הנדסת תוכנה שנקר
מהנדסים טאלנטים בתוכנה

SYNC ALONG

Software Engineering B.Sc. Final Project Write-up

Author: Shani Levi & Ayman Wahbani
Supervisor: Dr. Marcelo Sihman & Mr. Eyal Nussbaum

25/08/2022

Approved by: Eyal Nussbaum **Signature:** _____

Approved by: Marcelo Sihman **Signature:** Marcelo Sihman

A thesis submitted in partial fulfilment of the requirements for the degree of BSc. in Software Engineering

Abstract

Synchronization is widely observed in human communication and is considered to be important in generating empathy during face-to-face communication. Synchronization of non-verbal behaviors has psychologically positive effects.

Although the benefits of increased physical activity and exercise are universally recognized, many older persons remain sedentary, and relatively few achieve recommended levels of activity. Effective interventions to reverse the lack of physical activity in older adults are clearly needed.

The SyncAlong system is designed to allow elderlies who are unable to participate in physical activity the way to perform activities in a positive way through synchronized and positive activities by measurement of synchronization movement in an online joint activity of two participants in a remote way.

Acknowledgements

We would like to thank our supervisors, Mr. Eyal Nussbaum and Dr. Marcelo Sihman, who guided us through the project and accompanied us along it, for the exceptional guidance, advice and thought directions that are invaluable and helped us reach an effective and accurate solution to the needs of the project that contributed greatly to the successful completion of the project.

We would also like to thank the Kedar Center, Michal Rinott and Shachar Geiger, who helped us by giving us the advice and focus we needed, the open door and the encouragement and support.

Finally, but not least, we would like to thank our UI consultant, Ronen Kertis, who helped and motivated us and gave us a broad perspective on UI and gave points in this project that improved the final product.

Table of Contents

Table of Figures

1 INTRODUCTION

Synchronization is widely observed in human communication and is considered to be important in generating empathy during face-to-face communication. The benefits of increased physical activity and exercise are universally recognized, although physical activity is a noninvasive and cost-effective method of improving the quality of health, global statistics show that only a few middle-aged and older adults engage in the recommended physical activity. This is **due to a lack of motivation and companionship**.

Clarifying the relationship between changes in physical indicators and degree of empathy would improve interpretation of the cognitive relationship between body motion synchronization and degree of empathy from a physical aspect. Because body motions give different impressions, depending on speed and generation timing (Description given by Mehrabian and Williams, 1969; Miller et al, 1976).

SyncAlong provides a simple and convenient interface for the adult user for synchronized exercise in shared sessions performed simultaneously via peer-to-peer video connection. Key components of synchronized motion are the similarities between users in the context of joint motion over time in video chat frames. The system enables connection of elderlies remotely for the purpose of joint synchronized physical activity with friends / family.

1.1 Problem Description

Although the benefits of increased physical activity and exercise are universally recognized, many older persons remain sedentary, and relatively few achieve recommended levels of activity. Effective interventions to reverse the lack of physical activity in older adults are clearly needed. Between synchronized activity and mental state together with the psychological benefits of synchronized activity there is a real need for a synchronization measurement system because studies show that when the degree of synchronization is high then the degree of interpersonal empathy is high, investigating body movement, phase differences, frequency of synchronization, and body movements in relation to the degree of empathy

1.2 SyncAlong Motivation

Exercise and physical activity are thought to be among the most important lifestyle factors for the maintenance of health and prevention of premature disease and mortality. Yet sedentary lifestyles are common. Many people avoid exercise, and have done so across their lifespan. Exercise and physical activity are also considered important for positive psychological functioning.

Key facts about Physical activity:

- Physical activity has significant health benefits for hearts, bodies and minds
- Physical activity reduces symptoms of depression and anxiety
- Physical activity enhances thinking, learning, and judgement skills
- Physical activity improves overall well-being

Synchronization in joint physical activity has psychologically positive effects such as:

- Creating a positive feeling
- Creating a sense of belonging
- Reducing feelings of loneliness
- Increasing motivation in joint activities
- Increasing perseverance for a common goal (physical activity)

1.3 SyncAlong Goals

The goal of the project is to develop

- A platform that enables synchronized activity with family and friends remotely
- A system for older adults for increase motivating to do more sport activities
- A multi-participant training system designed to support interpersonal training in a synchronized manner between the participants.
- A platform for communication and sports activities with adults who cannot physically attend joint meetings.
- Positive response to participants when syncing in motion for the purpose of raising motivation.
- View data on synchronized activities.
- Calculating and identifying synchronized movement and providing positive feedback to the user in real time.
- Raising motivation and improving the mental and health condition of adults.

1.4 Overview of the SyncAlong Approach

The SyncAlong system evaluates synchronized activities performed simultaneously via video. Our approach uses pose estimation detection model and image analysis and will be displayed using a web system with peer-to-peer secure video.

For the video connection for open an activity session we use

1. WebRTC that leverages RTCPeerConnection to send stream data between browsers also known as peers.
2. Socket.io to be able to send control messages in rooms.

This architecture is known as JavaScript Session Establishment Protocol. Once this signaling process is successful, data can be streamed directly peer-to-peer between the caller and callee, in other words, between the users. (View in Figure 1)

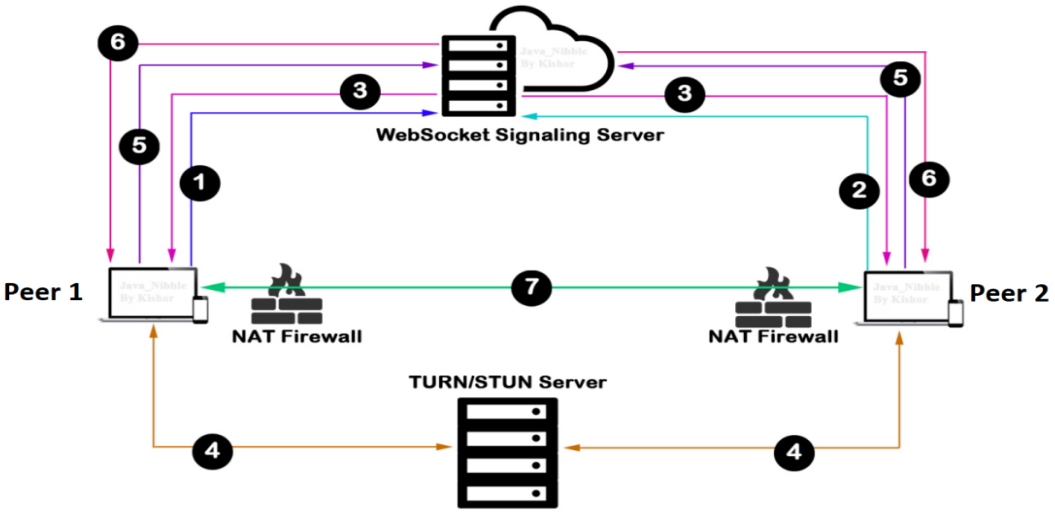


Figure 1-1: Communication on WebRTC ,Handshake req-res

The session is defined with a list of physical activities tailored to the limitations and abilities of the participants. The system identifies users estimated in real-time using **pose estimation model (computer vision)** and identifies synchronized movement of users during a joint physical activity. The system also uses **voice recognition** in order to support basic actions during the physical activities, e.g. stopping the activity using the word `stop`.

Synchronization Approach

Synchronization can be described from two perspectives: at the level of system dynamics and at the level of influence among system users. At the system level, synchronization refers to the coordination in time among the states or dynamics of the elements comprising the system. The system considers the differences in poses, and the **consideration of the time series** of users' movements, which is a critical component that creates visual aesthetics seen in synchronized movement.

1.5 Usage scenarios

SyncAlong provides support for synchronized joint exercise in the adult user and trainer user.

1.5.1 Scenario 1 - Trainer

Jasmine, 26, lives far away, and her beloved grandmother, Roberta. Jasmine is interested in performing a joint sporting activity with Roberta for the purpose of a joint positive activity and for the purpose of maintaining a healthy lifespan for Roberta. Jasmine opens new users in the SyncAlong system for herself and her grandmother and defines their data for the purpose of adapting the system to sports activity offers tailored for them.

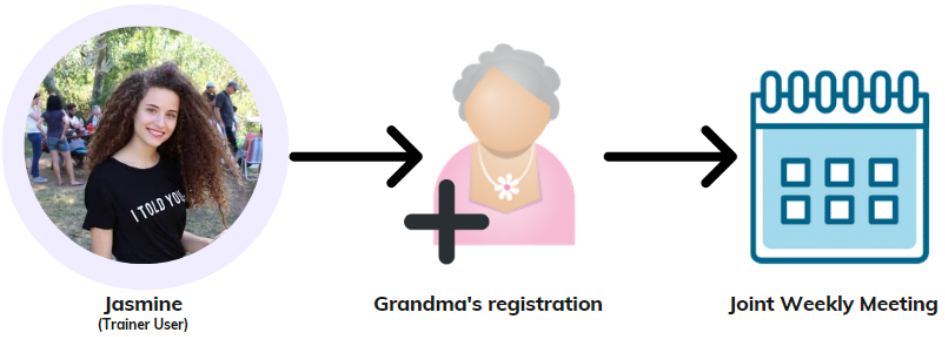


Figure 1-2: Scenario 1 -Trainer

1.5.2 Scenario 2 - Connection of an adult user to a joint meeting

Roberta is 80 years old, lives alone and suffers from loneliness and unwillingness to perform sports activities that are necessary to strengthen her joints, Roberta uses the system for a regular meeting with her granddaughter, Jasmine, she opens a meeting and with a click connects to the regular meeting with Jasmine, the meeting begins with a training list tailored to both Roberta's and Jasmine's data and abilities. Jasmine and Roberta are shown side by side. The system displays a simulation of the physical activity of the session, at the end of the simulation - the session begins and the session time appears on the screen. During the meeting, Jasmine and Roberta perform a synchronized hand action and a variable animation rotates between them, When out of sync and the animation disappears.

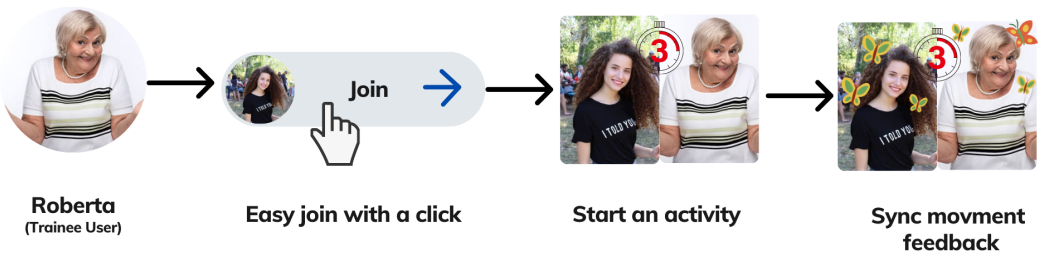


Figure 1-3: Scenario 2 - Connection of an adult user to a joint meeting and its operation

1.5.3 Scenario 3 - Summary and follow-up synchronization of a trainer use

At the end of a physical activity session with Roberta, Jasmine checks the results of the session. The system displays for Jasmine the list of activities she has performed in order of date and time. When entering the meeting control the system displays the shared video of the participants and allows Jasmine to go through the points in the video where unsync, low-sync and synchronization were detected. The video is displayed with the participants' posture assessment marked and presents indices of improvement or non-improvement of synchronization compared to a previous encounter that exists in Jasmine's system SyncAlong audience

1.6 Glossary

The following terms are divided into two categories: generic and SyncAlong specific terms.

1.6.1 Generic terminology

- **Real-time system** : Send and receive data instantly online across multiple clients. Two-way flow.
- **Signalling** : WebRTC allows real-time, peer-to-peer, media exchange between two devices. A connection is established through a discovery and negotiation process called signaling.
- **Synchronization of movement**: Synchronization of movement is defined as similar movements between two or more people who are temporally aligned
- **Section**: A list of varying sizes (few parts) of workouts whose total predefined time
- **Feedback**: Visual view of the quality of synchronization between users
- **Period time:** A period of time that elapsed from the moment an body part is in a certain place until the moment it returns to the position

1.6.2 SyncAlong specific terminology

- **Synchronization** : value in the range of 0 to 100.
  - **Sync** : value equals or above 75.
  - **No sync**: value ander 75.
- **AR Feedback**: Augmented reality (AR) is an enhanced version of the real physical world that is achieved through the use of digital visual elements, sound, or other sensory stimuli delivered via technology
- **MediaPipe** : Google Ai model , Human pose estimation from video\live stream.

2 LITERATURE SURVEY

This chapter includes a mixture of relevant academic material as well as a number of examples of related existing applications.

2.1 Problem Survey

The problem that SyncAlong is trying to address is the lack of motivation and lack of physical activity among the elderly, physical activity is of high importance in the physical and mental state.The need of a system that will encourage joint sports meetings and will know how to recognize a synchronized movement and provide feedback to participants in real time about their movement .

Synchronized joint activities have two key elements of synchronized movement: the similarities between the pose estimation of humans and the time series of movements. Research of OpenCv library-based interpersonal synchronization systems offers many methods for identifying human posture assessment in different techniques and obtaining different data reviewed in this section. Studies show that the way information is collected for the purpose of calculating synchronization between people can be examined in various indices.

2.2 Solution Survey

In this section we discuss academic and commercial surveys of existing similar or different solutions to the same problem described in the last section.

2.2.1 Encourage joint sports meetings

Two Riders IMS project

The project was motivated by the creators' interest in design for interpersonal synchronization, and by a commission for an exhibition about the history, science, and design of bicycles. The creators conceived a novel experience using a physical object designed to enable, motivate, and reflect interpersonal synchronization. The project was installed for a total of two years in three science museums in three countries: Israel, Italy and Germany. In the Two Riders installation, two people sit face-to-face on a pair of bicycles that have a shared front wheel. Each participant can pedal independently. Their pedaling generates music that is dependent on their synchronization: the better they synchronize their cycling, the richer the music becomes.

Figure 2-1: The Two Riders installation with participants.

"Flying with the Snark"[1]section:6.2

The Snark adventure game is a mixed reality game, where children hunt an elusive, virtual creature called the Snark, in a large interactive environment. It was designed by a team of researchers and designers headed by HCI researcher Yvonne Rogers. In this experience pairs of children were given jackets embedded with sensors to wear, and interacted with an abstract representation of the Snark, presented on a screen. The relationship between the children's actions and the Snark's "behavior" is described as such: if both jackets are banking the same way, then the Snark comes close and laughs. If both are flapping together then the Snark comes close and is very happy, soaring and swinging and gliding

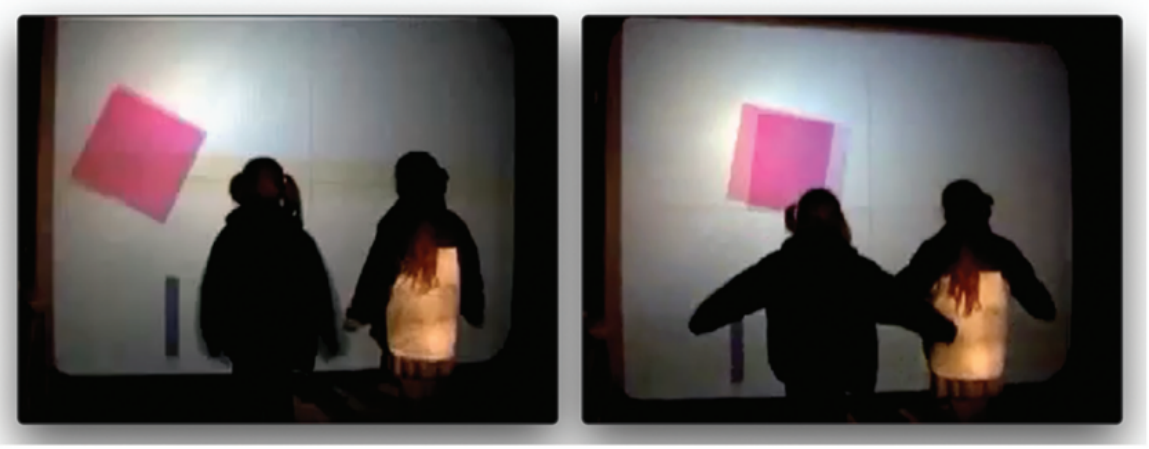


Figure 2-2: Frames from the movie documenting "Flying with the Snark". Reproduced with permission.

2.2.2 Synchronized joint activities

Research on Pose estimation, tracking and comparison[1]section:IV

In terms of the similarity of human poses: The article addresses given images, and images can be different sizes, a person can appear in a different part of the image, and by manipulating the image - by changing the size and scale of a bounding box of the person's image for consistent size. Additional steps involving normalization of the coordinate result as normalized L2 Vector array, And comparison by calculating the similarity of cosines

- The problem with this approach to the solution is the approach of image manipulation, the SyncAlong system produces a connection via stream and does not produce continuous images, continuous images can be produced but processing time will be required for this step In terms of aligning sequence of pose: As the poses can be viewed as multi-variate time series, a method called Dynamic Time Warping (DTW)

Match Pose - A System for Comparing Poses[2]: instead of mapping the skeletons length, the need is to compare the angles between their joints

3 TECHNOLOGICAL SURVEY

a. Problem Survey

Real-time peer-to-peer signaling and media exchange:

Both WebRTC and WebSocket are technologies for communication capabilities.

Client side (Different platforms for Web applications):

- The use of our system instructed by Kedar Center for sync-data tracking
- Easily and accessible by using the Web app.
- Switching between protocols, http←→webSocket.
- Supports libraries for pose estimation and voice recognition.
- Supports AR.

Algorithms requirements:

Motion synchronization is defined by two key elements:

- Match in human evaluation pose : a major requirement of our system is the ability to make an adjustment An engine that will adjust different posture values between users (participants in an activity).
- Match of speed/rate of movement: it is necessary to know the speed of movement of each participant, For the purpose of a synchronization index in the joint movement.

