Pose Royale

Cognitive Interaction with Robots

Giacomo Schiavo - Filippo Venturini - Giacomo Sanguin

January 2024

Contents

Chapter 1

Abstract

The idea behind our project is to design and implement a game in which players can have fun while performing various poses in front of their PC's webcam.

In detail, the game interface will display a pose for the player to mimic, along with a timer. The player must accurately emulate the pose before the time runs out.

The game will be structured into different levels with increasing difficulty. The rise in difficulty will challenge the player with more complex poses and less time to execute them.

Further analysis of player engagement is conducted using data extracted from the game, supplemented by information gathered from player surveys.

Chapter 2

System Requirements

2.1 Pose Royale: the idea

"Pose Royale" is a game that challenges players to push the limits of their coordination and timing, aiming to achieve a better score than others while having fun together.

The game is inspired by "Hole in the Wall" games [6], where participants are required to fit through a specific hole in a wall by performing a pose.

In detail, the player should first calibrate the distance from the webcam to ensure accurate gameplay. When the game starts, the player progresses through three levels: Easy, Medium, and Hard.

The system displays the pose the player needs to perform, and a timer counts down the seconds the user has to accurately emulate the pose.

Each level features has its own set of poses, increasing in difficulty, and the player has less time to perform the poses as the difficulty increases.

For every correct pose executed with the correct timing, the player earns one point. The ultimate goal is to attain the highest possible score.

2.2 Requirements Analysis

Given the description of the game provided above, we proceed to analyze the requirements that the system must satisfy.

- **Webcam integration**: The system must integrate the webcam to allow the player to interact with it.
- **Various poses with different difficulty**: The game should provide different kinds of poses with varying difficulty levels to challenge the player during different levels.
- **Pose Recognition technique**: The system must incorporate a Pose Recognition technique to detect the poses performed by the player during

the game.

- **Pose Detection algorithm**: There must be an algorithm that compares the pose performed by the player with the correct one to assess it.
- **Levels structure**: The game should be structured into different difficulty levels to be more challenging and enjoyable.
- **Timer mechanism**: There must be a timer logic to regulate the game progression.
- **Scores logic**: The game must include a logic to count the correct poses performed by the player and assign a score.

2.3 Human Robot Interaction

Following the description of the idea and the requirements of the system, let's now focus on the interaction between the human and the robot in Pose Royale.

In this system, the robot is a PC/laptop equipped with a **webcam**. Other requirements include a browser and an internet connection to connect the device to the game. An optional desk is recommended to place the webcam at a reasonable height for an enhanced gameplay experience. The simplest way to meet these requirements is, of course, a laptop or a desktop computer. However, any type of hardware that satisfies these specifications can be used for gameplay. This includes smartphones, tablets or any king of similar devices.

In this scenario, the human takes on the role of a player and can interact with the robot using a webcam. The interaction aims to provide the player with feedback, informing them of the results of their actions. This feedback will be delivered through visual displays.

Last but not least, in this context, the interaction with the robot is intended to make the player happy and satisfied.

2.4 Target Users

After analysing the requirements of our system, we're going to explain what are the possible target users of Pose Royale. This phase is really important because the users that will test our game will belong to one or more of the following categories.

- **Gaming Enthusiasts**: Video game enthusiasts who appreciate interactive and competitive gaming experiences. The incorporation of physical movement adds a unique dimension to their gaming preferences.
- **Families and Friends**: Groups of friends or family members looking for a entertaining and active social activity.

In both cases, the game is restricted to players taller than 150cm.

2.5 Research questions

The primary focus of our project lies in creating an immersive and joyful gameplay experience, where players are not mere spectators but active participants. They will immerse themselves in a dynamic environment, in which they will have to coordinate body movements, intuition and agility, in order to correctly overcome the various levels presented.

To comprehensively evaluate the effectiveness and user experience of our game, we have formulated several research questions. These questions delve into critical aspects of the development, spanning from the technical aspects of pose recognition to the psychological analysis of user engagement.

2.5.1 User Experience and Engagement

"Is the game capable of satisfying the player, guaranteeing a pleasant and enjoyable experience?"

Null Hypothesis (H0): There is no significant difference in the game's capability to satisfy the player, and it does not guarantee a pleasant and enjoyable experience.

Alternative Hypothesis (H1): The game is capable of satisfying the player, guaranteeing a pleasant and enjoyable experience.

2.5.2 Impact of difficulty level

"What is the perceived level of difficulty among players while engaging with the pose recognition game, and how does this perception correlate with their overall satisfaction and performance in the game?"

Null Hypothesis (H0): There is no significant correlation between the perceived difficulty level among players in the pose recognition game and their overall satisfaction and performance in the game.

Alternative Hypothesis (H1): There is a significant correlation between the perceived difficulty level among players in the pose recognition game and their overall satisfaction and performance in the game.

2.6 Variables

In this section we are going to analyze for each of the two research questions described above, the dependent and independent variables, to better understand what is the focus of our study during the design and the development of Pose Royale.

2.6.1 User Experience and Engagement

Dependent Variable:

- **User Satisfaction and Engagement** (measured through a scale, rating, or qualitative feedback)

Independent Variables:

- **Game Features**: Various aspects of the game, such as interface design, sound effects, graphics, etc.
- **User Interaction**: How users interact with the game, including controls and feedback mechanisms.
- **Game Dynamics**: The overall flow, pacing, and challenges presented in the game.

2.6.2 Impact of difficulty level

Dependent Variable:

- **User Perception of the level of difficulty**: measured through a scale from 1 to 5, this is the subjective assessment by players of how challenging or difficult they find the game

Independent Variables:

- **Overall Satisfaction**: This variable represents the participants' satisfaction with the game, and it serves as one of the factors that may influence or be influenced by the perceived level of difficulty.
- **User Performance in the game**: This variable represents how well players perform in the game, specifically in terms of correctly executing poses.

