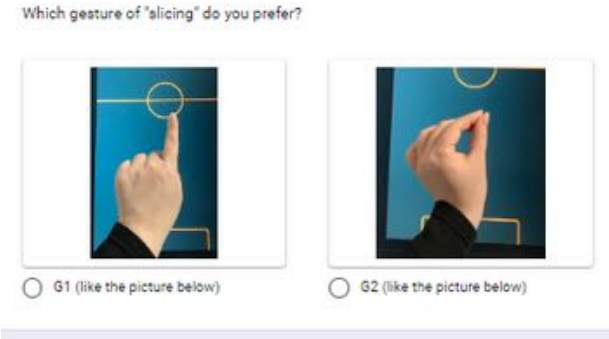
Which gesture of "slicing" do you prefer? (2 choices)
We asked the user to select their preferred gesture of interaction for slicing.

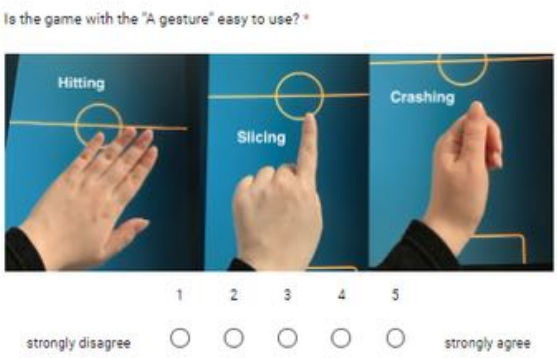
Which gesture of "squashing" do you prefer? (2 choices)

We asked the user to select their preferred gesture of interaction for squashing the puck.









https://docs.google.com/forms/d/e/1FAIpQLSfJKg4Lzegewez0FRsFHngKIh2jdqz7FjXdAnJDzKwzOW9qug/viewform

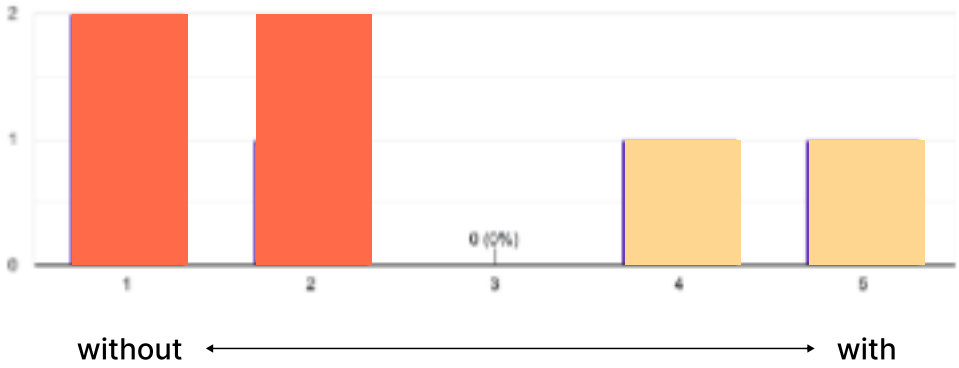
RESULTS

After analyzing the responses from the questionnaire, we were able to prove our hypothesis. To our surprise, one of the hypotheses turned out to be contradicting to our expectation.

01

Playing with credit card is more engaging than playing without credit card.

=> No, **66.6%** users prefer playing without credit card.



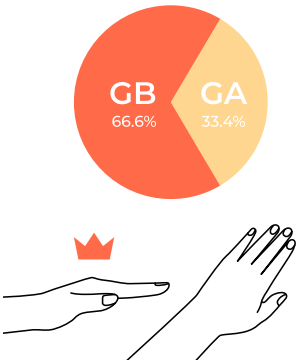
02

Gesture group B is easier to play than gesture group A .

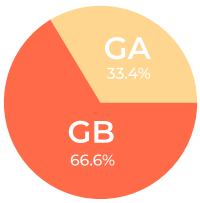
=> Yes, but don't include squashing gesture, it is **50%** each