Server side:

- Permanent availability is required for end users based on Web RESTful services.
- Receiving requests from several customers at the same time.
- Switching between protocols, http←→webSocket.
- Create and update information that is in the database.
- Large videos are required to be saved

Data base:

- Entities with a fixed structure with predetermined parameters.
- Queries which consist of cutting and joining several different entities in the system.

b. Solution Survey

Real-time peer-to-peer signaling and media exchange : source-link sec-link

	WebRTC	WebSocket
Pros	• This reduces latency and also permits things like transcoding, recording, and other server-side integrations such as SIP which would be much more difficult in a peer-to-peer connection. • WebRTC is more secure • No software installation required • High-quality data transmission using modern audio and video • Video or audio quality adjusts to the digital environment during the connection • Reliability and safety of data - the server protects connections by encrypting them via TLS and SRTP protocols	• WebSockets is compatible with almost all existing browsers • The need for a video conferencing server (which mixes video and sound) for audio and video conferences in a group, since the browser is unable to synchronize two or more incoming streams • Cross-communication (cross origin communication) - it creates certain security risks but is needed for extensive functionality; • Cross-platform compatibility (web, computer, mobile devices); continuous connection from the backend to frontend in a web application working with a server

Cons	• The need for a video conferencing server (which mixes video and sound) for audio and video conferences in a group, since the browser is unable to synchronize two or more incoming streams	• **Slow** interaction response. The result of sending data to the WebSocket becomes known after 75 seconds of the timeout. • Security issues in WebSockets: processing denial; possible private data leaks; client-server encryption, etc.

Client side : source-link

	Angular	React
Pros	• Provides a complete application development application in the MVC architecture. • Allows testing to be performed using one tool. • Allows for a uniform version update for the entire infrastructure	• UI software development infrastructure that uses additional libraries as needed for development Client side, which allows flexibility both in terms of UI and in terms of functionality. • Easy to understand and use libraries that allow faster development. • Uses virtual DOM trees that enable enhanced performance and Improved user experience.
Cons	• Large library that requires deep understanding and a lot of learning time for new developers. • Includes many features embedded in the heart of the infrastructure. On the one hand it is an advantage because it does not require the use of additional directories but on the other hand limits the development. • Uses real DOM trees which are updated from beginning to end in each Update which degrades the performance of the system.	• Requires several tools to perform tests. • Requires the use of external libraries for consuming various functions. • Updating the version is not a simple procedure because it is necessary to verify versions of Third-party libraries used in development

Algorithms :A Libraries for Machine Learning in JavaScript, For Pose,Hand and Head Estimation :

*More ML libraries	TensorFlow.js	MidiaPipe / MP
Pros	• It has better computational graph visualizations. • It has the advantage of seamless performance, quick updates and frequent new releases with new features. • It can be deployed on a gamut of hardware machines, starting from cellular devices to computers with complex setups.	• MediaPipe offers cross-platform, customizable ML solutions for live and streaming media. • Working on a Graph-based framework, MP is tremendously faster than TensorFlow, especially so on the browser. • The visualizer runs these models in browser as wasm, WASM is a compilation product that

	• Tensorflow is highly parallel and designed to use various backends software (GPU, ASIC), etc	technology allows to run heavy software at a speed that is almost as fast as they run on a computer.
Cons	• No support for Windows • Missing Symbolic Loops	• It has limited support for hardware acceleration. • It does not have default access to the file system in the browser host environment
Runtime (FPS)	**Desktop:** 42 | 35 | 29 **MacBook Pro:**52 | 40 | 24 **iPhone :** 43 | 32 | 22 **Slower**	**Desktop:** 150 | 130 | 97 **MacBook Pro:**75 | 67 | 34 **iPhone :** 9 | 6 | N/A **Faster**

Server side: source-link

	Node.js	PHP
Pros	• Faster development, requiring fewer lines of code. • Same language to server and client side helps synchronize between them. • There are external modules that can be used according to the development needs. • Known as a fast engine in the market compared to the other options. • Open source with broad community support.	• Open source programming language is common, especially for server-side web application programming. • Can be run on a wide range of operating systems and servers. • Enables efficient work with tabular databases using the ORM system. • Enables development cost savings if development is done by non-software infrastructures cost.
Cons	• Creates a bottleneck with heavy processes. • Known as problematic in performing asynchronous operations.	• Is known to have multiple security vulnerabilities. • Some of the software infrastructures for working with this language have little support.

Data base: source-link sec-link

	SQL	NoSQL
Pros	• The information is stored in a readable and simple way in the tables. • Cuts and combinations of tables can be made. • Avoidance of duplications. • Information Integrity (ACID)	• Very flexible and unlimited data structure. • Has enormous storage capabilities • NoSQL offers speed at the expense of information integrity (ACID).
Cons	• Limit a size in a particular field so that data may be lost. • Has a limited structure.	• Less readable and easy to track. • Limited ability in comparisons, analyzes and combinations

4 REQUIREMENTS AND SPECIFICATION

Basic requirements for behavior of the SyncAlong system.

4.1 Functional requirements

The following requirements relate to the basic functionality of SyncAlong system:

Trainer User Functionals :

1. Registered trainer type users can create new users as trainees users, and receive viewing and editing permissions of these users (CRUD for trainee profile)

2. A trainer user can schedule a joint sports meeting with an adult user who is under his permission,responsible for setting the time of the meeting, the list of activities. (including CRUD for trainee meetings)

3. A trainer user can make a request for a quick meeting with users who are online.

4. Tracking the amount of sessions performed and the synchronization of the adults users is displayed to the trainer user on the landing page in a convenient and accessible way.

Overall System Functionals :

1. Users can register in the system as a trainer/instructor, regular user or a senior user.

2. When requesting to open a room, the system defines a room for the users according to the participants limitation data. The system allows the user to change the room settings such as: room type/name, participant invitation, activities listed.

3. A training set list tailored to the meeting of the participants based on users profiles data.

4. The physical training displays by presenting a short simulation before each start of a new training in the defined training set.

5. A list of predefined types of training according to the type of training and the type of treatment in the body part is managed automatically.

6. Participants are displayed side by side in a web system during a joined physical activity.

7. Pose estimation of the users in a joint meeting is detected on real time via steam using a pc camera.

8. During a meeting with participants and a list of training set attached to the meeting - the system will allow participants by identifying predefined voice recognition to mark the system to:

    - Move to the next workout on the list.
    - Stop to rest.
    - End the session.

9. During an active meeting the level of synchronization between the participants is checked for each frame and provides a visual feedback to the participants' screen if there is synchronization

10. Meeting data are automatically saved at the end of the list of exercises in a joint session ,and allow meeting observation and receiving feedback and improving synchronization

11. Viewing of the session recording and the sync levels results is available for each completed meeting.

12. A joint meeting that started and did not finish the list of exercises and the recording of the meeting was saved –is defined as an active meeting, when the participants are identified as connected to the website the system will alert them if they wish to connect or end the meeting.

13. Production environment: the application will work on PC via web interface

4.2 Non-Functional requirements

The following is a list of non-functional requirements:

1. Privacy: User videos are secure and not exposed to unauthorized users
2. Extensive data storage support
3. The system saves videos on the server-side using aws storage
4. Performance: Maximum accuracy of time synchronization of various devices and speed of movement while streaming video.

5. Resilience: Handling communication disconnection during a meeting and providing to the user who left due to communication disconnection to return to the meeting

6. Data retention: Meeting data retention will be performed for all phases of the session along divided frame segments

4.3 Specification - the scope of the SyncAlong project

The scope of the SyncAlong system is about synchronized joint physical activity. Addresses a few key points:

- Real-time Communication
- Client-server computing or networking
- 2D human pose estimation involves estimating the articulated 2D joint locations of a human body.
- Real-time synchronization algorithm & AR display in real time

4.4 The SyncAlong approach

The SyncAlong system evaluates synchronized activities performed simultaneously via video. Synchronization refers to the matching between **time** and the users' **poses** which together describe the pose similarity estimation.

Poses are defined as the human's landmarks identified in real time through pose estimation and the pose similarity between two persons. **time** is defined as time in milliseconds during landmarks data collection.

The pose similarity estimation is the first step and a critical component in the SyncAlong system. The second step in the sync calculation process is based on the calculation of distances of the normalized users' poses (which is the result of pose similarity step), This step relies on the summation of the **Euclidean distances of body parts and the similarity of the distance between two persons**.

Humans can have different heights and body shapes, they might be in different parts in the picture: one person may have been standing close to the camera, another might have been faraway. All these problems must be solved in order to output a correct result.

The estimation of the human pose from the video is collected in real-time using a MediaPipe Pose-model which is an ML solution for high-fidelity body pose tracking, The landmark model in MediaPipe Pose predicts the location of 33 pose landmarks.

In the SyncAlong system and for its purpose (main landmarks of users' pose are taken into account for the purpose of reducing the size of the passing data by focusing on main landmarks. e.g. right and left hands landmarks, right and left legs landmarks.) we forced on 12 pose landmarks (see figure below) :

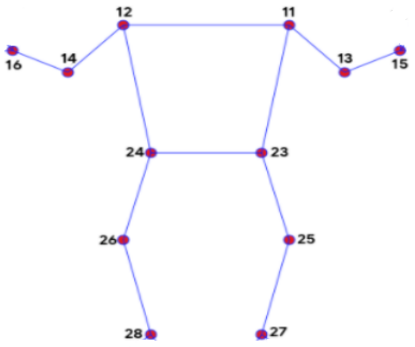
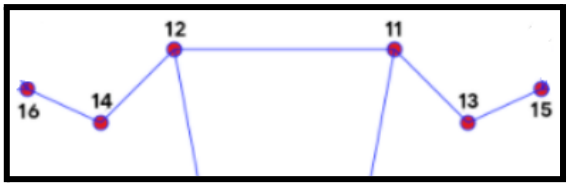


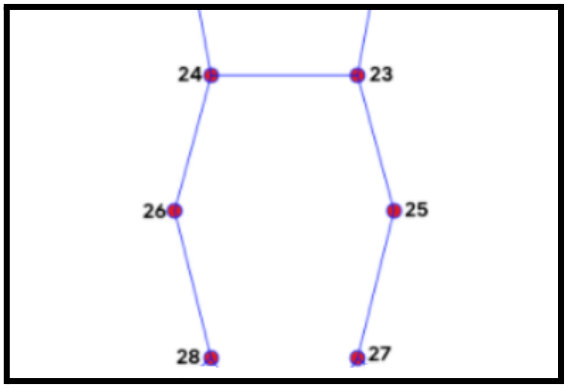
Figure 4-1: pose estimation MediaPipe Model(Google Ai)

MediaPipe Pose Landmarks are divided into central regions for accuracy by subdividing into small body regions. For the purpose of calculating the synchronization, we decided to perform a synchronization test for each part separately.

Upper body part and lower part, each side (right and left) are examined separately and the average inclusion of both is the area of synchronization by region.



Upper part



Bottom part

Figure 4-2: Division into regions of pose estimation

data receiving from pose for each frame:

```
{

  image: ImageBitmap,

  leftHandLandmarks: Array(21),

  faceLandmarks: Array(478),

  poseLandmarks: Array(33),

  ea: Array(33), …

}

poseLandmarks{

  0: {x: 0.5797398090362549, y: 0.7509459257125854, z:-2.0631303787231445, visibility:
0.9994146227836609}

  1: {x: 0.6200095415115356, y: 0.6761197447776794, z: -2.052156925201416, visibility:
0.999018669128418}

        … (and more points)...

 32: {x: 0.5109288692474365, y: 0.6718034744262695, z:-2.0453882217407227, visibility:
0.9992113709449768}

  length: 33 }
```

Figure 4-3: Output of pose estimation MediaPipe Model(Google Ai)

In order to shorten the running time on all points, the points will be filtered for the type of action performed (hand action will be considered an action on the upper torso) so that only points of the same body area will be sent to the sync algorithm.

Our Approach for Human Pose Comparison is by using Procrustes Problem,

Procrustes analysis is a form of statistical shape analysis used to analyze the distribution of a set of shapes. To compare shapes of two objects, the objects must be first optimally "superimposed". Procrustes analysis is performed by optimally **translating**, **rotating** and **uniformly scaling** the objects. In other words, both the placement in space and the size of the objects are freely adjusted.

The aim is to obtain a similar placement and size, by minimizing a measure of shape difference called the Procrustes distance between the objects.

Procrustes analysis is the comparison of two sets of configurations (shapes).

*The **Generalized Procrustes analysis** algorithm outline is the following:*

1. *arbitrarily choose a reference shape (typically by selecting it among the available instances)*
2. *superimpose all instances to current reference shape*
3. *compute the mean shape of the current set of superimposed shapes*
4. *if the Procrustes distance between the mean shape and the reference is above a certain threshold, set the reference to mean shape and continue to step 2.*

Figure 4-4: Generalized orthogonal Procrustes Algorithm

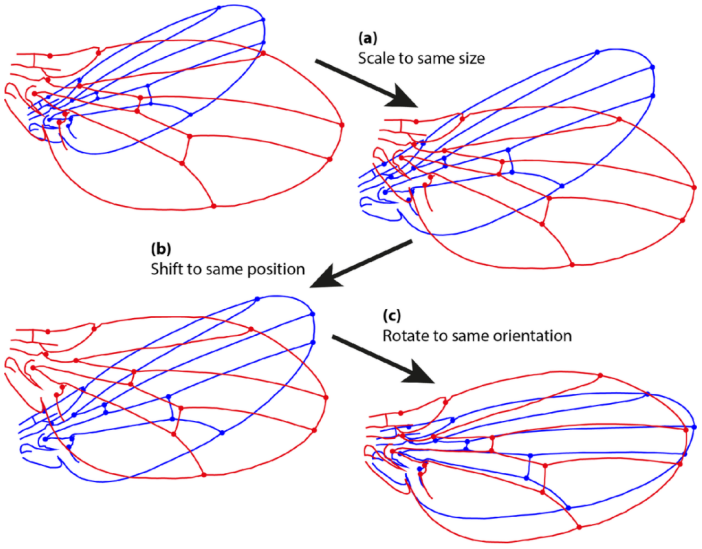


Figure 4-5: Generalized orthogonal Procrustes steps

Given two curves in a metric space, The Fréchet distance is a measure of similarity between curves that takes into account the location and ordering of the points along the curves (as detailed at the beginning of the above section 4.4). Therefore it is often better than the Procrustes distance solution.

An advantage of the coupling measure is its efficient computability by dynamic programming, without the need of complicated data structures. The algorithm **dF** in Figure 4-6 coded.

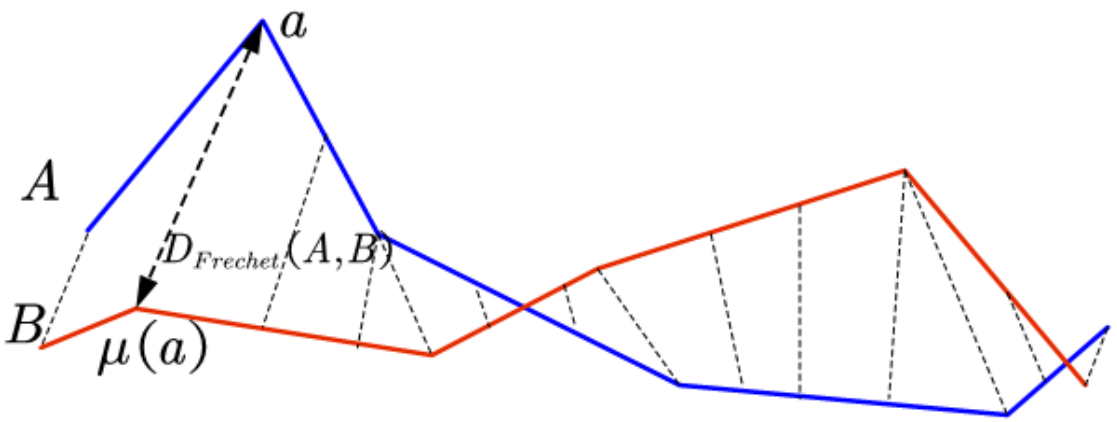


Figure 4-6: Fréchet distance measurement.

5 THE ARCHITECTURE

In order to describe the architecture of the SyncAlong system we begin with a description of the communication channels that enable information transfers in the system, their meaning and use in the system. Next, an explanation of the process of transmitting information in the media in a joint sport meeting of two users. Finally, the activities that occur when using the system are explained, resulting in a description of the life cycle of the system - the stages in which the system goes through when using it.

5.1 SyncAlong Data

5.1.1 The Representations Of Information Transfer And Data Storage

SyncAlong's basic data unit consists of 3 communication channels: HTTPS , Socket.IO, WebRTC and using data storage using AWS

- **Information passes through the HTTPS protocol** – information of the users that passes and has no importance in simultaneous communication between users, information about the data of the connected user, including full CURD.
- **Socket.IO Information Transmission ("Real-Time Communication")** – Socket.IO is a library that enables low-latency, bidirectional and event-based communication between a client and a server.

  Socket.IO library enables real-time communication between users in the system, such as: requests for initial connection of the meeting, request for initial connection, updating preliminary data for the purpose of connection before making the connection via the webRTC
- **WebRTC ("Real-Time Communication")** – facilitates real time data communication between two peers. To connect two users over WebRTC, we exchange information to allow browsers to talk to each other. This process is called signaling and it is facilitated by using NodeJS and a socket server. Other than signaling, no data has to be sent through a server. When a connection is successfully established and authentication and authorization are complete, stream data exchanged between peers is directed to a React component for rendering.
- **AWS Storage Data** – While MongoDB can be utilized for file storage, in the system, file storage is left to aws storage systems. Amazon Simple Storage Service (Amazon S3) is an object storage service offering industry-leading scalability, data availability, security, and performance. With cost-effective storage classes and easy-to-use management features.

5.1.2 Software Architecture Pattern

The architectural pattern captures the design structures of various systems and elements of software so that they can be reused.