Chapter 3

Description

3.1 Technologies

In this section we are going to analyse what technologies we used and why are necessaries for the development of Pose Royale.

3.1.1 Google MediaPipe

Google MediaPipe is an open-source framework developed by Google for building real-time applications with computer vision and machine learning [5]. It offers pre-built solutions for tasks like hand tracking and pose estimation, and it is therefore the main technology at the base of our project, permitting an extremely accurate **pose detection and recognition**.

3.1.2 React

React is a JavaScript library for building user interfaces, maintained by Facebook [9]. It allows developers to create interactive and reusable UI components, facilitating the development of single-page applications. It employs a virtual DOM for efficient updates and enhances performance. We choosed React as a framework for implement Pose Royale as a **web application**.

3.1.3 GIMP

GIMP is a free and open-source raster graphics editor [3]. It provides a robust set of tools for tasks like photo retouching, image editing, and graphic design. In our project, it was used for the **creation of the poses** that the player has to reproduce.

3.1.4 Figma

Figma is a powerful cloud-based design tool specifically crafted for collaborative interface design and prototyping [2]. It serves as a centralized platform where design teams can create, edit, and iterate on user interfaces in real-time. We used Figma for design **Pose Royale UI** via **mockups**.

3.1.5 Netlify

Netlify is a development platform that includes build, **deploy**, and serverless backend services for web applications and dynamic websites [8]. It's essential for make Pose Royale playable just by the URL address of the application.

3.1.6 GitHub

GitHub is a web-based platform for **version control** using Git [4]. It enables developers to host, review, and collaborate on software projects and it is then nowadays, a fundamental tool to simplify and make development more efficient. We used GitHub for **collaborate and work as a team** during the deployement of Pose Royale.

3.2 Software design

3.2.1 Game design

Pose Royale is a React-based web application that provides users with an interactive experience focused on various poses. The game is structured with three distinct levels, each containing a set of poses that players must successfully achieve within a predefined time frame.

The game includes three levels: **easy**, **medium**, **hard** and each pose increases with the difficulty's level. The poses are separated as follows:

- **Easy**: warrior pose (wPose), tree pose (oPose), double hand pose (doubleHandPose), nail pose (nail);
- **Medium**: incline left pose (inclineLeft), incline right pose (inclineRight), star pose (star);
- **Hard**: crab pose (crab), right leg raise pose (rightLegRaise), balance right pose (balanceRight).

For each game, there's a different timer to perform the pose: after the completion of the timer, the detection algorithm checks the player's pose and decides whether to award a point or not.

For easy difficulty, the timer is set to 5 seconds, 7 seconds for medium difficulty, and 9 seconds for hard difficulty.

The game starts with a "Tutorial" serving as a calibration phase for the player. Following a small countdown, the game starts with the set of easy

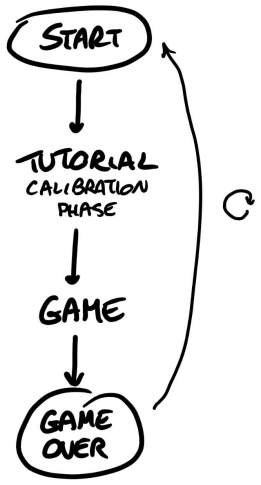


Figure 3.1: Game structure

poses, presented consecutively. After completing all the poses of one level, the poses of the next level are loaded immediately. After the last level, there's a Game Over screen and the average accuracy is printed in the web console. This metric will be used later for analysis. To play another match, it is necessary to reload the page.

3.2.2 Implementation

The entire game operates through React's mechanics, with the main component being the App component which handles the routing system. For this project there are two main pages: the Home Screen and the Game Screen. We can ignore the first page and focus on the second one, where the game lies.

The game is structured in the class *GameController* that has all the methods and properties to manipulate the game. Within the *GameController* there are 3 instances of the *Game* class, each featuring different poses and different timers for pose execution. To establish a seamless connection between poses and games, we created a *gameConfig* file containing all these information so to ensure an efficient and organized representation of the game structure.

The dynamic aspects of the game are overseen by the *GameManager*: its role is to call *GameController*'s functions so to maintain proper game behavior and keeps memory of all game's states.

The *GameManager* functions also help the communication between Canvas and Video components, capturing webcam frames, sending them to the Pose Landmarker API and filtering out the necessary body landmarks.

The API's response provides all body landmarks, from which only the essential ones are filtered. The rendered content on the screen is managed entirely

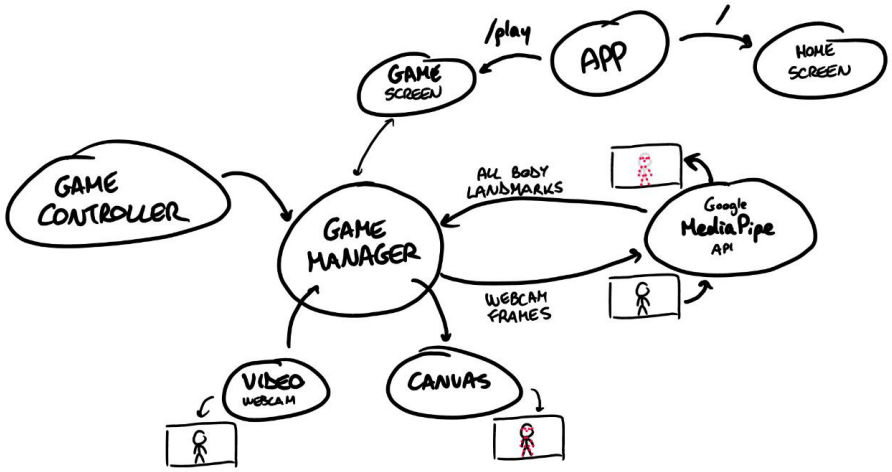


Figure 3.2: App structure simplified

by the HTML Canvas component.