GA=gesture group A
GB=gesture group B

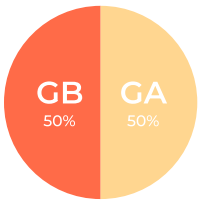
hitting

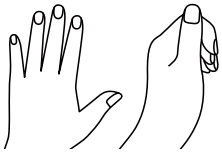


slicing



squashing

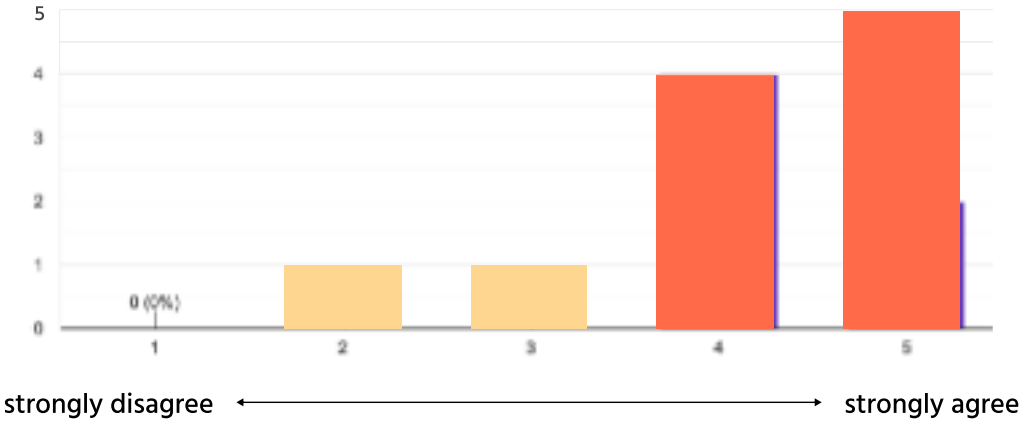




03

It is fun to play air hockey with gestures.

=> Yes, **83%** users rated it more than 4 of 5.



Some Comments from the Users regarding the both Prototypes

“I believe the one with the stick gives me some form of guide and so I know the ball is going to hit the stick with my controls. With the bare hand tracking, I’m not 100% certain I’m going to hit the ball.”

“IIt was interesting, but i felt the pros and cons each game. In general I liked playing with just my hand(first one) but when crashing I liked playing with a card, because it was easy and worked well.”

“It lags a little bit”

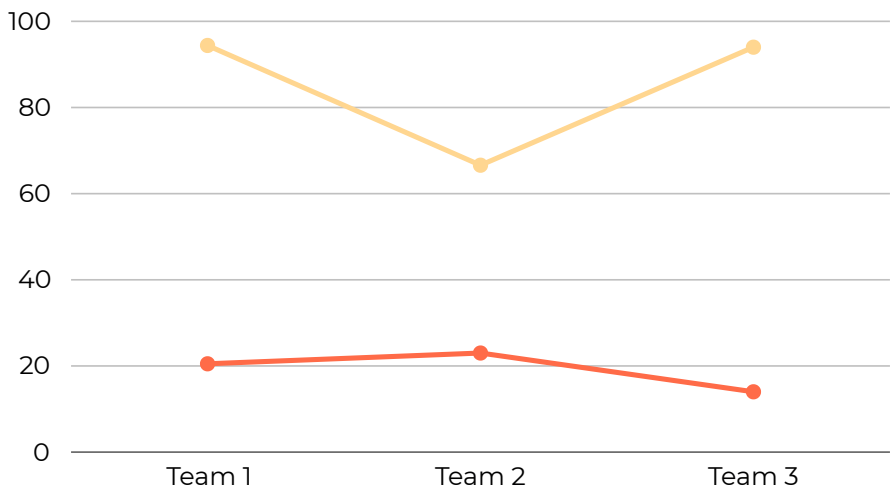
PERFORMANCE METRICS

During the testing phase of each prototype, we also observed some metrics like Success rate and error rate. The Success rate indicates when a correct gesture was performed successfully whereas the error rate indicates the amount of time, the gestures didn't perform at all. The error rate for our game was a little high and the game itself didn't perform too well due to non-technical implementation.

Succes Rate for "Hitting" Gesture

We calculated the success rate for the both hitting gestures in the prototype A and B by taking an average of successful hit divided by the total number of hitting gesture performed.

Avg Success Rate for the Hitting gesture in both prototypes ->
(T1 + T2 + T3) successful hit gesture performed / Total number of times hit gesture performed * 100

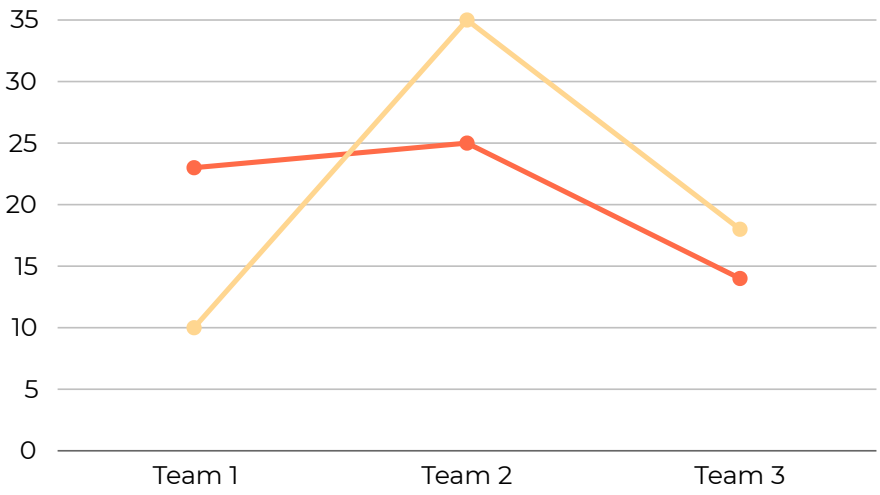


In the above figure, orange line indicates the hitting gesture sucess rate for Gesture Group A and yellow line indicates the hitting gesture sucess rate for Gesture Group B