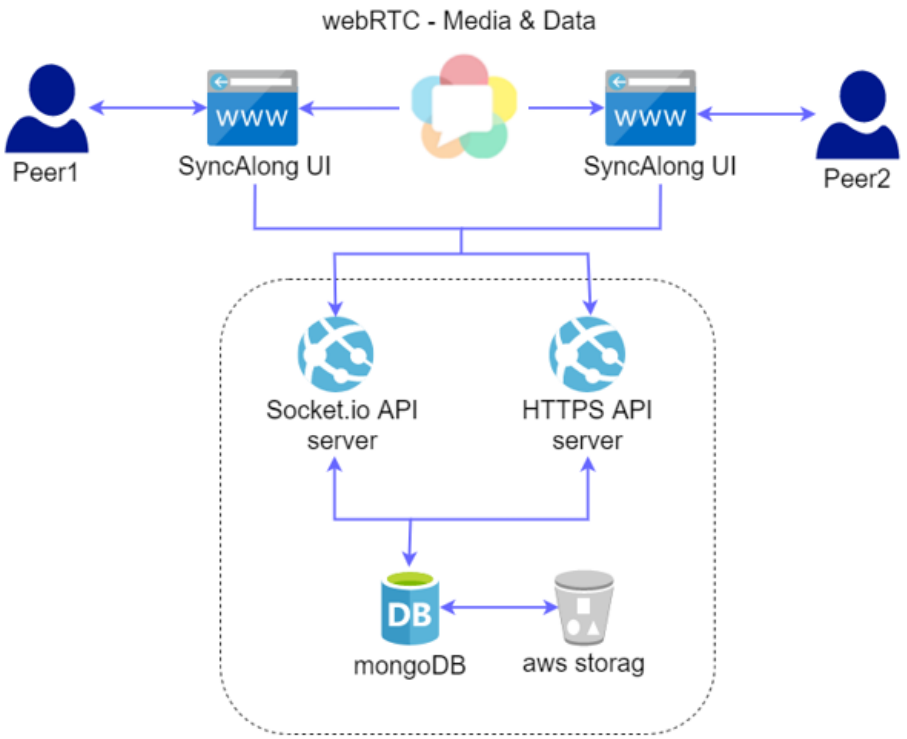


Figure 5-1: Software Architecture Pattern

5.1.2.1 *Main process description* - ***Connection creation process, activity process, synchronization process.***

Two users, an adult user (Trainee) and a trainer user, can connect to a pre-arranged joint sporting session or for a quick session created by the trainer user (Figure5-2). When connecting users to the sharing session, participants connect to a shared video chat room (Figure 5-2 conection_section).

At the end of the communication connection, the activity begins with voice recognition ("start"), and by going over a list of sports exercises defined for the current session. There is a continuous guarantee of data from both users of the server to consume a synchronization calculation according to the transfer of data collection of human postures (Figure 4-4 activity_data_section).

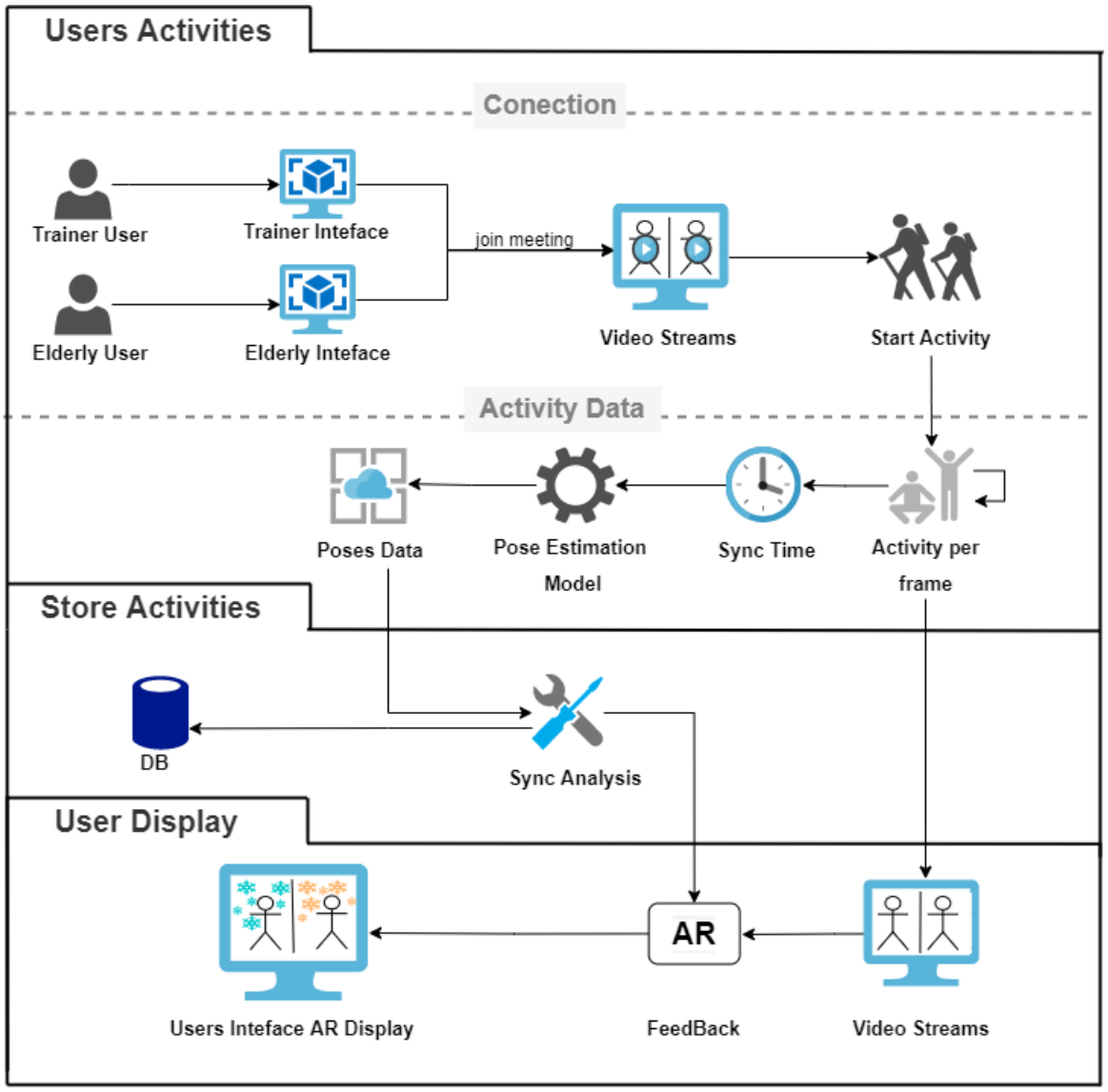


Figure 5-2: The process of transferring information in connection with users to a joint and synchronized sports meeting

5.2 SyncAlong Processes

We will describe the process of transferring information in connection with users to a shared and synchronized sports meeting which is shown in the figure 4-4 above.

**The connection will be made between the users after each user has successfully completed the initialization of the human posture identification library (MediaPipe library) and the identification of the person is done in real time.*

1. **User Activity layer**:
   a. **Connection** - Two users, an adult trainee and a coach user, can connect to a pre-arranged joint sporting session or to a quick session created by the coaching user. When connecting participants connect to a shared video room and their Stream transitions to each other via a connection in a browser supported via webRTC.

      * *Connection made successfully, activity not yet started*

   b. **Activity Data** - From the moment the Peers connect, they transmit information of their human positions sequentially to each frame. The activity begins with voice recognition ("start") and is performed over a list of sports exercises defined for the current session.During the sporting exercise the two participants simultaneously and simultaneously (the closest time between them) transfer their human poses obtained from the MediaPipe library to the server.

2. **Store Activities layer**:
   - **Sync Algoritem**: The server listens to the peers' information and when it detects that two peers are in the same room and the information received from them is at the same time or time approximately the server transmits the information to the synchronization algorithm. The algorithm receives information that includes an array of landmarks of each peer1 and peer2, the algorithm returns a number between 0 and 1 as a sync result. The result is saved in DB including the result value and the result time.
3. **User Display layer**:
   - **AR Display** (output): client side receives sync result each frame during 30s of activity, according to the sync results (syncing is above 0.75) an animation adapted to a predefined type of activity will appear on the screen.

     The display is based on the positions of the human poses identified in real time by the MediaPipe library.

5.3 The life-cycle view of a system

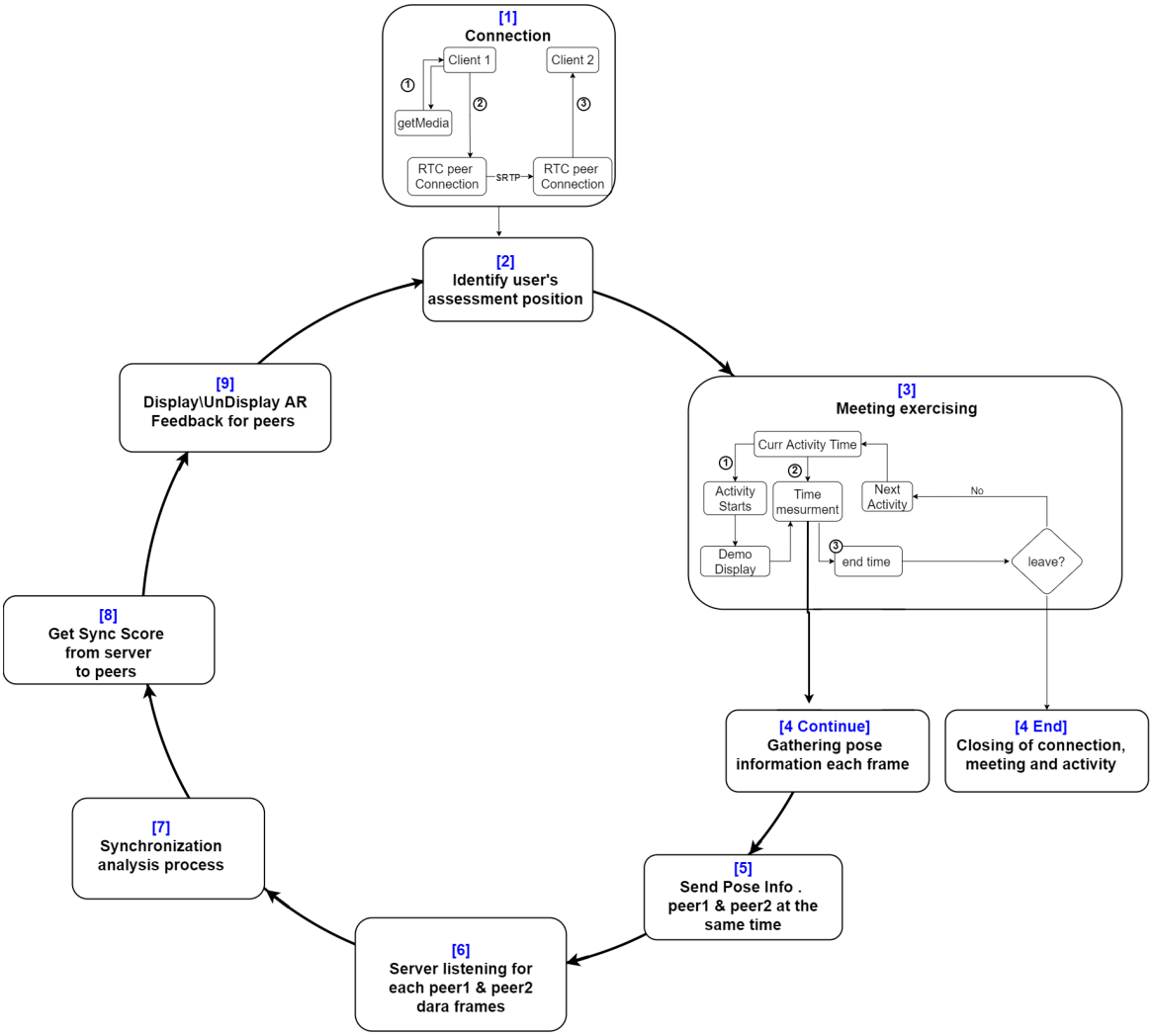


Figure 5-3: The life-cycle view of a system

6 SYSTEM DESIGN

6.1 Data components

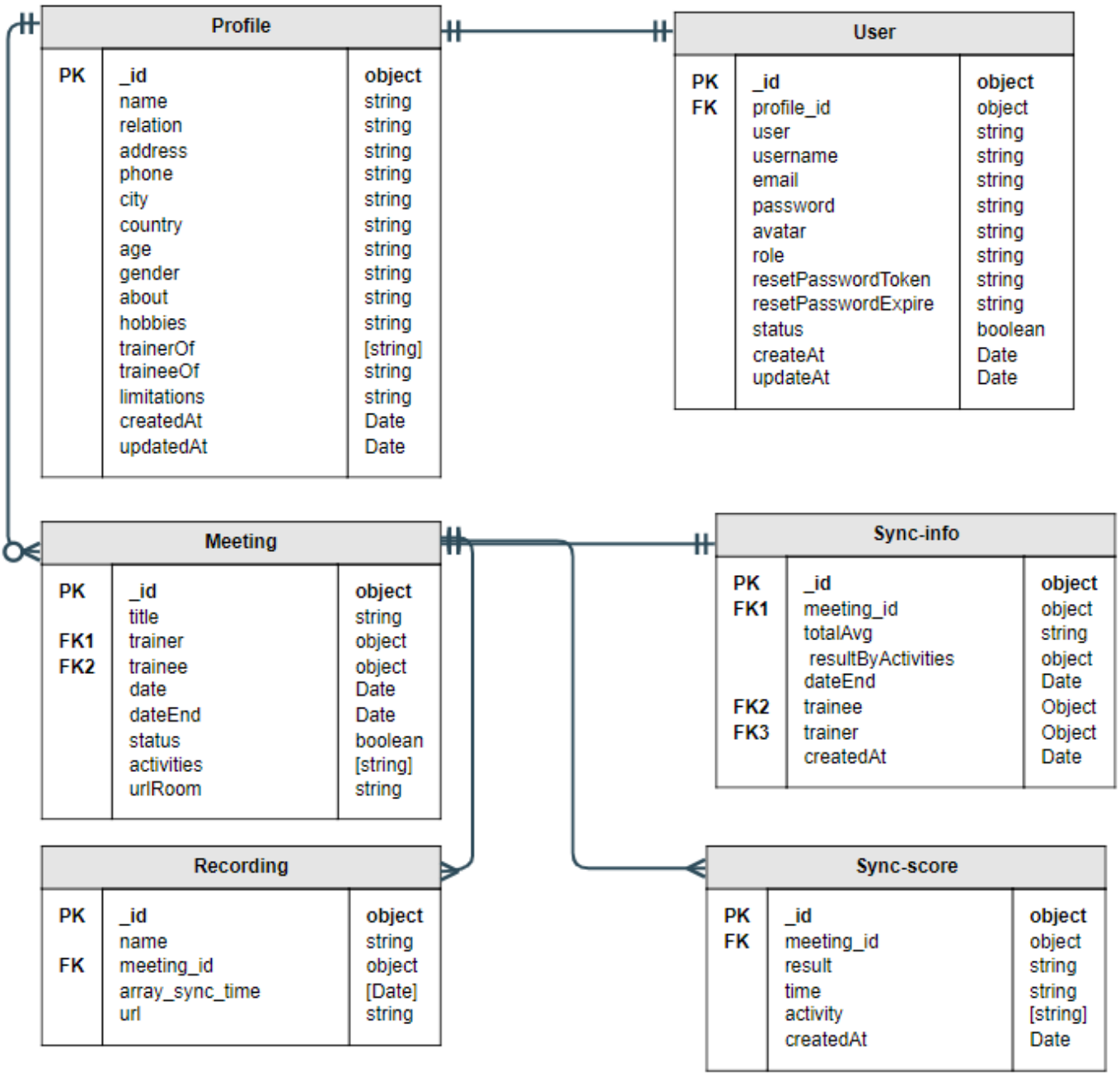


Figure 6-1: Data components

6.2 Class Diagram

The context between classes, attributes, methods and operations.

Trainer and Trainee class inherits from the User class , Trainer and Trainee are object from the inherited User class.

Room(Active Meeting) class has a composition relationship and, it exists only with the existence of Activity, ScheduleMeeting, PoseMessage, SyncScore and Voice Recognition classes.

AR Feedback class exist only for a unique user, composition relationship and to User class and to SyncScore classes

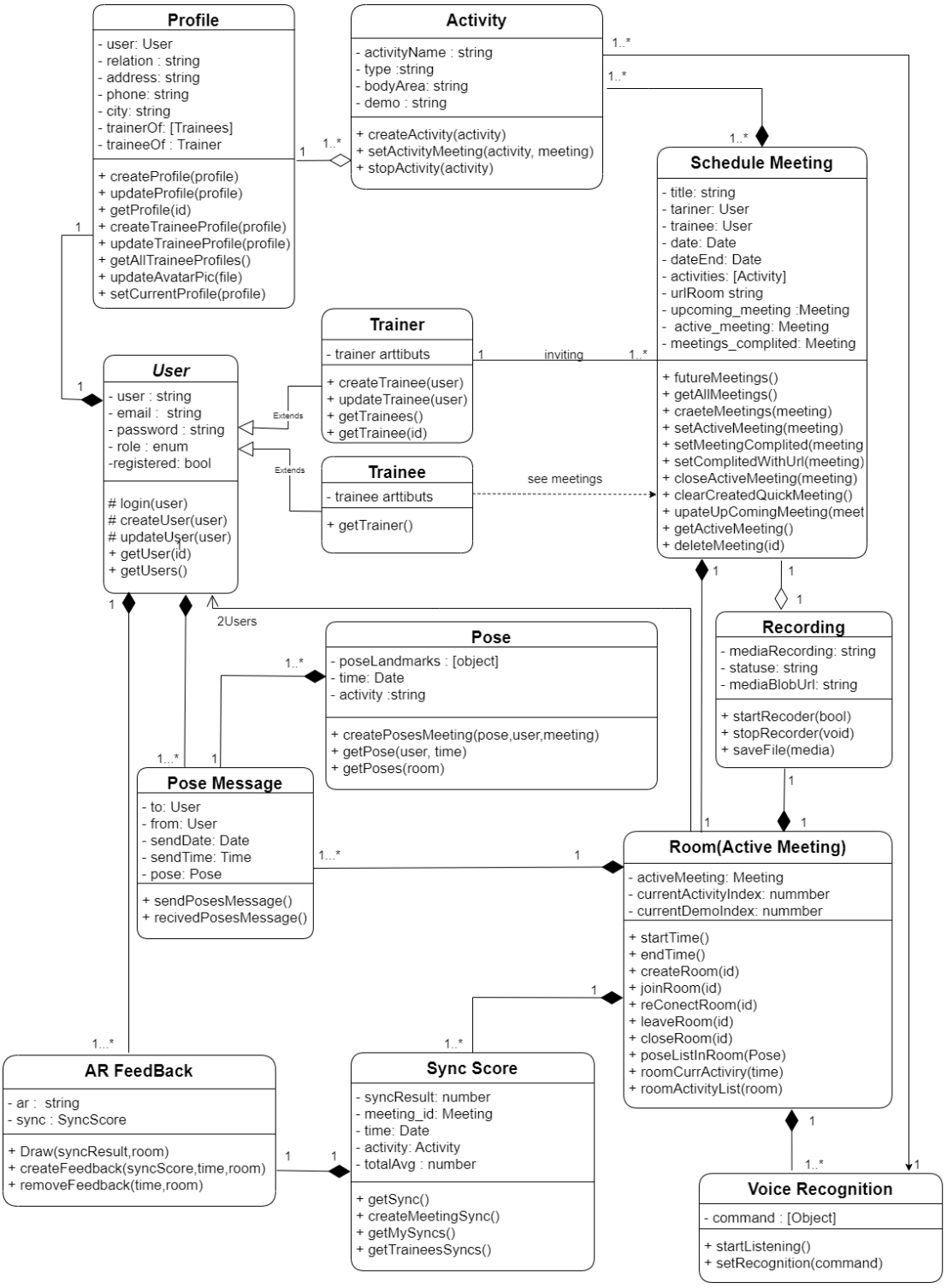


Figure 6-2: Class Diagram

6.3 Communication components (user-computer interaction)

SyncAlong system has two types of users: an adult user and a trainer use

1. Trainer user has high privileges and can perform all the functions related to the operation of the system, also has access to the data of the trainees who is on his list of responsibilities and has full CRUD excess.
2. Trainee(edarly user) has low privileges, e.g. view meetings, view and change profile, ability to accept or reject receiving a request to connect with a trainer user to sport session. minimum functionality for a convenient and friendly interface for the adult user.