The images and coordinates for the poses are stored in the *poses* folder while additional UI components and helper functions reside in the *utils* folder.

The approaching wall (pose image) is animated using CSS properties and a cubic-bezier timing function. This curve makes the animation fast at the start and slower at the end, allowing players sufficient time to refine their poses.

3.2.3 Pose sampling

While some members of the team focused on software design and implementation, others concentrated on the pose sampling aspect.

A crucial component of Pose Royale comprises the set of poses that users need to emulate. Once the application prototype was ready, we utilized Google MediaPipe's Pose Landmarker's API [5] to sample the coordinates of the joints for the correct poses, automatically generating a JSON [7] file for each pose.

We have identified some key landmarks for accurate pose calculation. These include the nose, right and left indexes, right and left elbows, right and left shoulders, right and left hips, right and left knees and right and left ankles. These selected points serve as reference markers in determining the precision of the poses.

From these JSON files, we employed GIMP [3] to create a graphical representation of the poses. This representation will be displayed during the game and utilized to verify the correctness of the user's pose.

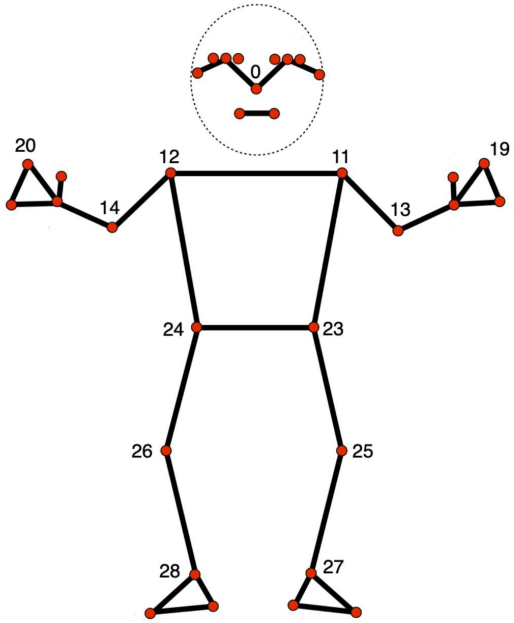


Figure 3.3: Chosen landmarks for our game (numbered dots)

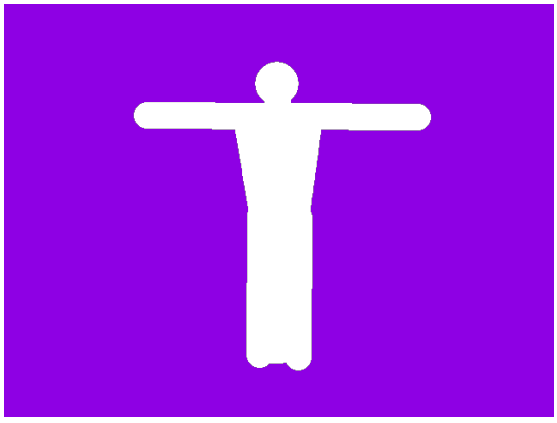


Figure 3.4: Pose graphic produced

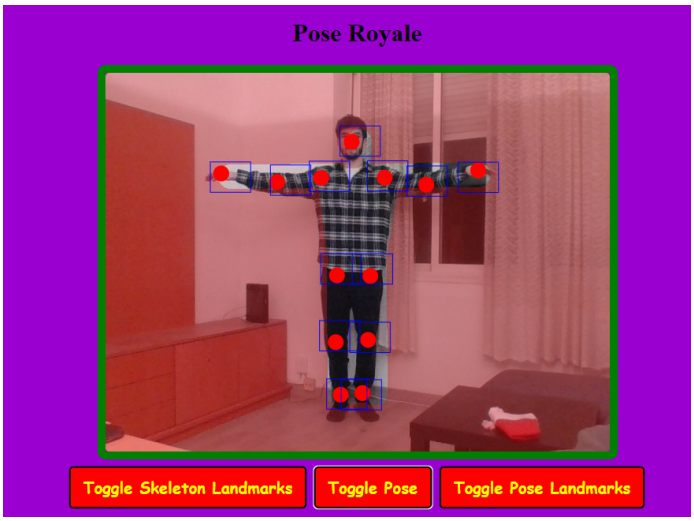


Figure 3.5: Detection algorithm visualized in previous version of the game

3.2.4 Detection algorithm

The detection algorithm works in this way: for each point we set an error threshold starting from the perfect point (model point); if all the filtered points of the player reside inside the error radius, then the pose is considered correct (see figure 3.5). The accuracy of a pose is calculated based on the number of points that reside inside the error radius, divided by the total number of points. The error threshold's value is chosen arbitrarily. A higher value implies an easier game while a lower value implies more strictness.

3.2.5 User Interface design

The team designed a User Interface take inspiration by some well known game. We choose font colors and styles for make the user feel happy and enjoy the experience of play Pose Royale, keeping at the same time an high Usability of the game.

The interface is composed by the following elements:

- **Pose to perform:** the silhouette of the pose that the user has to perform;
- **Difficulty of the pose:** level of difficulty of the pose to perform (Easy, Medium or Hard) indicated by the label on the lower right part of the screen and the border of the image;
- **Score:** the number of correct poses performed by the user;
- **Timer:** timer that show the time remaining to perform a pose;

- **Pose Checker:** the emoji help the player to understand he/she is performing the correct pose;
- **Award Emoji:** this emoji will be shown only at the final second and indicates whether the player is awarded with a point or not.