Succes Rate for “Slicing” Gesture

We calculated the success rate for the both slicing gestures in the prototype A and B by taking an average of successful slice gesture divided by the total number of slicing gesture performed.

Avg Success Rate for the slicing gesture in both prototypes ->
(T1 + T2 + T3) successful slice gesture performed / Total number of times slice gesture performed * 100

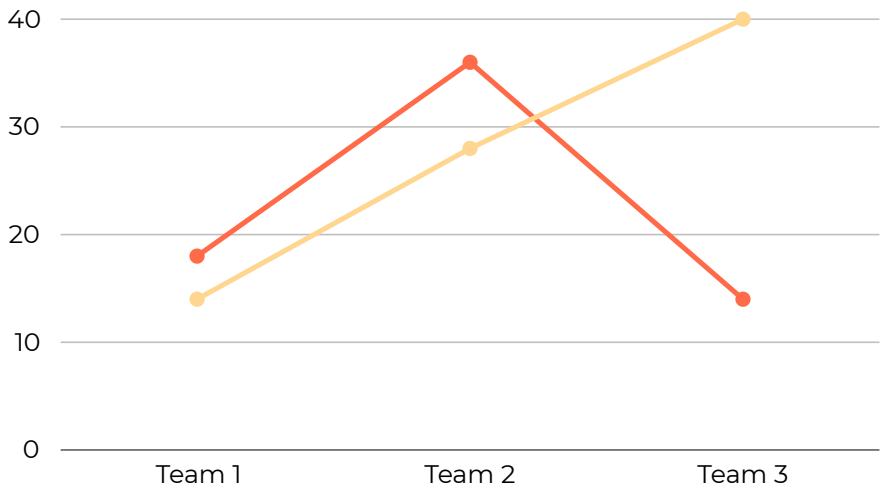


In the above figure, orange line indicates the slicing gesture sucess rate for Gesture Group A and yellow line indicates the slicing gesture sucess rate for Gesture Group B

Succes Rate for “Squashing” Gesture

We calculated the success rate for the both slicing gestures in the prototype A and B by taking an average of successful slice gesture divided by the total number of slicing gesture performed.

Avg Success Rate for the slicing gesture in both prototypes ->
(T1 + T2 + T3) successful slice gesture performed / Total number of times slice gesture performed * 100

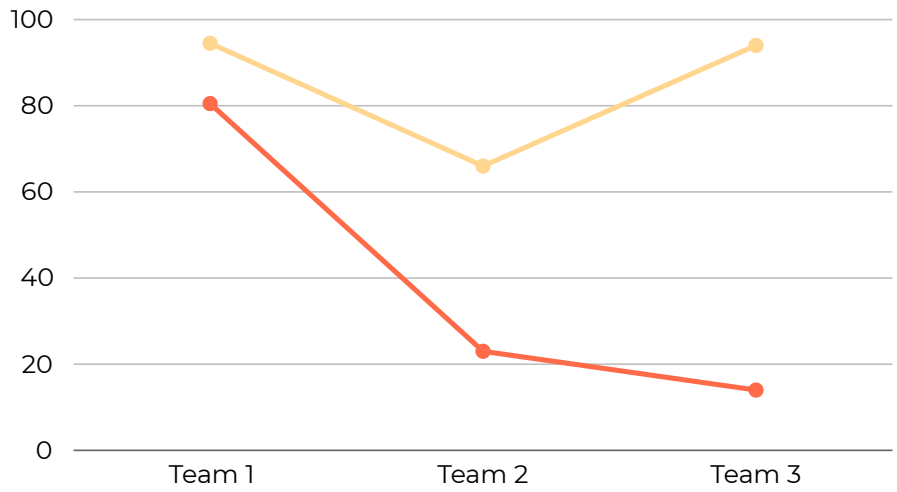


In the above figure, orange line indicates the squashing gesture sucess rate for Gesture Group A and yellow line indicates the squashing gesture sucess rate for Gesture Group B

Error Rate for “Hitting” Gesture

We calculated the error rate by subtracting the success rate by 1 as the error rate was the opposite of the success rate.