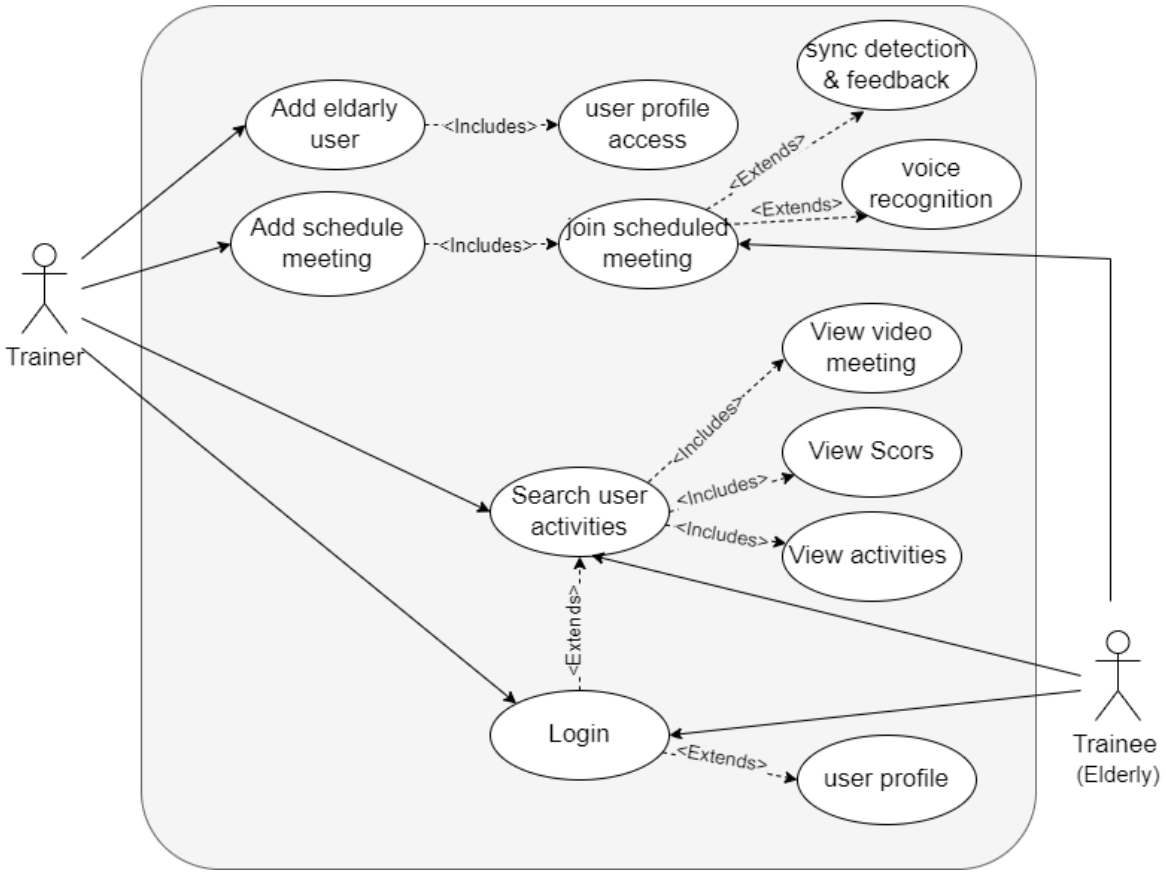


Figure 6-3: user-computer interaction

6.4 Activity diagram

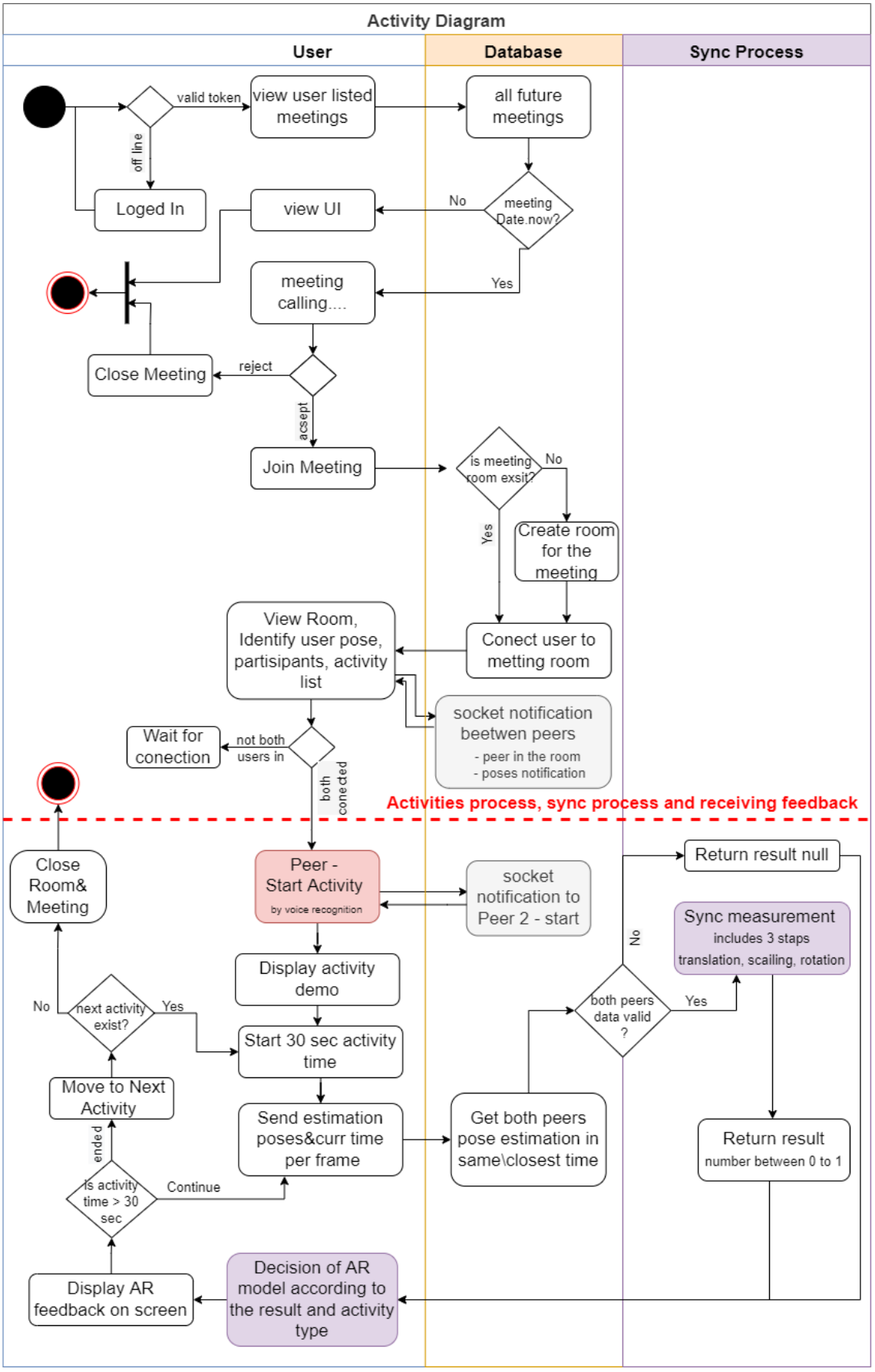


Figure 6-4: Activity diagram - sync process

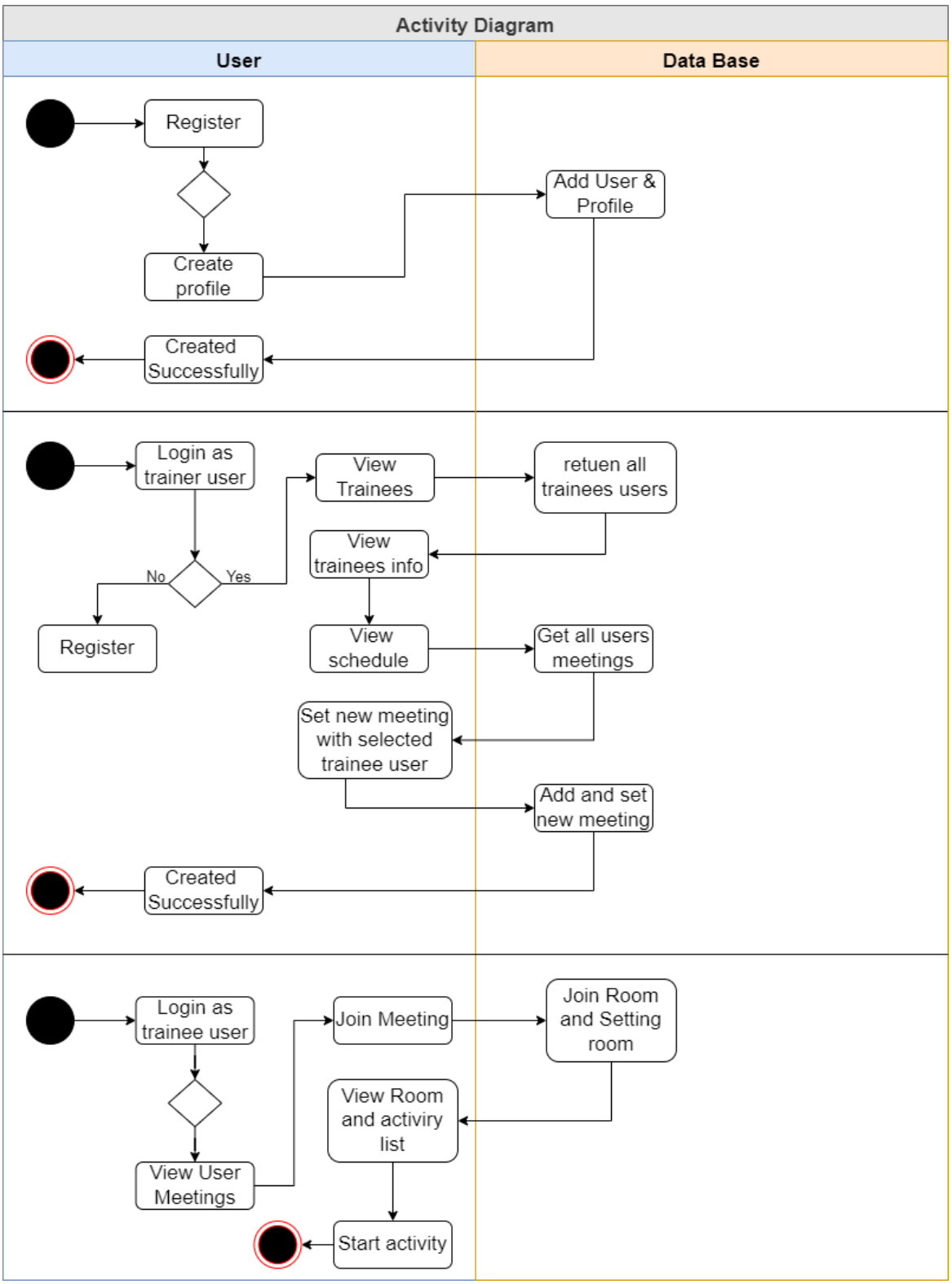


Figure 6-5: Activity diagram - users process

6.5 Sequence diagram

The process of starting a joint sports meeting, during the joint connection - initializing the list of exercises and going through the list of exercises, which includes: displaying the demo, transferring data of human positions and displaying the feedback when the result of synchronization.

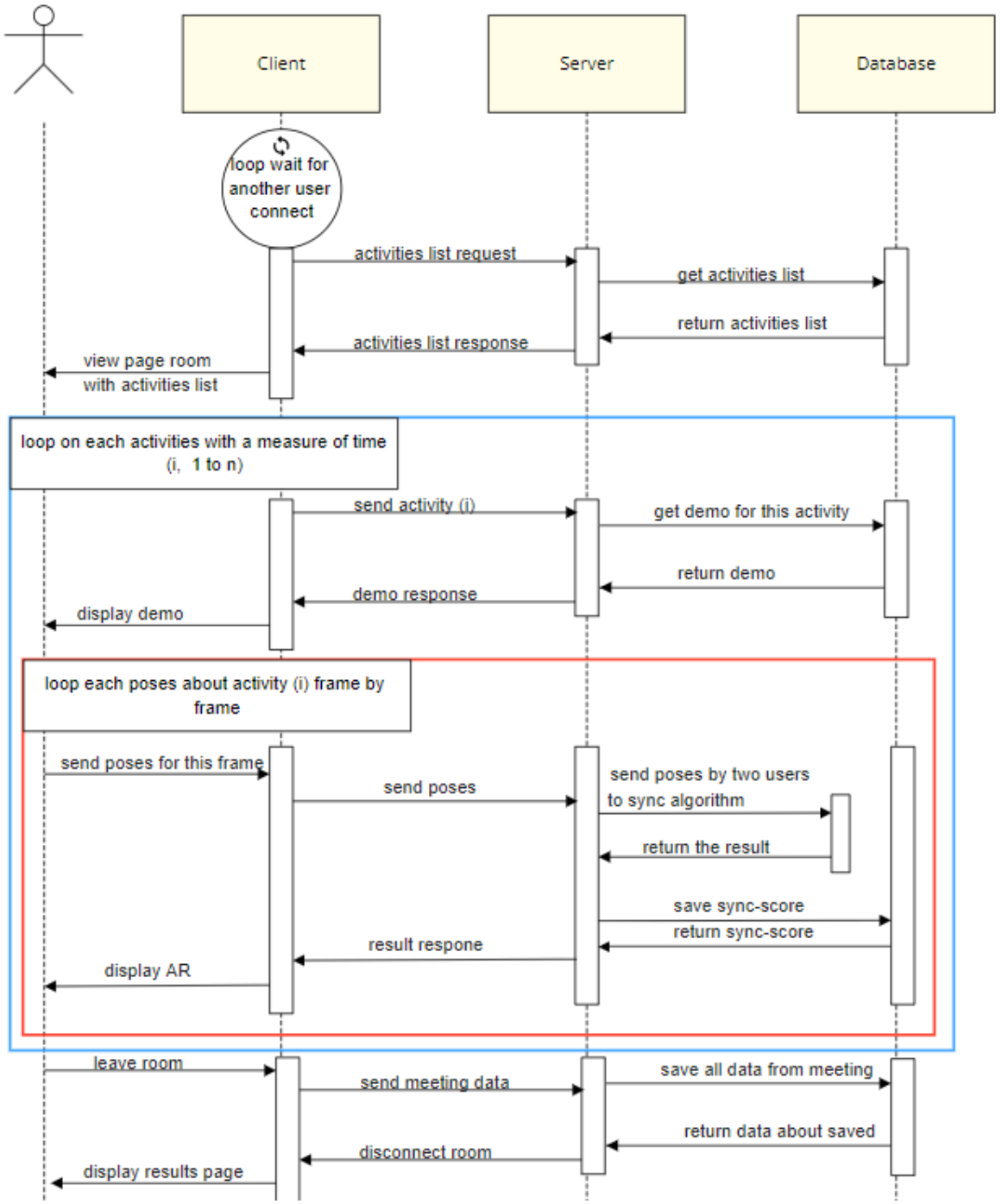


Figure 6-6: Design sequence diagram

7 IMPLEMENTATION

SyncAlong system is a **web application** built as a client-server software application in which the client (or user interface) runs in a web browser and the server is run on a remote server. The system supports **real-time** communication between users through a **socket.io**-based server, a dedicated server for real-time requests and handles the transfer of information between users in real time. The need for a system that supports real-time requests comes from one of the main aspects of the system - to communicate between participants. Socket.io-based **real-time communication** allows information to move between users through the server in a continuous, cost-effective and documented manner.

Socket requests support:

- Request for initial contact
- Repeat response
- Updating user actions to another user, e.g : updating an older user that a trainer user has scheduled a meeting for a few more minutes.

In addition to real-time communication between users using the system, The system supports real-time communication through webRTC which helps real-time data communication between two peers. To connect two users via WebRTC, we exchange information to allow browsers to talk to each other. This process is called signaling. Apart from signaling, there is no need to send data through a server. When a connection is successfully established and authentication and authorization are complete, flow data exchanged between peers is directed to the React component for processing and passing information through the browser.

The use of real-time communication in the browser (webRTC) takes place in the system when requesting to connect users to a video chat meeting, which is actually a connection to the shared sports meeting between an adult (trainee) user and a trainer user.