The UI interface can be see in figure 3.6.

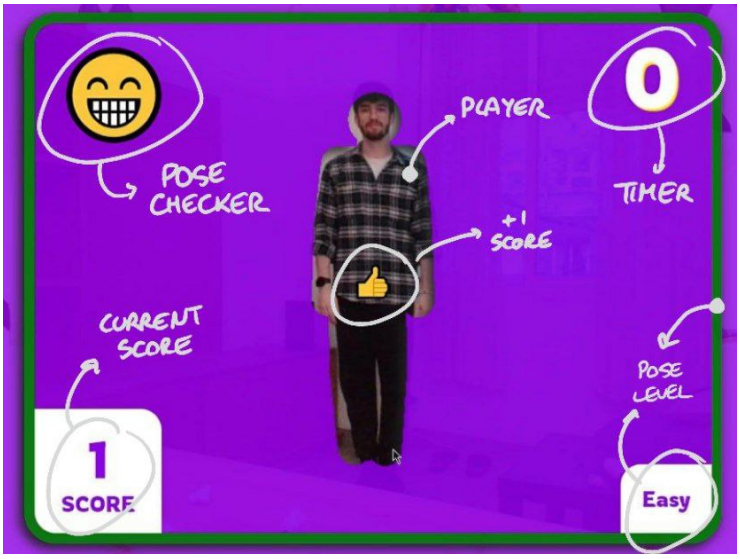


Figure 3.6: Pose graphic produced

Chapter 4

Tests

4.1 Testing Methodology

In this section, we will explain the methodology applied to test the results obtained. The first important consideration is the implementation of a prototype, which is essential to check the current state of the system and verify that the core functions of the software are working.

4.1.1 Prototype testing

In our case, the prototype consists of a web application in which the **Pose Detection** and **Pose Recognition** algorithms are correctly implemented. To test these features, only the developers of the team took on the roles of testers (we also belong to the Game Enthusiast target user category). The final users will be involved only during the testing phase of the complete game, in which we can also evaluate the game mechanisms and the user experience.

During this phase, every member of the team first checked if the system detects correctly every joints of the body, also using different types of PCs and webcam. After that everyone tried to perform the pose displayed in the prototype, and checked if the recognition works correctly.

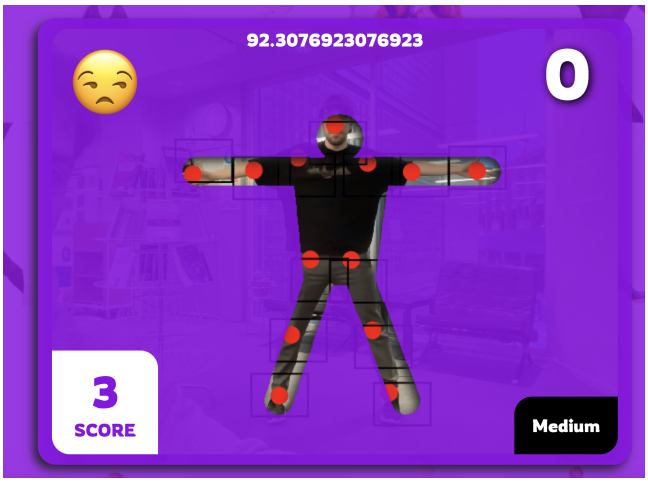


Figure 4.1: Prototyping on previous version of the game

4.1.2 Final game testing

For the final version of the game, testing must be more structured and should obviously include users who belong to one or more of the target user categories (see Section 2.2). In this testing phase, we must consider not only the technical aspects of the game, as in the prototype testing, but also features related to the user experience and the users' interactions with the system.

Firstly, we incorporated into the game some mechanisms to retrieve data that will be used for the evaluation of Pose Royale, such as the players' scores or the number of correct poses for each game, among other metrics.

Once the mechanisms for collecting data were set up, we organized different testing sessions with users from the target user category. Most of them are students of our age, with some belonging to the Game Enthusiast category, while others are just a group of friends looking to have fun.

After each session, given that we made Pose Royale publicly accessible using Netlify [8], we sent the link to other potential target users to gather as much data as possible. They played remotely, even without being present at the testing session.

Following each user's gameplay, we asked them to complete a closed-ended survey to obtain qualitative data about their experience with Pose Royale. In the next section, we will describe in more detail the evaluation methodology we applied and how the data are used to gather information.

4.2 Evaluation Methodology

In this section, we will explain in detail the evaluation methodology applied to assess Pose Royale. As described in the previous section, the team collected

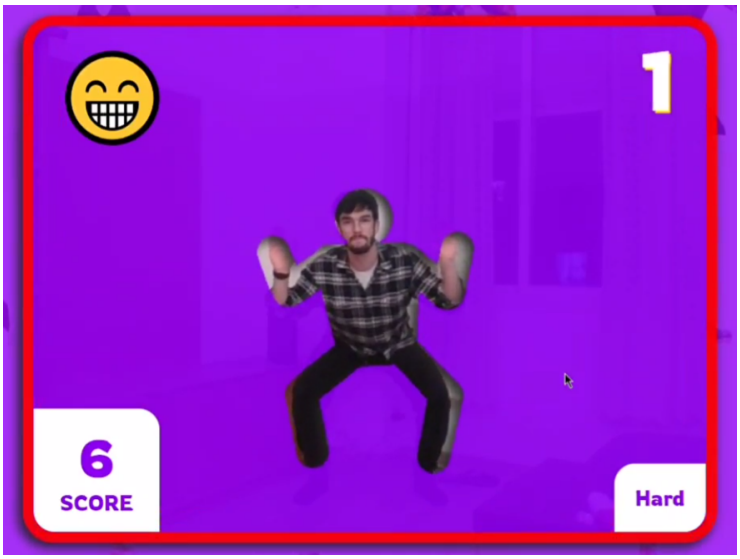


Figure 4.2: Final game version

data using various strategies during the test sessions with users.