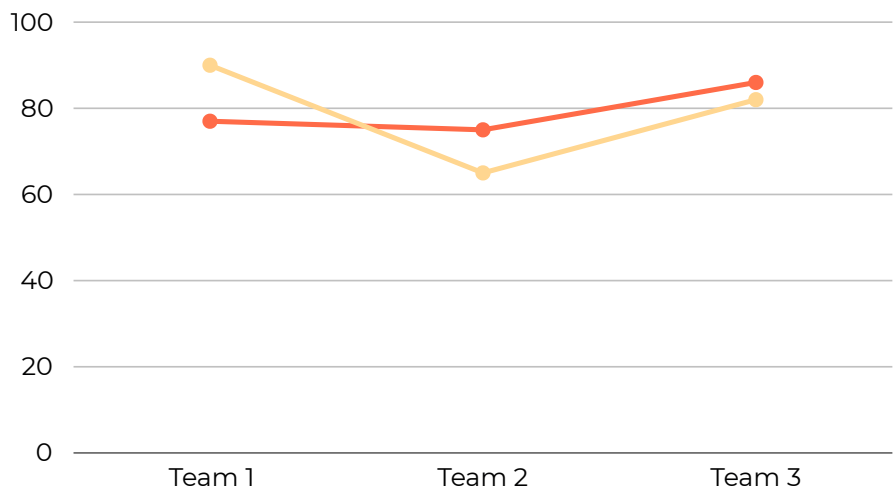
Avg Error Rate for the slicing gesture in both prototypes -> 1 - (T1 + T2 + T3) successful slice gesture performed / Total number of times slice gesture performed * 100



In the above figure, orange line indicates the hitting gesture error rate for Gesture Group A and yellow line indicates the hitting gesture error rate for Gesture Group B

Error Rate for “Slicing” Gesture

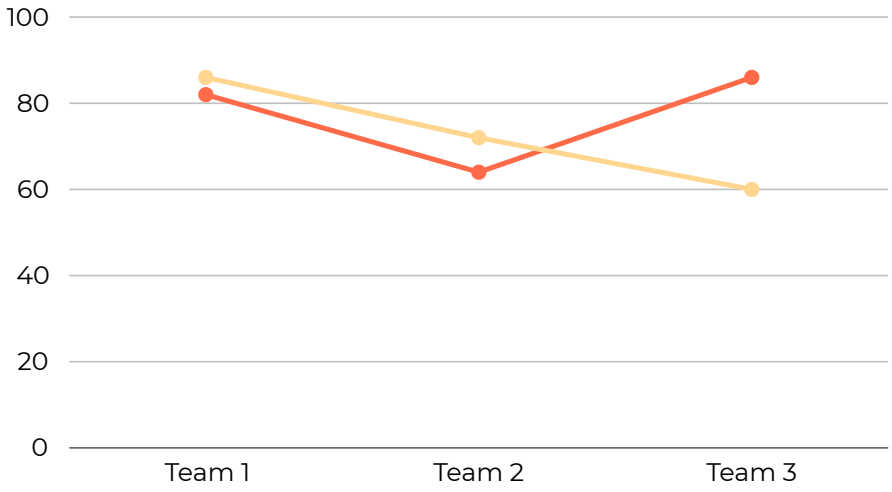
Avg Error Rate for the slicing gesture in both prototypes -> 1 - (T1 + T2 + T3) successful slice gesture performed / Total number of times slice gesture performed * 100



In the above figure, orange line indicates the slicing gesture error rate for Gesture Group A and yellow line indicates the slicing gesture error rate for Gesture Group B

Error Rate for “Squashing” Gesture

Avg Error Rate for the squashing gesture in both prototypes -> 1 - (T1 + T2 + T3) successful squash gesture performed / Total number of times squash gesture performed * 100



In the above figure, orange line indicates the squashing gesture error rate for Gesture Group A and yellow line indicates the squashing gesture error rate for Gesture Group B

CONCLUSION

Treat On You has proven to be an interesting game concept capable of improving traditional air hockey through novelistic interaction methods, by making them more entertaining and challenging. According to the user test, even though users played the game better with Group B gestures, eventually became the preferred gestures.

However, some aspects of this field should be improved:

- The slicing gesture is often confused with the squashing gesture as the logic behind these gestures is that if the ball touches the pointer finger and thumb finger together whereas for the squashing gesture, when it touches the middle finger, so one of our goals could be to increase the accuracy rate of these gestures by implementing an external library to better map with our game.

- Another potential area of improvement is optimizing the game as it lagged a lot.

- Our game was played on a small monitor screen. a bigger monitor can be replaced for more space for playing.

Overall, with given improvement perspectives and additional features, the project has the potential to offer an immersive game environment.