During the joint sports meeting, information is transferred between the Peers for each frame in two communication channels:

1. Through the socket.io channel (including passing through the server) - for the purpose of calculating the synchronization algorithm
2. Through webRTC channel exchanging data from browser (without server side) - for the AR feedback to the screen

7.1 Algorithm Design

7.1.1 Algorithmic component diagram

Synchronization algorithm includes two dependent aspects:

1. Definition of similarity between two different people landmarks by Procrustes analysis
2. Definition of similarity between two normalized landmarks by Fréchet distance

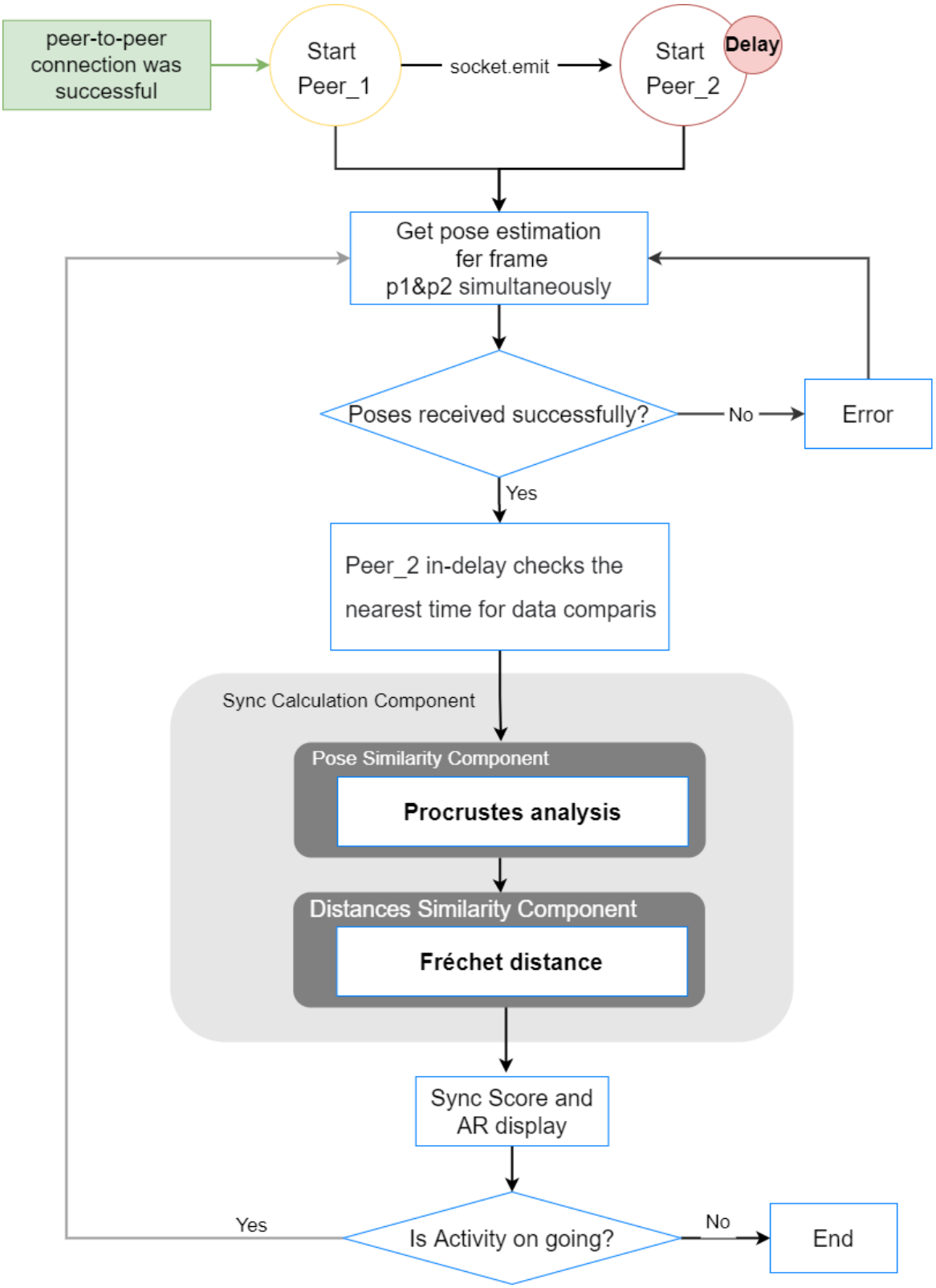


Figure 7-1:SyncAlong sync algorithm diagram

7.1.2 Expansion of the algorithmic component

When we think about the problem,we see that there are many uncertainties to be addressed: humans can have different heights and body shapes, they might be in different parts in the picture: one person may have been standing close to the camera, another might have been faraway. All these problems have to be solved in order to output a correct result. we to take all the steps followed Procrustes analysis

The comparison is made using the square root and the calculation of distances between them.

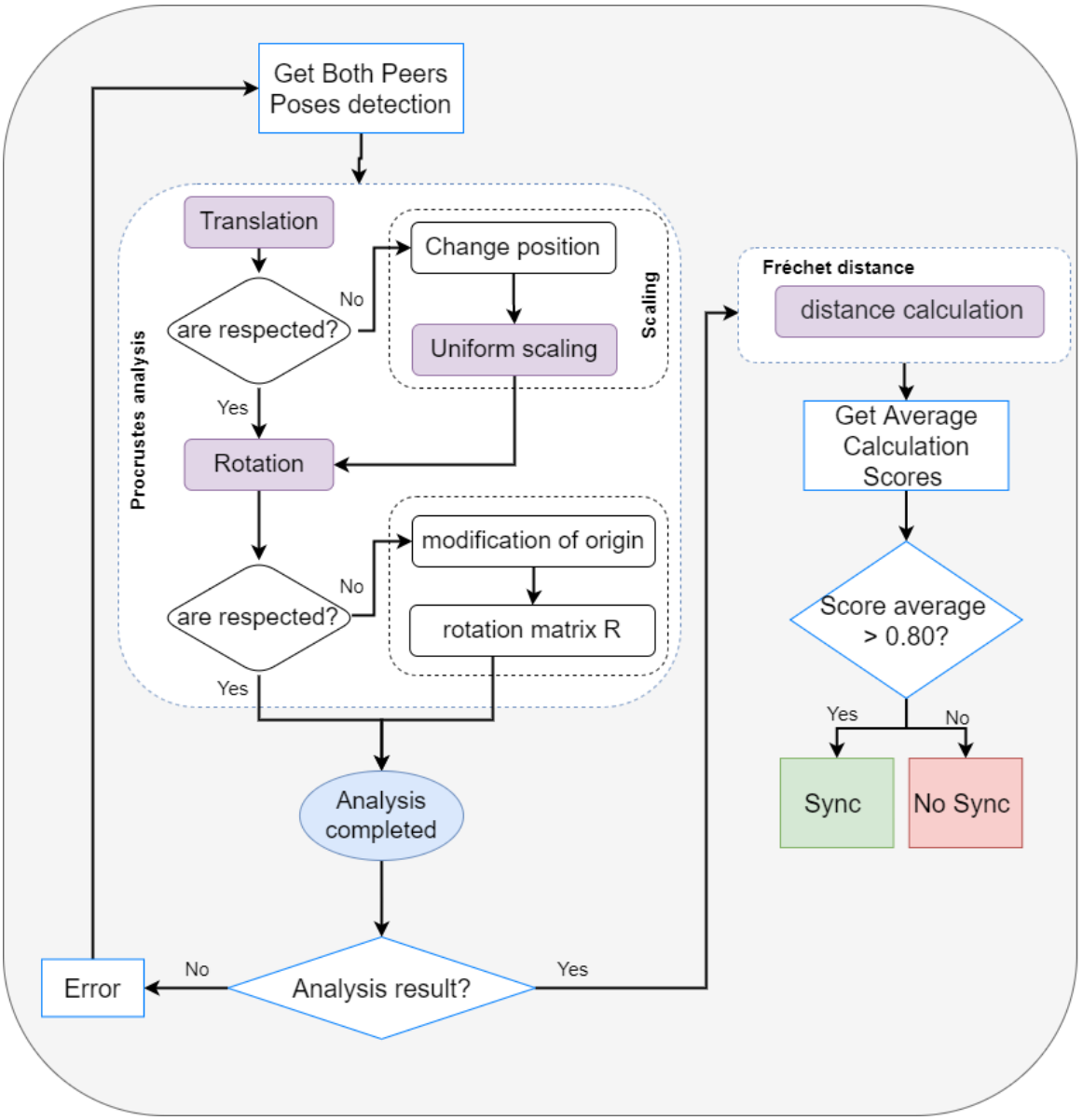


Figure 7-2: Sync algorithm steps

Procrustes analysis Steps , similarity of 2 shapes:

- **Translation**:

  The translation of the object is defined by changing the object so that the average of all points of the object (i.e. its center) is at the origin.

  - An object is a list of points landmarks human pose estimation.
  - The change is performed on both objects together.

- **Scaling**

  The scale component can be removed by scaling the object so that the root mean square distance (*RMSD*) from the points to the translated origin is 1. This RMSD is a statistical measure of the object's **scale** or **size**

$$s = \sqrt{\frac{(x_1 - \bar{x})^2 + (y_1 - \bar{y})^2 + \cdots}{k}}$$

  The scale becomes 1 when the point coordinates are divided by the object's initial scale:

$$((x_1 - \bar{x})/s, (y_1 - \bar{y})/s).$$

- **Rotation**

  Removing the rotational component is more complex, as a standard reference orientation is not always available. Consider two objects composed of the same number of points with scale and translation removed

  A rotation by θ angle $(u_1, v_1) = (con\theta w_1 - sin\theta z_1, sin\theta w_1 + con\theta z_1)$

  Where (u, v) *are the coordinates of a rotated point. Taking the derivative of* $(u_1 - x_1)^2 + (v_1 - y_1)^2 + \ldots$ *with respect to* θ *and solving for* θ *when the derivative is zero gives*

$$\theta = tan^{-1}(\Sigma^k_{i=1}(w_i y_i - z_i x_i)) / (\Sigma^k_{i=1}(w_i x_i + z_i y_i))$$

Fréchet distance Steps, similarity of 2 shapes distance:

- Set array of max distance for each angle in tha angles-list-options
- Select min from max-array-distance

$$F(A, B) = inf\ max\{d(A(\alpha(t)), B(\beta(t)))\}$$

7.2 Interfaces

This section will present the user interface, which is the web-based application. The section will include screenshots of the different screens of the GUI.

7.2.1 Human-computer interfaces (GUI)

First trainer screen – default view: The first screen displayed to the trainer user is a view of the adult list, a view of the nearest appointment and the time left until its start, a data graph showing the sync of all users under his responsibility, future appointments scheduled with the older users under his responsibility.

The user can make a "quick meeting" on this page with an older user who is connected to the system at the same time - this choice will result in socket requests being sent between users until a connection is established and the shared meeting page will be routed. The session supervised in this way will include a set of a random list of sports exercises.

(Figure 6-3).

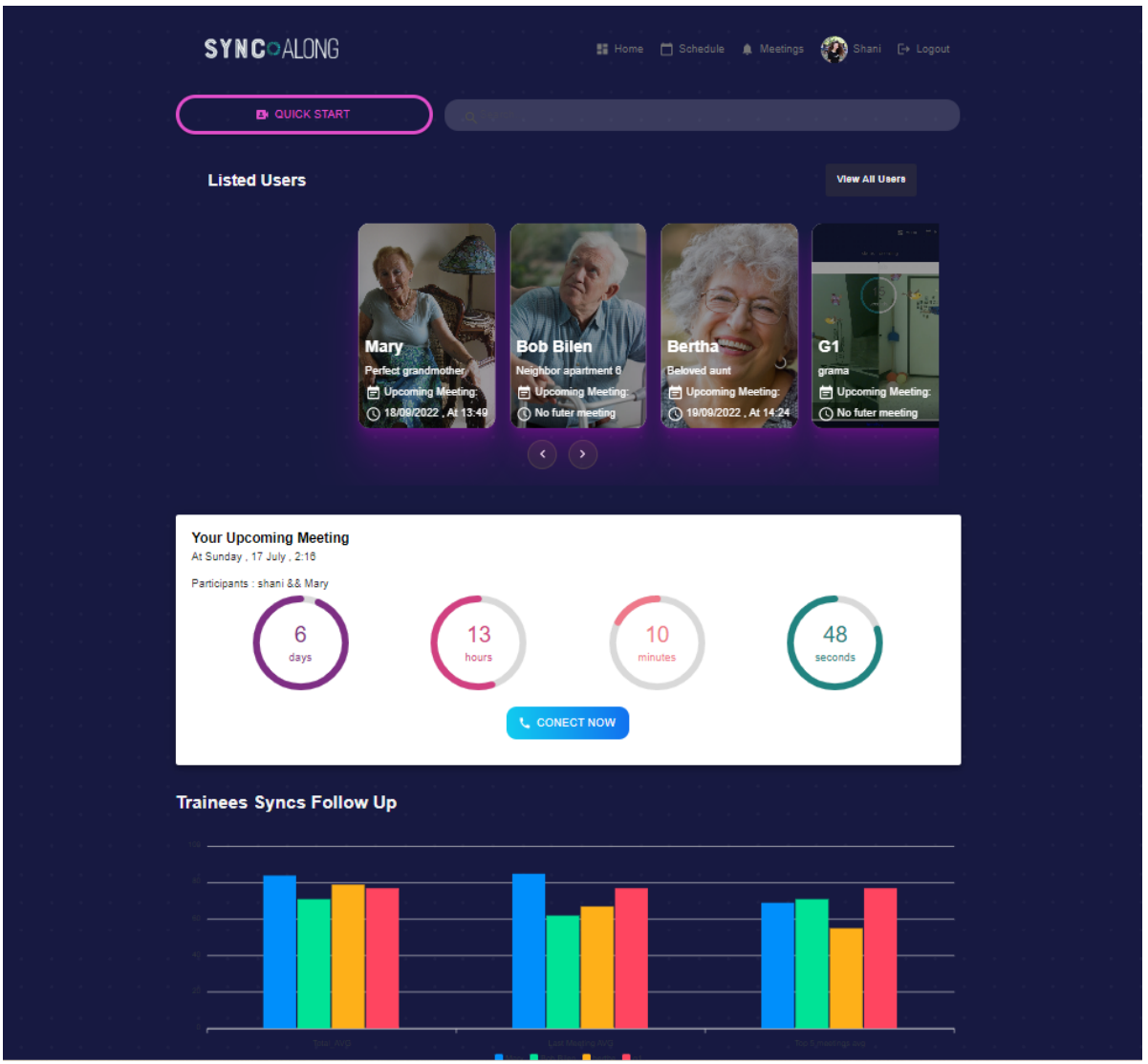


Figure 7-3: Default view (First trainer screen)

Adult User Information Page (for Trainer User use): A trainer user can view detailed information of each adult under his responsibility, the adult page includes information about the appointments he has made, synchronization metrics, progress over time, and access to the user profile with full functionality (CRUD) and the possibility of scheduling future appointments with the same user (Figure 7-4).

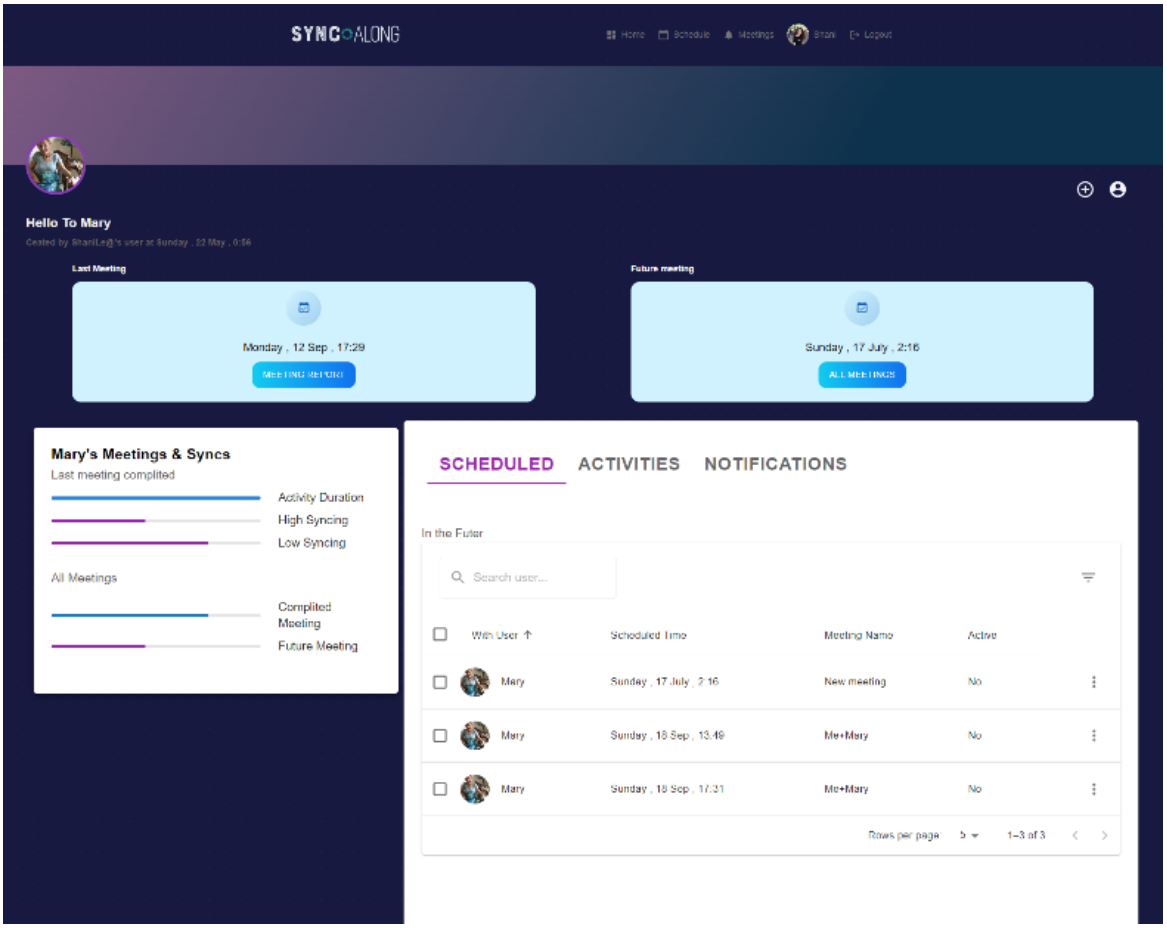


Figure 7-4: Adult User Information Page (For Trainer User Use)

Trainer Profile page : allows changes, adding more adult users, direct access to the older user