The evaluation of Pose Royale is grounded in the research questions outlined in Section 2.4, and our evaluation strategies are fundamentally based on both qualitative and quantitative data.

4.2.1 Qualitative

Qualitative evaluation is conducted by having the user complete a post-experience survey, from which we gather information to address the research questions described in the sections before.

In detail, we aim to evaluate the **User Experience and Engagement** by investigating the following topics:

- **Users' General Enjoyment:** Assessing how the game makes the player feel, as providing entertainment and joy is the primary objective of Pose Royale.
- **Game difficulty:** Evaluating how the player rates the game in terms of difficulty. This aspect is crucial for balancing the game's parameters to ensure an engaging yet not overly challenging experience.

We also seek to evaluate the effectiveness of **Feedback Mechanisms** by understanding users' opinions on the following items:

- **Calibration phase:** This phase aims to help the player find the correct position to play and feel comfortable with the game setup.

- **Pose checker feedback:** Assessing the necessity of providing easy and fast feedback to the user, if he is reproducing the pose in a correct way or not. A way of giving a little help to the player.

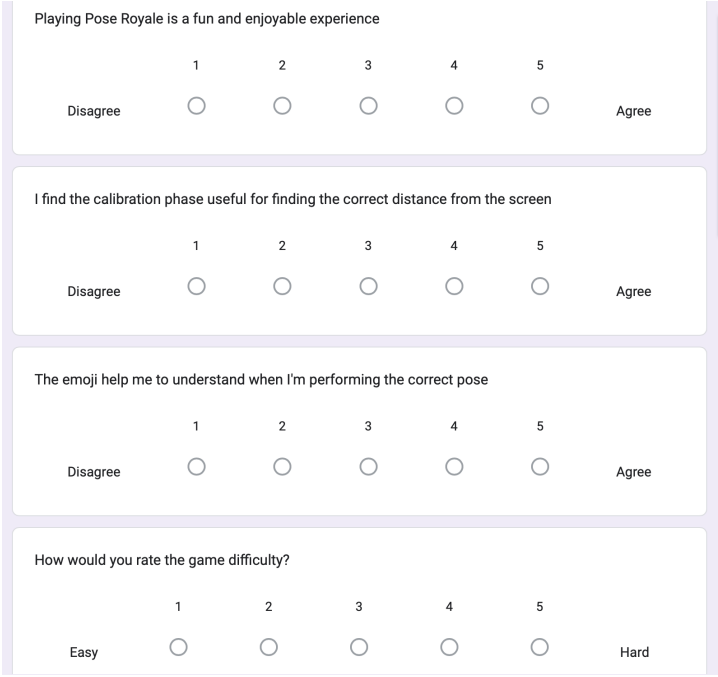


Figure 4.3: First part of the survey

4.2.2 Quantitative

To assess the efficacy of Pose Royale in terms of quantitative metrics, we have implemented a data retrieval mechanism within the game's source code. This mechanism allows us to gather relevant information that directly addresses our research questions, especially regarding the **game's difficulty**.

We focused, in particular, on the following two metrics:

- **Final Score**: an integer value, ranging from 1 to 10, representing the accurate replication of poses by the player. The score is determined by the number of correctly executed poses.

- **Average Accuracy**: Our detection technique is based on having one square for each joint of the correct pose. To perform a correct pose, a player has to fit all the squares. Therefore, we can assess the user's

accuracy in performing a pose by calculating the ratio between the correct squares and all the squares of the pose.

Given this information, in the next section, we are going to explain how we elaborated those to retrieve some information and visualize it by plotting graphs.