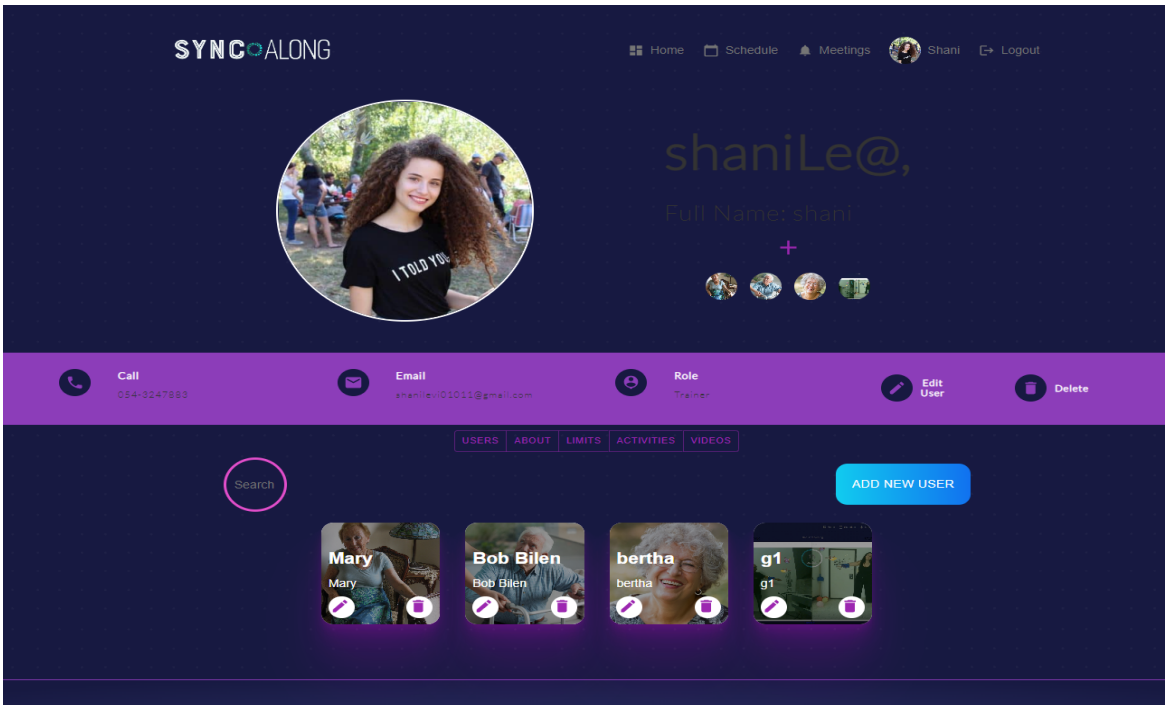


Figure 7-5: Trainer Profile Page

Add New Trainee: Adult Adding Page: Includes user page first page, user profile second page. (Figures 7-6).

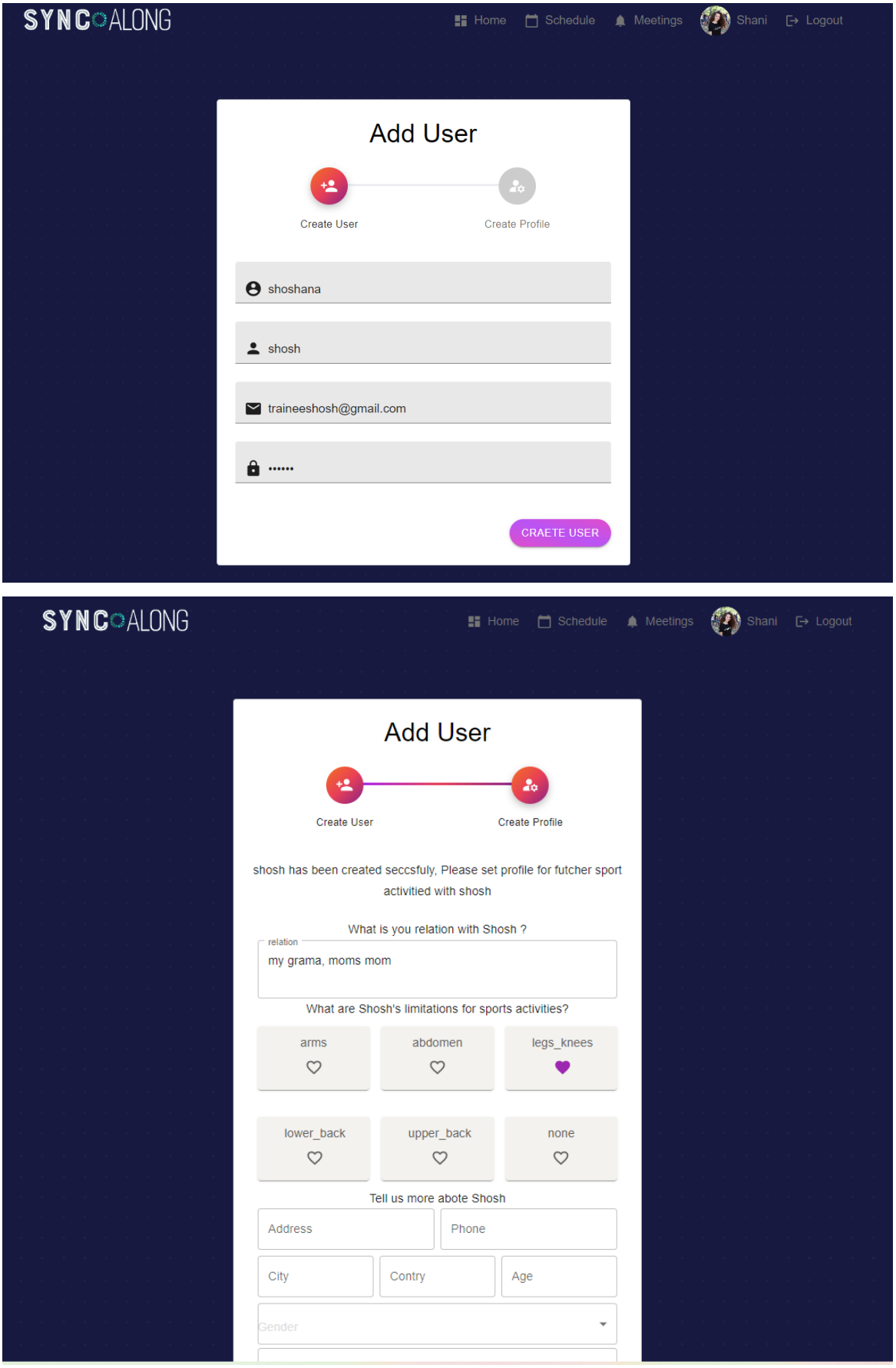


Figure 7-6: Add New Trainee

Create New Meeting: Figures 7-7 describes creating a meeting through the meetings page, Figure 7-8 describes creating a meeting with online users, exercises in a random way.

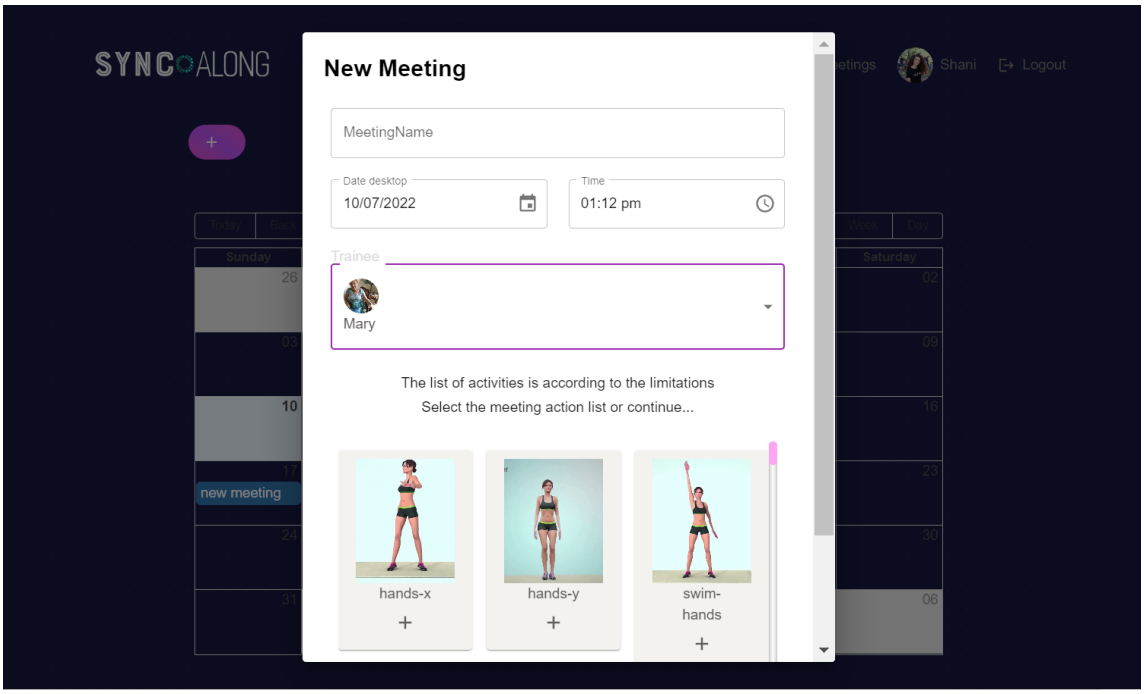


Figure 7-7: Scheduling a new sort meeting , trainer user functionality.

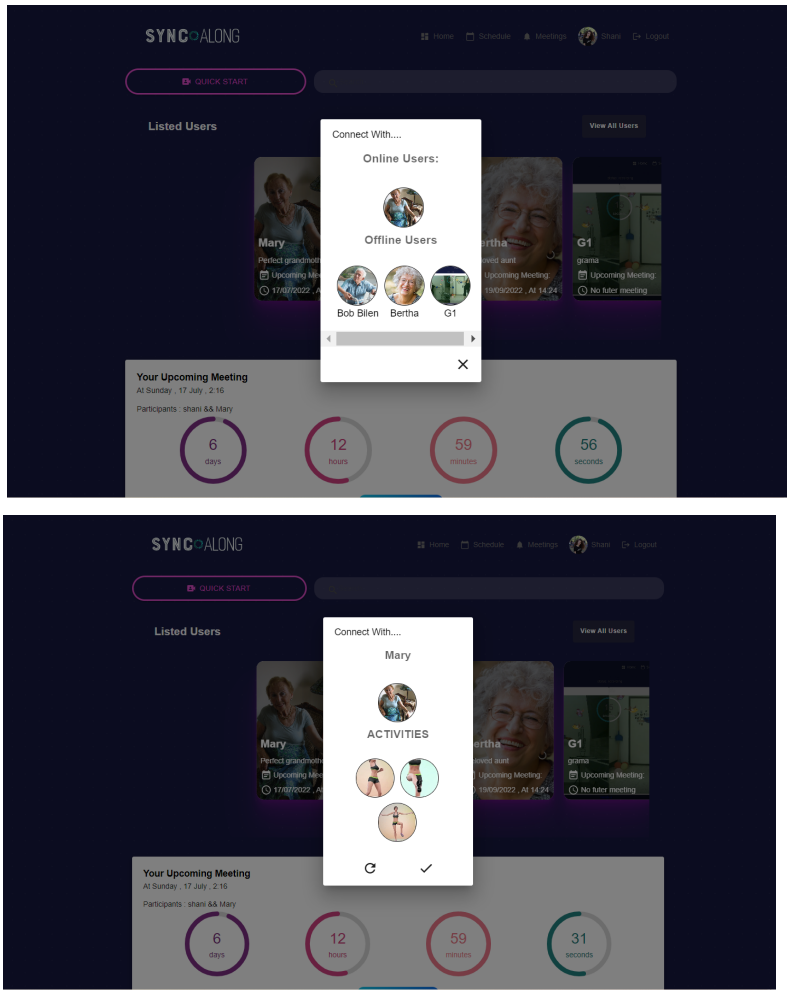


Figure 7-8: Quick meeting button , trainer user functionality.

Adult User Home Page: Displays the upcoming appointment for the coaching user and information about future appointments and appointments made.

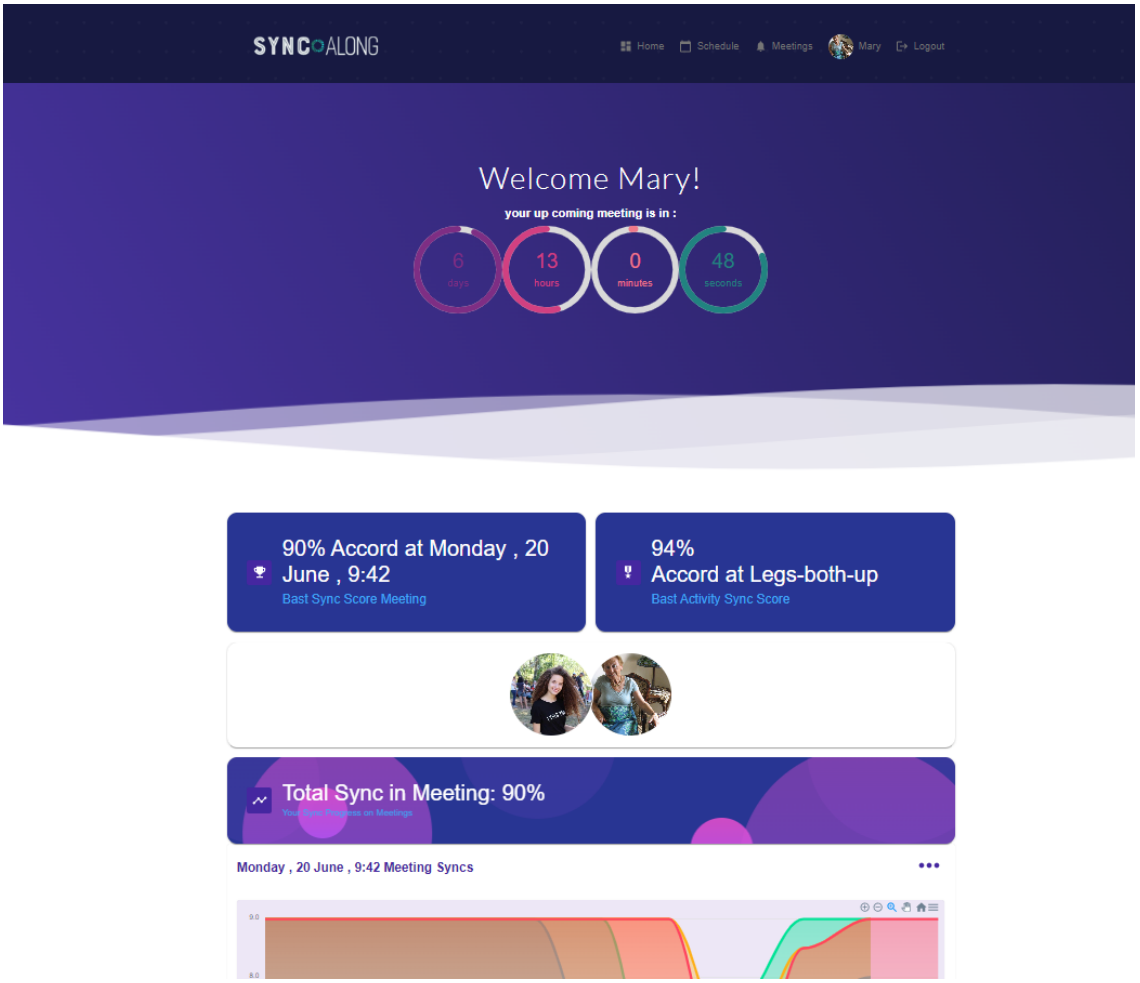


Figure 7-9: Adult User Home Page

Incoming call: Incoming call to connect for a joint meeting. Answering a call will result in the transition to the shared meeting page which includes a connection media stream and audio

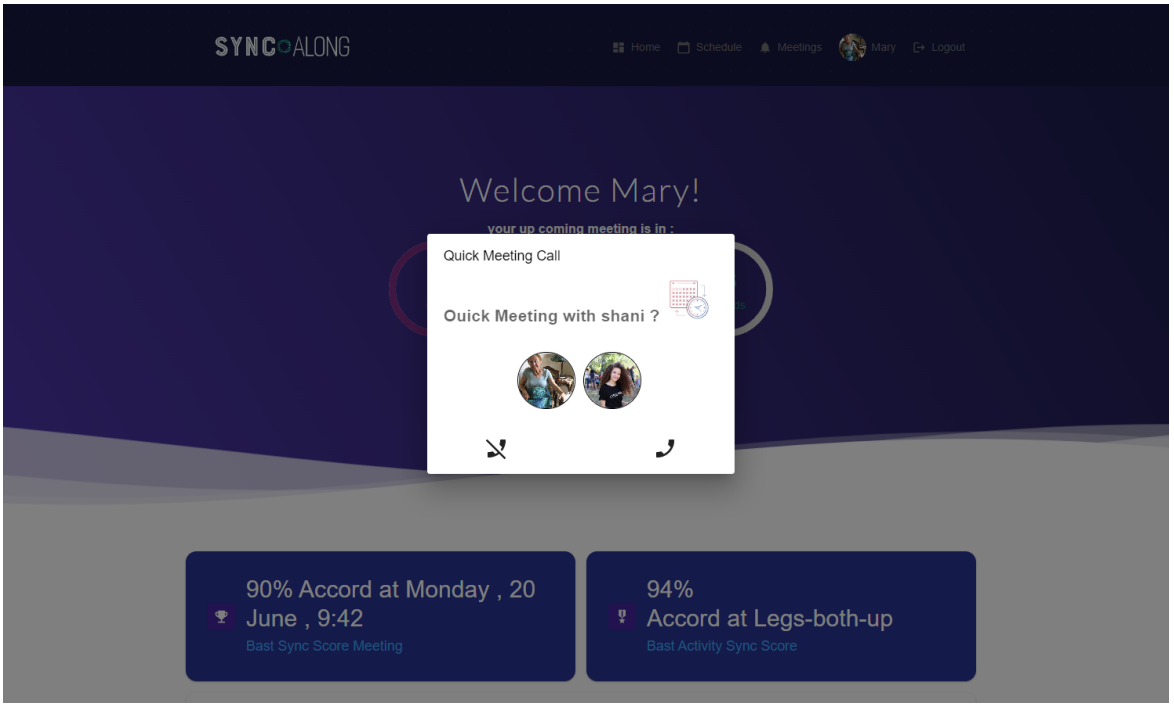


Figure 7-10: Incoming call

Start sport meeting by demo display: after connecting users the sport activity is starting by viewing the activity demo.

The list of exercises for that session is predefined and determined by the user coach when scheduling the session. The activity demo is 4 seconds long..

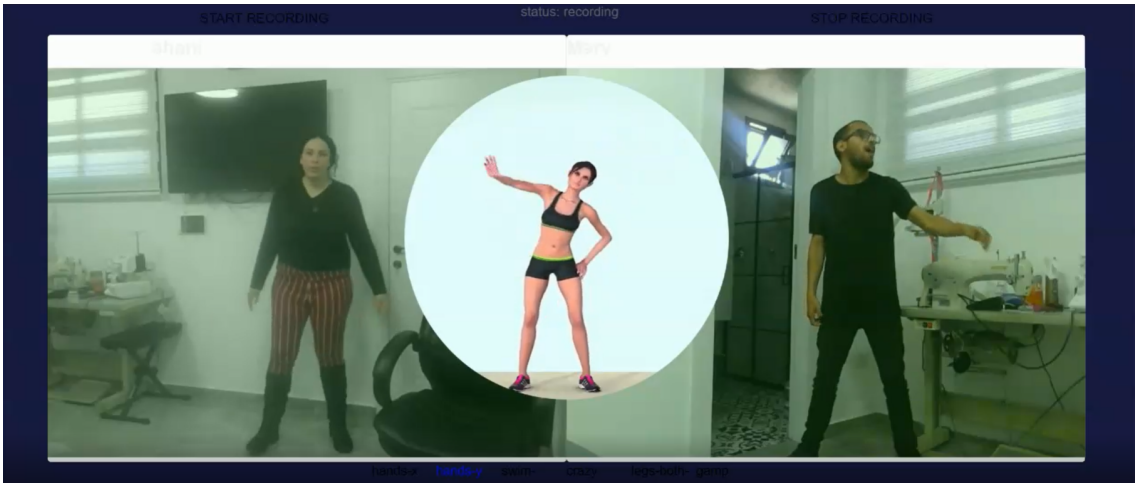


Figure 7-11: activity demo display

Activity Action: The duration of one activity is 30 seconds. During the activity time the users perform an action according to the demo and when a synchronization is detected, positive feedback of animation will be received.

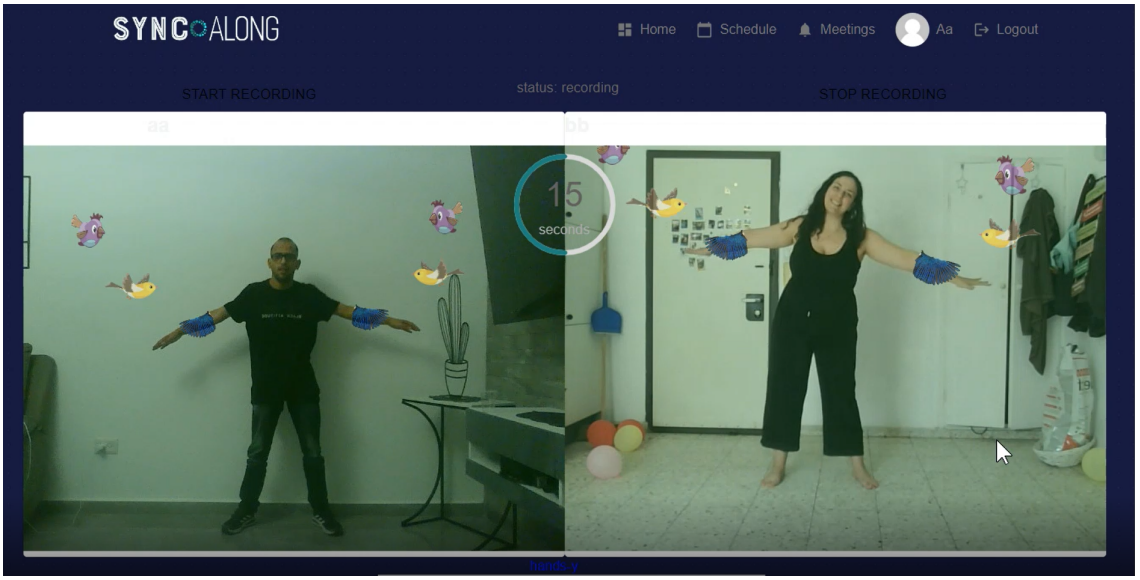


Figure 7-12: Activity Action

Sync Results Page: A summary page is created after completing a list of exercises in the session, including recording and synchronization data throughout the duration of the session.

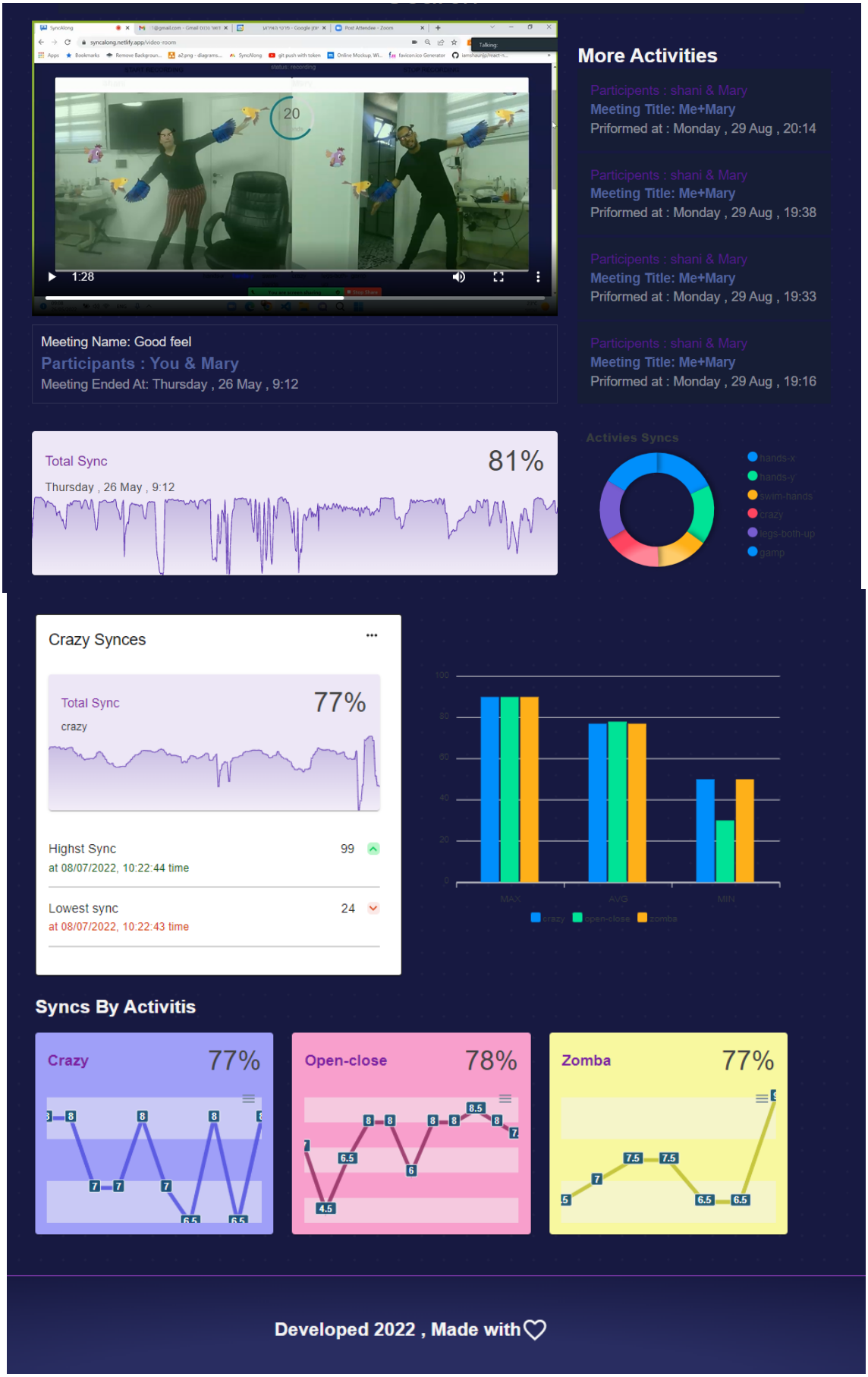


Figure 7-12:Sync Summary Page

7.3 Development environment

- Project SyncAlong developed in server and client side in JavaScript.
- Front End - Using ReactJs Hooks with Redux using Material UI library for ui-design.
  - Includes use of webRTC architecture through peerjs and socket libraries.
  - Includes use of MediaPipe Google AI library for pose estimation model.
- Back End - Using NodeJs with express ,sockt.io , authentication and authorization sports.

7.4 Programming languages

Based on the selected technologies specified in the technology survey section:

- The client-side - this part is composed of components which were developed using:
  - JavaScript
  - ReactJS using Redux for state management
  - HTML, CSS, Material UI
    - Development through the React open source library saved the team time learning and developing in a new language. We also support external libraries for accelerated component development user interface. Since this language is based on the JavaScript language we created uniformity in the whole system on the client-server side

    Main Libraries used:
  - PeerJs library acts as a wrapper over WebRTC
    - The use of WebRTC designed for high-performance media delivery meets the system requirements in the best way and therefore we used WebRTC for connection.
  - Socket.io-client library
  - MediaPipe GoogleAI
    - MediaPipe offers us a fast framework that although it has limited use at the moment it addresses the needs of the system
  - Voice recognition

- The backend side - this part is composed of components which were developed using:
  - Node.js & MongoDB database
    - Because the client-side development language is executed in JavaScript, maintaining consistency and usage in one language the development of the whole system (Full-Stack) will lead to a great advantage , which is why NodeJS is chosen for server development language.n.

7.5 Limitations of the system

The system restricts the user to make a multi-user sports session.It **allows two users to connect to a shared meeting** and transfer information between them. In fact, the system was designed to communicate with a single user when connecting to him and referring to him was made according to the ID of the room and the user's ID only.

On the client side, when making a connection request and answering the connection, the users know at that time what the identification of the participant in front of them is and what the identification of the common room is, according to which throughout the serial meeting the transfer of information between them is done directly according to the unique identification. The system is built so that this is possible with both Socket and webRTC data transfer, The connection of the webRTC is made in front of Peer to each other according to the information passed between them in the Socket.io data transferring at the time of the initial connection approval. In the aspect of synchronization calculation, **the synchronization calculation is based on the assumption that the information obtained is for two users only**.

The human pose estimation is collected in real time when the user enters the joint sports meeting and the information that passes between the participants(peer1 and peer2) is according to the unique id of the room or id of another peer. The sync-algorithm is based on Procrustes analysis that is used according to a comparison of two variables.

Procuster analisis has another step for multi-variable comparison for multi-object matching but it changes the simplicity, efficient and fast calculation (which is the calculation time of the algorithm is important for a quick response on the customer side to provide feedback in a response that simulates real time)

Moreover, the nature of the project was variable, as it was required to define what multi-participant synchronization was. Out of 3 participants only 2 are synchronized - is there synchronization in the room? Is the synchronization measured in front of a "source" participant or is it necessary to define a loop that checks each participant as a "source"? Which micromanages the algorithm and may impair real-time performance.

7.6 Installation and configuration issues

The SyncAlong system does not require an installation process that includes setting up variances.

The available environments are based on Windows. Mac and Linux have not been tested, but there is not expected to be a problem with these interfaces.

In order to start a system sport session, access to the computer's camera and audio is required. Without this access it is not possible to carry out the joint meetings in the system.

7.7 Information Security

SyncAlong system has implemented several useful information security tools:

1. User authentication process - users can be identified by logging in to email or username and password. The user receives a token from the central server with which the client side is identified in the sending requests to serve.
2. User password encryption with Salt and Hash algorithm and writing passwords after the encryption in the database
3. Secure data transfer between client side and system servers over HTTPS protocol which encrypts the transmitted data and protects the system from man in the middle attacks.
4. Restriction of user privileges (permission) - Restriction of access to functionality central to the system for users. For example, only trainer type, non-adult, may create additional users, older users, and allows to schedule an appointment with them and manage the activity schedule with them.

   The system has two types of users with different privileges:

   - Trainer user - has the highest permission and can perform all actions.
   - Trainee user - an elderly user, has viewing permission only and minimal functionality such as changes in personal profile.

7.8 Risk Management

- **Sync algorithm, accuracy and reliability.**
  - For dealing the risk we performed a test which is described in detail in part 8, verification of the main algorithm in three stages:
    - Splitting the assessment procedure by regions.
    - Changed evaluation of rotation angles to improve accuracy.
    - Synchronization meter was evaluated according to repeated tests.
- **Delay response in users connection.**
  - After several attempts of sending information in the register and measuring the time from the moment of sending to the moment of return. It was jointly decided with the moderators that in order to reduce information transfer times, the users transfer information to each frame and directly to the server. The response is at the same time by socket respons to the room.id .
    - The delay time tests are reviewed and detailed in section 8.
- **Delay of AR feedback.**
  - Further to the tests of delay during response from the server, in light of the implementation of AR on the page when receiving the information - no additional delay was created in the response of the feedback.

7.9 Exceptions Management

Error management and handling of **client-side** exceptions:

1. Validation folder: Includes small code files that include spot checks for re-checking such as: checking if an object is empty or not initialized, checking its date - whether it is future time or not required for identification in the system for viewing future meetings, and so on

.

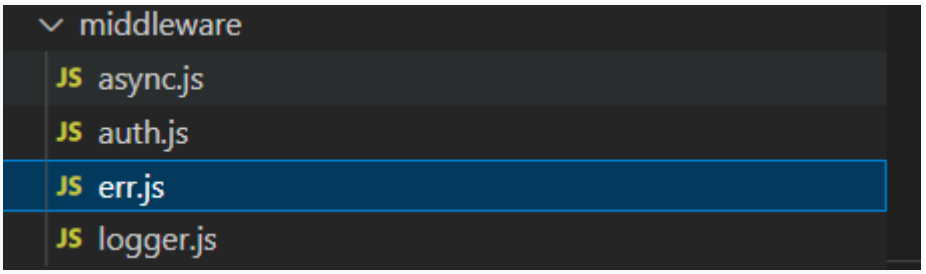
2. Performing tests according to server calls for initializing the data required by the server and capturing the error and presenting it to the user without performing a server call.

3. Handling errors in FORM throughout the system using a global variable via redux, when a server error is detected that the redux store returns, it will update the variable and the system knows how to listen to this variable and give feedback to the user.

```
42      const [errors, setErorrs] = useState({});
43
44      useEffect(() => {
45          if (props.auth.resisterd) {
46              navigate('/auth/login');
47          }
48
49          let errors = {};
50          if (props.errors === 'Duplicate field value entered of email') {
51              errors["email"] = "*email already exists";
52          }
53          if (props.errors === 'Duplicate field value entered of usename') {
54              errors["nickname"] = "*nick name already exists";
55          }
56          if (props.errors === 'Please add a valid email') {
57              errors["email"] = "*not valid email";
58          }
59          if (!isEmpty(errors))
60              setErorrs(errors);
61          return;
62      }, [props.errors, props.auth.resisterd])
63
```

Error management and handling of **server-side** exceptions:

1. The server is wrapped in a Middlever whose job is to catch 400-500 server errors
   Managing cases and returning the response type from the server is managed in a file named err.js



2. In addition to instances of authentication of server call accesses per call by a logged in user with a token or public request

```
6   //for privit routes
7   // Checks whether the token of tha user is valid or not
8   exports.protect = asyncHandler(async (req, res, next) => {···
33  });
34
35  // Grant access
36  exports.authorize = async (friendId, trainerOf) => {···
44  };
```

3. Each function on the server is wrapped in an auxiliary function of try...catch
   This way we make sure that in each function a try is executed first, and if it throws an exception, the code in the catch block will be executed.

```
6   const asyncHandler = fn => (req, res, next) =>
7       Promise.resolve(fn(req, res, next)).catch(next);
8
9   module.exports = asyncHandler;
```

7.10 Versions Control

For version management we used git for local management and github for shared management

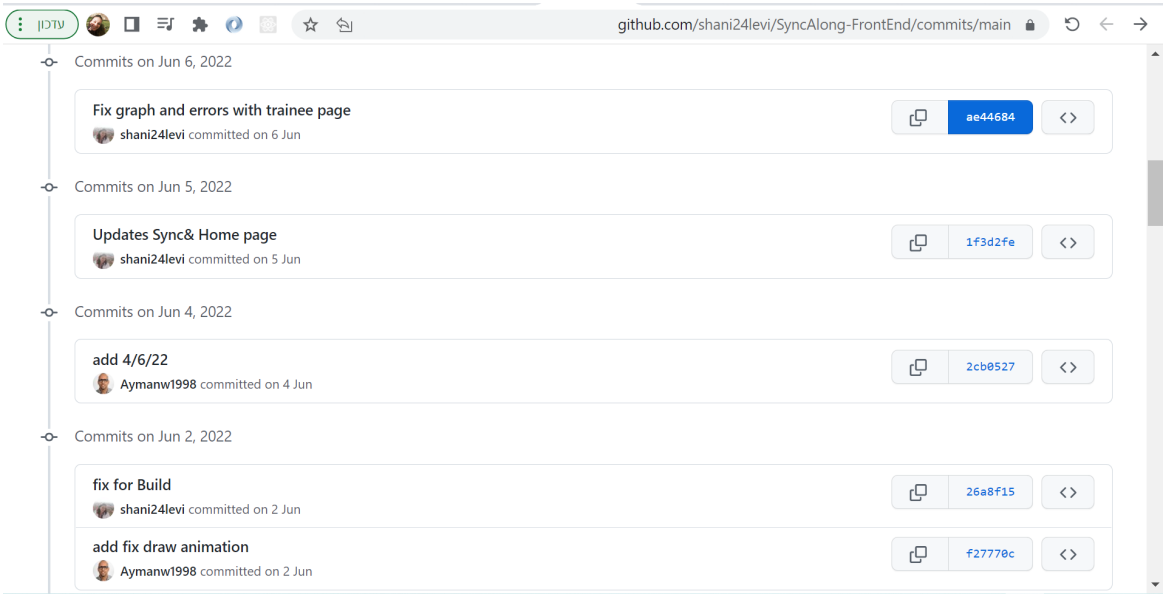


Figure 7-13: version control

7.11 Project Management

This section includes tasks and time management.