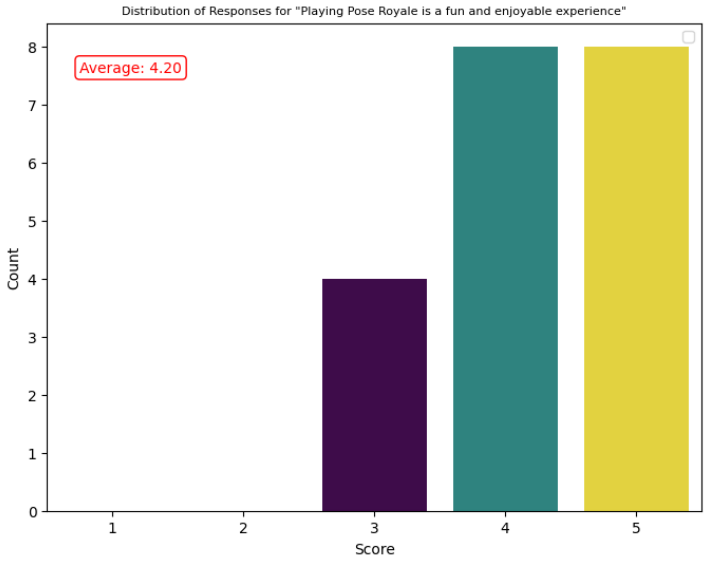
4.3 Results

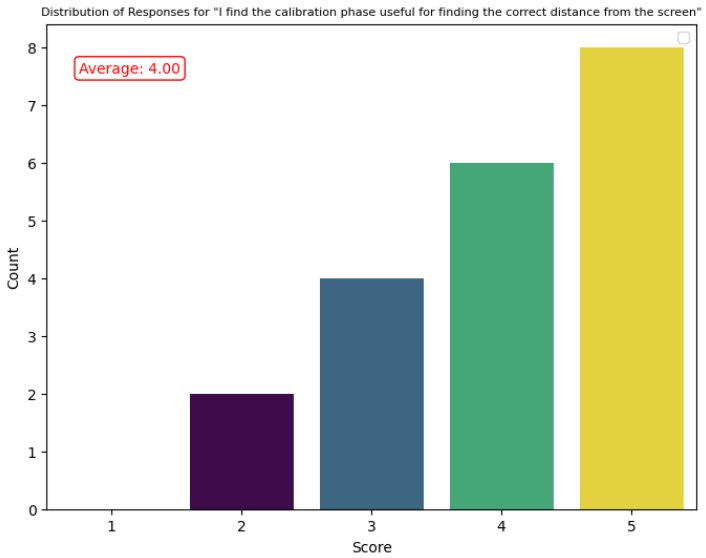
During the testing phase, we gathered data from 20 players, a quantity we deemed suitable and ample for our research objectives, considering the two research questions that we proposed. Subsequently, we systematically acquired both qualitative and quantitative data from each player, and stored them in a dataset.

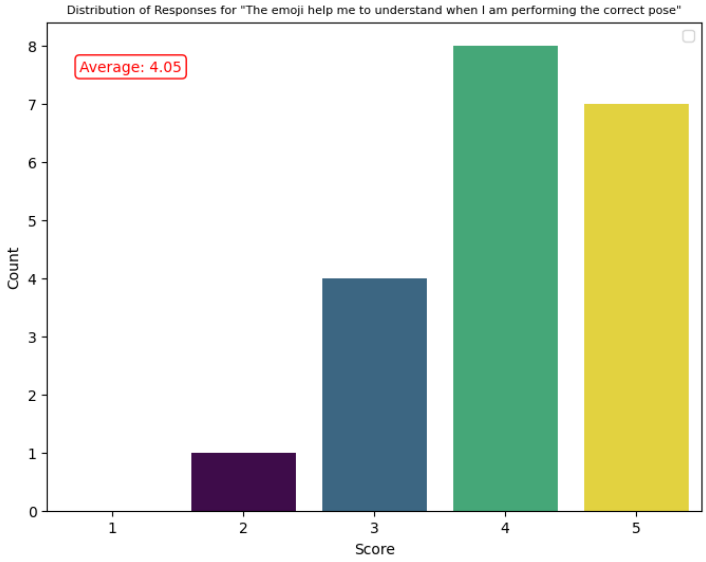
After this phase, we started looking into the data, in order to operate a deeper analysis. First of all we focused on the first research question, and in doing so we analyzed the answers to the survey, in particular the answers to the three following questions, which are related to the user experience and his general enjoynment in playing the game:

- "Playing Pose Royale is a fun and enjoyable experience"
- "I find the calibration phase useful for finding the correct distance from the screen"
- "The emoji help me to understand when I'm performing the correct pose"

The results obtained are shown in the following three graphs:



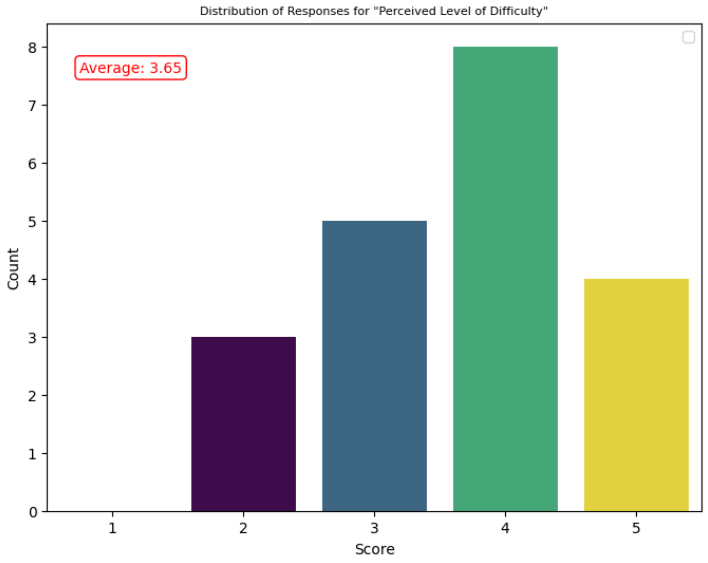


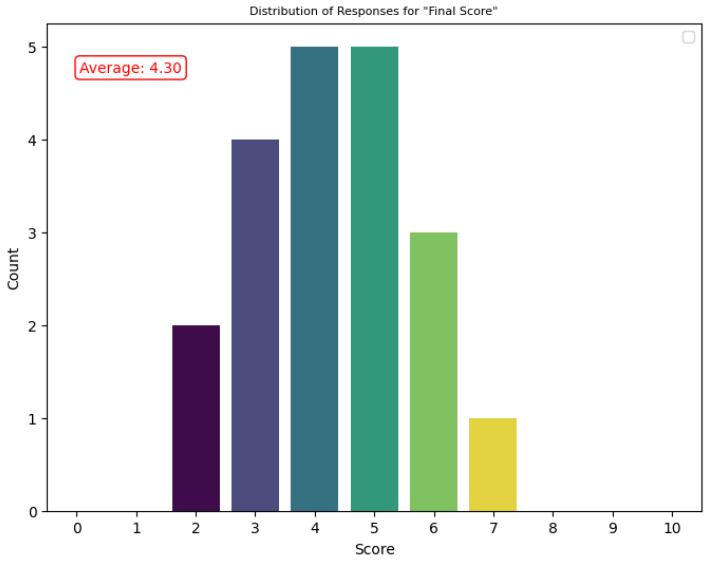


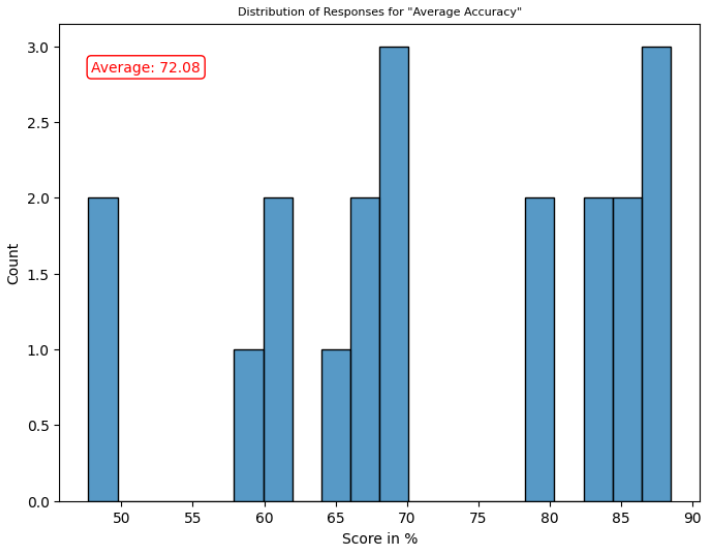
As we can observe, the results are extremely satisfactory, since all participants expressed a positive feedback on the gaming experience and on the integrated features. Specifically, the first question regarding the overall enjoyment, indicates that no participant found the game lacking in entertainment value. Notably, the average score of 4.2 out of 5 underscores the highly favorable reception. Furthermore, the implemented calibration phase and the guidance provided by the emoji throughout the game, designed to enhance user experience, both garnered an average score exceeding 4 out of 5. This positive reception affirms that our conceptualization and implementation were well-received. While rec-

ognizing that there is room for improvement, the current results are undeniably encouraging.

Our analytical focus then shifted to the second research question, regarding the level of difficulty of the game. Initially, we plotted bar charts to visually represent the perceived level of difficulty, as indicated by responses to the question "How would you rate the game difficulty?" in the survey. Additionally, we plotted bar charts for the quantitative data, including Final Score and Average Accuracy, to effectively showcase the results derived from the testing session:







The findings from our analysis reveal intriguing insights. The average final score, standing at 4.30 out of 10, indicates that the game poses a significant challenge for players. Notably, none of the 20 participants achieved a perfect score of 10, at least not on their initial attempt. The Perceived Level of Difficulty reported by players further reinforces this observation, with an average rating of 3.65 out of 5. Remarkably, none of the players perceived the game as "easy". The Average Accuracy, even if a score higher than 70% can be considered as high, it determines the opposite, since in the single pose, having an accuracy lower than 100% means that the player failed that specific pose.

One compelling approach for analysis emerges when examining the interplay between the Final Score and the Perceived Level of Difficulty. Essentially, we seek to determine whether there exists a correlation between a player's performance and their subjective assessment of the game's difficulty. This exploration holds the potential to inform adjustments aimed at optimizing the difficulty level, thereby enhancing user engagement and satisfaction. Presumably, a player's perceived difficulty should align with their performance; any disparities may signal areas where adjustments in game design are warranted. To do so, we decided to use the **Spearman rank correlation coefficient**, (ρ), a non-parametric statistical measure that assesses the strength and direction of the monotonic relationship between two variables. It is a strong type of test, since it works with ordinal or ranked data and is robust to outliers. The procedure involves a specific formula to compute the coefficient: $\rho = 1 - \frac{n(n^2-1)}{6\sum_i d_i^2}$.
The resulting coefficient ranges from -1 to 1, where a positive ρ implies a monotonic positive relationship, a negative ρ indicates a monotonic negative relationship, and ρ close to 0 suggests a weak or no monotonic relationship.

Setting up the hypotheses is a crucial step in the process. In our case:

- **Null Hypothesis (H_0):** There is no monotonic relationship between the

perceived difficulty level and the score.

- **Alternative Hypothesis (H_1):** There is a monotonic relationship between the perceived difficulty level and the score.

These Hypothesis will then be evaluated wit the associated **p-value**, which quantifies the strength of evidence against the null hypothesis, which posits no correlation. A smaller p-value indicates stronger evidence against the null hypothesis. Conventionally, if the p-value falls below a chosen significance level (in our case we chose a level of 0.05), we may reject the null hypothesis, suggesting a statistically significant monotonic relationship between the variables.

After running the test on the data we collected, these were the results:

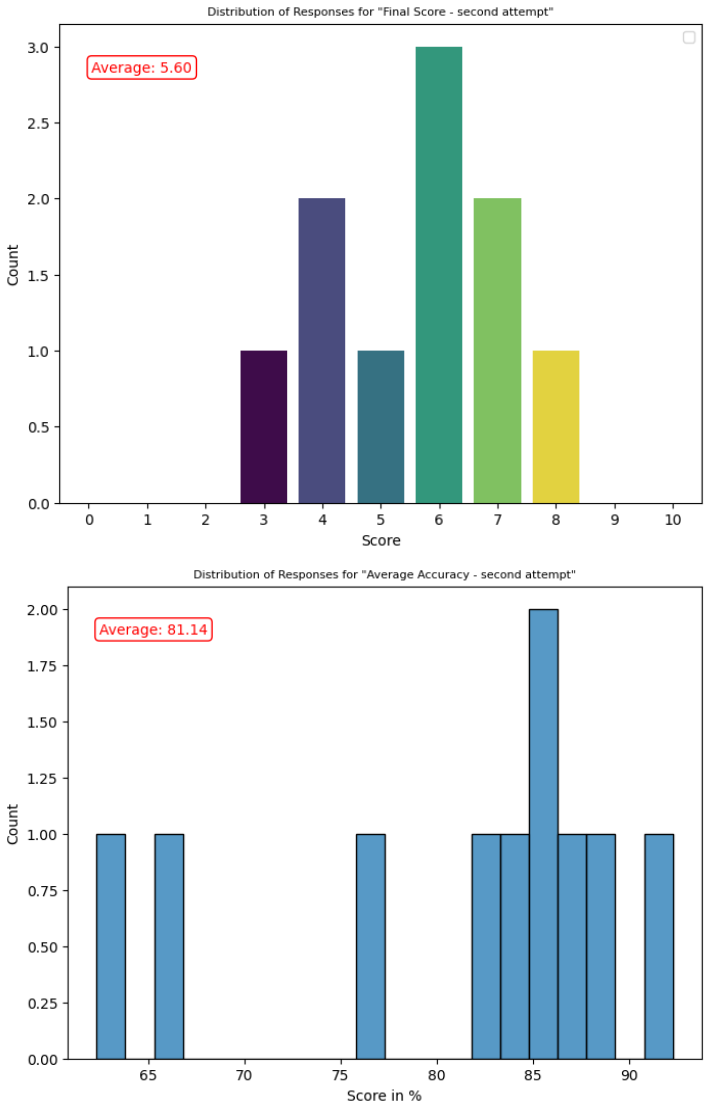
- $\rho = -0.7803066928537232$
- P-value $= 4.9403 \times 10^{-5}$

A ρ of approximately -0.7803 indicates a robust and statistically significant negative correlation. This implies that, as players perceive the game's difficulty to increase, there is a tendency for their final scores to decrease. The p-value of 4.9403×10^{-5} is less than the chosen significance level of 0.05. Therefore, we have sufficient evidence to reject the null hypothesis, supporting the statistical significance of this correlation.

An alternative approach involves instead the analysis and comparison of a player's performance across two distinct attempts. This examination aims to determine whether our game allows for improvement over time, considering the reasonable level of difficulty that emerged from the players' opinion. A positive correlation between the player's experience and their current performance suggests that experience gained from the first attempt positively influences performance in the following attempts, indicating skill development.

This is a fundamental aspect: if players consistently see improvements in their performance across attempts, it can enhance their sense of achievement, engagement, and satisfaction with the game. Understanding this correlation then, is crucial for optimizing the gaming experience.

We then subsequently collected some additional data, asking 10 of the players that tried the game during the testing phase, to take another attempt. The results are the following:



As we can observe, the improvement in the performance of the player is particularly evident. The average Final Score improved by 1.3 points, while the Average Accuracy improved by almost 10%.

In addition however, we decided to run a more detailed test, the **Paired T-test**, a statistical method which is well-suited for analyzing paired observations, such as in our case, scores from repeated measurements on the same individuals. It compares the mean difference between paired observations to a null hypothesis that assumes no significant difference. The **t-statistic**, calculated from the sample data, is then used to compute a **p-value**. If the p-value is below a predetermined significance level (in our case we chose a value of 0.05), the null

hypothesis is rejected, suggesting that there is sufficient evidence to conclude that a significant difference exists between the paired groups.
In this case, the hypotheses are the following:

- **Null Hypothesis (H_0):** There is no significant difference between the scores of attempt 1 and attempt 2 for the same player.
- **Alternative Hypothesis (H_1):** There is a significant difference between the scores of attempt 1 and attempt 2 for the same player.

After running the test on the data we collected, these were the results:

- T-statistic $= -4.33$
- P-value $= 1.90 \times 10^{-3}$

A negative t-statistic of -4.33 suggests that, on average, the scores in the first attempt were significantly lower than the scores in the second attempt.

Furthermore, the small p-value of 1.90×10^{-3} indicates strong evidence against the null hypothesis. Therefore, we can reject the null hypothesis and conclude that there is a statistically significant difference in scores between the two attempts.

Chapter 5

Conclusions

In this project, we successfully conducted a study on the interaction between a user and a robot. In our case, the robot is a hardware device equipped with a webcam, a browser and an in internet connection. The system with which the user interacts is a game designed to entertain the player.

Initially, we analyzed the application's requirements, considering both its internal structure and its interaction with the user. Subsequently, we explored several research questions that defined the study's focus and provided guidance. After selecting the appropriate technologies, we leveraged our expertise in software design to design and structure the application.

Once we developed a playable version of Pose Royale, we initiated a study to test and evaluate the application, adhering to the guidelines outlined by the research questions. Therefore, our efforts not only resulted in the creation of software but also in the acquisition and analysis of information regarding the accuracy and effectiveness of the interaction between the user and the robot.

In conclusion, we take pride in the work accomplished with Pose Royale, considering it an enriching experience that significantly expanded our knowledge in this field of study.

Chapter 6

Annexes

1. Project Code: https://github.com/giacomoschiavo/pose-royale
2. Game Link: https://poseroyale.netlify.app/play

Bibliography

[1] ARI. https://pal-robotics.com/robots/ari/.

[2] Figma. https://www.figma.com/.

[3] GIMP. https://www.gimp.org/.

[4] GitHub. https://github.com/.

[5] GoogleMediaPipe. https://developers.google.com/mediapipe.

[6] Hole in the Wall Game. https://en.wikipedia.org/wiki/Hole_in_the_Wall_(American_game_show).

[7] JSON. https://www.json.org/json-en.html.

[8] Netlify. https://www.netlify.com/.

[9] React. https://react.dev/.