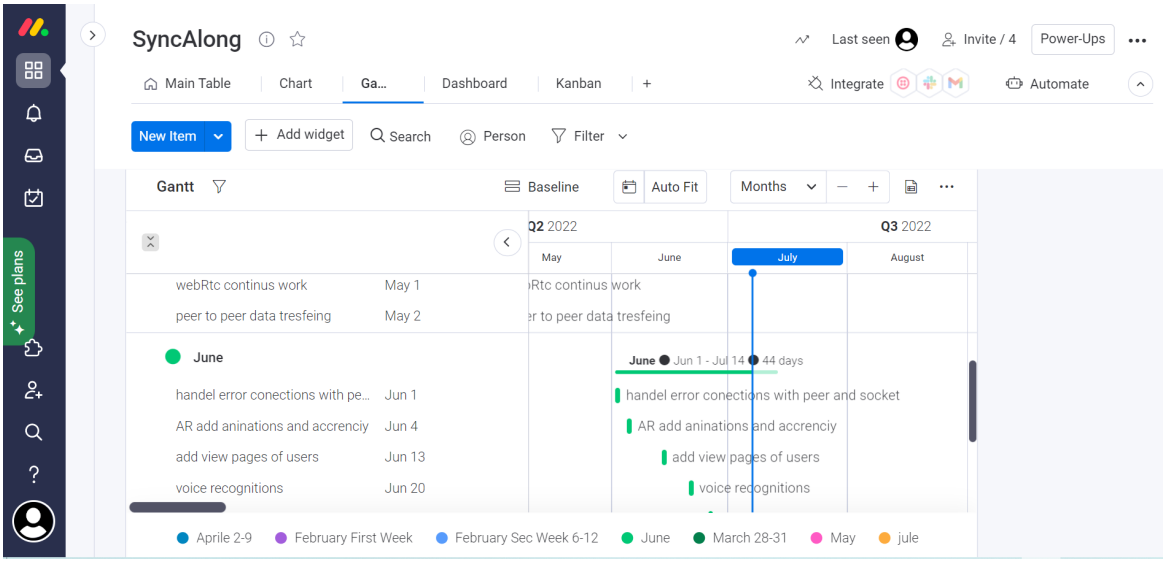


Figure 7-14: Project Management

7.12 Code

client :
https://github.com/shani24levi/SyncAlong-FrontEnd

server-https:
https://github.com/Aymanw1998/SyncAlong/tree/master/Server

server-socket.io:
https://github.com/Aymanw1998/SyncAlong/tree/master/ServerConection

8 SYSTEM VALIDATION

1. **Data accuracy test from the Google AI (Media Pipe) model:** During development and learning how to use and the data obtained from the Media Pipe library, we performed repeated tests to identify pose estimation and examine spot tests to isolate a case of misidentification. The library provides a secure visibility variable for each detection point 0-1 that defines the percentage of reliability / success of the detection,

   After performing several tests - a measure less than 0.6 was defined as a low measure of spot identification in space.

2. **Time Delay Test**: It is required to find out what causes the biggest performance delay in the project, in order to do this we have isolated the time that each process takes. We measured the time delay from the initial retrieval of the data to the receipt of the data and then we measured the time delay of each function, method and process

   Test tests

Delay time from moment of sending back	הבדל זמן ביניהם (במילי שניות)	Peer2-client-time-receiving	Peer1-client-time-receiving	Time after algorithm & sending back	Time before send algorithm	Peer1-time-receiving in server	Peer1-time-sand from client	Test id
Less than half a second	39	8:47:03 571	18:47:03 610	18:47:03.293	18:47:03.286	18:47:03.257	18:47:03.256	1
Close to a second	35	18:57:14 120	18:57:14 155	18:57:13.512	18:57:13.502	18:57:13.478	18:57:13.475	2
half a second	148	19:01:22 548	19:01:22 696	19:01:22.167	19:01:22.136	19:01:22.090	85 19:01:22	3
Less than half a second	173	19:10:15 54	19:10:15 227	19:10:14.75	19:10:14.717	19:10:14.682	679 19:10:14	4
Close to a second	300	19:12:15 932	19:12:16 319	19:12:15.690	19:12:15.676	19:12:15.646	643 19:12:15	5
half a second	350	19:14:00 383	19:14:00 741	16:14:00.109	16:14:00.094	19:14:00.055	53 19:14:00	6
Less than half a second	217	19:17:19 115	19:17:19 332	16:17:18.943	16:17:18.929	16:17:18.896	893 19:17:18	7
Close to a second	100	19:18:48 287	19:18:48 378	16:18:48.011	16:18:47.998	16:18:47.964	961 19:18:47	8
Less than half a second	74	19:19:55 855	19:19:55 781	16:19:55.432	16:19:55.418	16:19:55.377	374 19:19:55	9
Less than half a second	690	19:20:53 962	19:20:54 345	16:20:53.821	16:20:53.807	16:20:52.762	769 19:20:53	10
Less than half a second								TOTAL

Figure 8-1: timing tests

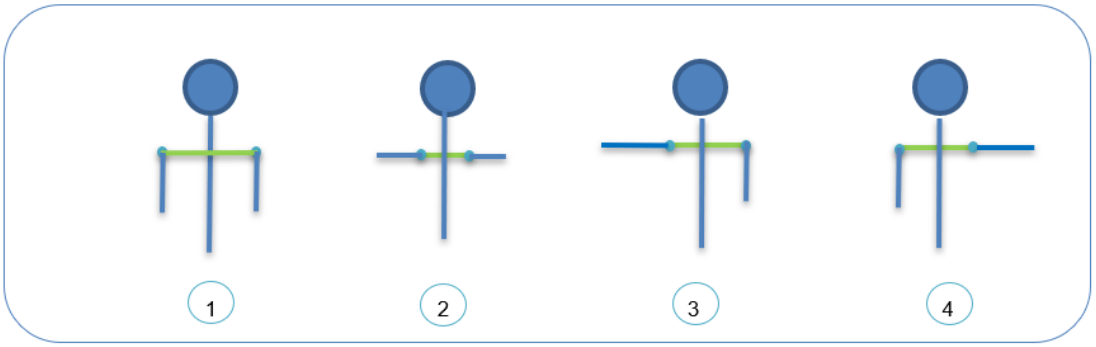
Test findings and summary of Time Delay Test

From the moment Peer1 data is sent from the client to the server until the return from the server to the client to Peer1 and Peer2 takes less than a second to about half a second.

The time difference between Peer1 and Peer2 on the client side is between 100-400 milliseconds (up to about half a second)

3. **Checking the accuracy of synchronization algorithm data**: During the development process we performed validation of the main algorithm by the following means:
    - ❖ Split estimation procedure by areas such as upper body part area according to right and left side and lower body part area according to right and left side.
    - ❖ The process of changing the algorithm settings such as changing the estimation of rotation angles in order to improve the accuracy of the matching result.
    - ❖ A synchronization meter was evaluated according to repeated tests of the synchronization algorithm and end-case tests against different human positions.

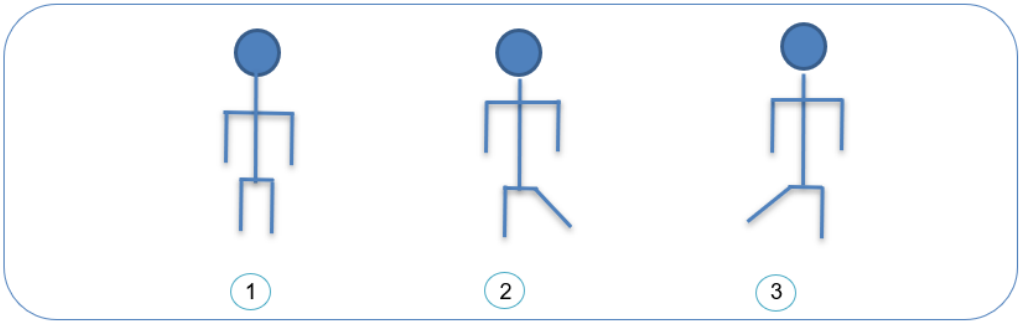
TESTS UPPER AREA (HANDS):



Case	Test-1	Test-2	Test-3	Test-4	AVG
1+2	0.0 :Left 0 :right **0** :Both	0.12 :Left 0 :right **.0** :Both	0.07 :Left 0 :right **.0.03** :Both	0.09 :Left 0.06 :right 0.07Both	0.04
1+3	0 :Left 0.96 :right **0.48** :Both	0.09 :Left 0.974 :right **0.535** :Both	0.12 :Left 0.96 :right **0.54** :Both	0.07 :Left 0.926 :right **0.49** :Both	0.5
1+4	0.92 : Left 0.11 :right **0.51** :Both	0.980 :Left 0.01 :right **0.498** :Both	0.89 :Left 0 :right **0.44** :Both	0.873 :Left 0.045 :right **0.45** :Both	0.45
2+3	0.891 :Left 0 :right **0.445** :Both	0.94 :Left 0.10 :right **0.524** :Both	0.88 :Left 0.013 :right **450.** :Both	0.96 :Left 0.019 :right **0.494** :Both	0.47

Figure 8-2: synchronization algorithm accuracy testing - upper area

TESTS BOTTOM AREA (LEGS):



Case	Area	Test-1	Test-2	AVG

1+2	All body area
	
	Button 20%
	Upper 80%
		:Upper hands
		}
		0.949 :left
		0.95 :right
		{
		:Button legs
		}
		0.0215 :left
		0:right
		{
		:both
		0.7676
			:Upper hands
			}
			0.965 :left
			0.94 :right
			{
			:Button legs
			}
			0 :left
			0.044 :right
			{
			:both
			0.7627
				0.76

1+3		:Upper hands
		}
		0.996 :left
		0 :right
		{
		:Button legs
		}
		0.970 :left
		0.963 :right
		{
		:both
		0.592
			:Upper hands
			}
			0.866 :left
			0.94 :right
			{
			:Button legs
			}
			0.850 :left
			0.971 :right
			{
			:both
			0.9062
				0.70

2+3		:Upper hands
		}
		0.9523 :left
		0.4761 :right
		{
		:Button legs
		}
		0 :left
		0:right
		{
		:both
		0.38
			:Upper hands
			}
			0.84 :left
			0.98 :right
			{
			:Button legs
			}
			0.048 :left
			0.044 :right
			{
			:both
			0.743
				0.53

Figure 8-2: synchronization algorithm accuracy testing - bottom & all body area

4. **Voice recognition system test**: For user convenience, the system supports identifying basic commands when meeting participants, repeated tests to identify keywords and the accuracy of the recognition by adding words similar to the keyword identified by the voice recognition library, tested for their effect in the system for both participants Simultaneously during a real-time connection session
5. **Product Verification**: Suspended video in front of the customer after development on the full functionality required for the system, all of whose requirements were met in the system and properly defined. We received feedback and corrections in VIEW made in accordance with the comments.

   Displaying error messages is appropriate for the following user:

   - Communication error or failure while connecting to the meeting.
   - Error or communication failure in disconnecting a user while making a meeting.
   - Entering an incorrect email address or password while logging in to the system.
   - Entering an incorrect email address while logging in to the system.
   - Email address There is no system for granting administrator privileges to the user registered.
   - Failure of an attempt to make a meeting of an adult user in the system.
   - Server failure when uploading a video after the end of the sports meeting
   - Image/video not uploaded complete or failed

   Displaying confirmation messages for performing the following actions:

   - Successful completion of a completed meeting
   - Successful identification of a single person when connecting using the MediaPipe library.
   - Successful upload of photos or video.
   - Successful addition of additional users to the system by a trainer.
   - Successful addition of sports meetings by a trainer

9 SUMMARY, EVALUATION, CONCLUSIONS AND FUTURE WORK

9.1 Summary

The SyncAlong system for managing and analyzing joint and synchronized sports sessions of two users (a trainer user and an adult user), collects real-time human evaluation data and calculates movement synchronization for each frame along a segment of a sports exercise lasting 30 seconds and displays the synchronization meter according to an augmented reality display to the user's screen.

As part of the project, an innovative algorithm was developed and implemented to detect motion synchronization in real time. To the best of our knowledge, this is the first such algorithm to detect synchronization of people in an online real-time meeting.

9.2 Evaluation and conclusions

The SyncAlong system was developed from the idea and need of the Kader Center, the Center for Innovation and Technology. The system is able to detect, calculate synchronization, put feedback on the screen and record the session.

As part of the project, algorithmic components were developed for the calculation of synchronization based on transformation and change based on the collection data of human posture assessment and the use of *Procrustes analysis* for the imagination of the participants, and the use of *Fréchet distance* for the imagination of the synchronization between two objects that underwent a transformation matching imagination.

This is the first project that provides a response to movement synchronization between people in a real-time activity. So far, no system has been found that enables the detection of synchronization between participants in real time within the framework of a joint sports meeting.

9.3 Future work

There are several issues which the current version of SyncAlong does not deal with and that should be addressed in further developed:

- **Improving the accuracy of the synchronization calculation component:**

  The synchronization algorithm was defined based on repeated attempts of meetings during the development and the assumptions were accepted according to the understanding of the subjects in the meeting (by Shani and Ayman), when using additional participants and receiving a test from them it will be possible to change the synchronization indicators according to the understanding of the rule, which will result in improving user experience and synchronization accuracy.
- **Expansion of the system to more than two participants in a sports meeting**: as of this time, the system allows a joint meeting that includes only two participants and the synchronization is done user against user.
- **Handling a reliable communication hug**: changing the use of the webRTC API in direct use and without a wrapper library with an external server that may delay/prevent the connection request.
- **Improving augmented reality to 3D.**

10 REFERENCES

[1] Rinott, M., & Tractinsky, N. (2021). Designing for interpersonal motor synchronization. Human–Computer Interaction, 1-48..c

[**2**] Human Pose Comparison and Action Scoring

https://medium.com/analytics-vidhya/human-pose-comparison-and-action-scoring-using-deep-learning-opencv-python-c2bdf0ddecba

[3] Move Mirror
https://blog.google/technology/ai/move-mirror-you-move-and-80000-images-move-you/

[4] Move Mirror: An AI Experiment with Pose Estimation in the Browser using TensorFlow.js

https://medium.com/tensorflow/move-mirror-an-ai-experiment-with-pose-estimation-in-the-browser-using-tensorflow-js-2f7b769f9b23

[**5**] Single Pose Comparison
https://becominghuman.ai/single-pose-comparison-a-fun-application-using-human-pose-estimation-part-2-4fd16a8bf0d3

[**6**] A WebRTC based Efficient *https://arxiv.org/pdf/2105.07558.pdf*

[7] Comparison of Human poses with PoseNet
https://medium.com/@priyaanka.garg/comparison-of-human-poses-with-posenet-e9ffc36b7427

[8] Machine Learning :: Cosine Similarity for Vector Space Models
https://blog.christianperone.com/2013/09/machine-learning-cosine-similarity-for-vector-space-models-part-iii/

[9] Procrustes Analysis Based Handwriting Recognition
https://thottingal.in/blog/2020/02/29/procrustes-analysis-based-handwriting-recognition/

[10] H. Alt and M. Godau. Measuring the resemblance of polygonal curves. In Proc. 8th Annu. ACM Sympos. Comput. Geom., pages 102–109, 1992

[11] T. Eiter and H. Mannila. Theory Distance and Similarity: Measures and Computation. Technical Report CD-TR 94/60, CD-Laboratory for Expert Systems, TU Vienna, Austria, February 1994.

11 APPENDIX A: Background material

Synchronization is widely observed in human communication and is considered to be important in generating empathy during face-to-face communication. Synchronization of non-verbal behaviors has psychologically positive effects.

Although the benefits of increased physical activity and exercise are universally recognized, many older persons remain sedentary, and relatively few achieve recommended levels of activity. Effective interventions to reverse the lack of physical activity in older adults are clearly needed.

The SyncAlong system is designed to allow elderlies who are unable to participate in physical activity the way to perform activities in a positive way through synchronized and positive activities by measurement of synchronization movement in a joint activity of two participants.

תקציר

סנכרון נצפה באופן נרחב בתקשורת האנושית ונחשב כחשוב ביצירת אמפתיה במהלך תקשורת פנים אל פנים. לסנכרון של התנהגויות לא מילוליות יש השפעות חיוביות מבחינה פסיכולוגית. למרות שהיתרונות של ביצוע פעילות גופנית מוכרים בכל העולם, אנשים מבוגרים רבים נשארים בישיבה ומעטים מגיעים לרמות פעילות גופנית מומלצות.

יש צורך בבירור בהתערבות טכנולוגית יעילה על מנת לשפר את חוסר המוטיבציה לבצע פעילות גופנית בקרב מבוגרים.

מערכת SyncAlong נועדה לאפשר לאנשים מבוגרים שאינם מסוגלים להשתתף בפעילות גופנית את הדרך לבצע פעילויות בצורה חיובית וליצור סביבה של פעולות גופניות משותפת ומסונכרנת, באמצעות תמיכה בתקשורת בזמן אמת בין שני משתתפים peer-to-peer לחיבור מפגש ספורטיבי משותף הכולל זיהוי תנוחות האדם בזמן אמת , חישוב סינכרון תנועה של שני משתתפים וקבלת פידבק חיובי בעת זיהוי סינכרון תנועה באמצעות מציאות רבודה

הכרת תודה

ברצוננו להודות למנחים שלנו, מר אייל נוסבאום וד"ר מרסלו שייכמן, שהנחו אותנו בפרויקט וליוו אותנו לאורכו, על ההכוונה, העצות וכיווני החשיבה יוצאי הדופן שסייעו לנו בדרך ותרם רבות להצלחת הפרויקט.
כמו כן, אנו רוצים להודות למרכז קדר, למיכל רינות ושחר גיגר, שנתנו את העצות והמיקוד שהיינו צריכים, הדלת הפתוחה, העידוד והתמיכה.

לסיום, אך לא פחות חשוב, אנו רוצים להודות ליועץ ה-UI שלנו, רונן קרטיס, שעזר ונתן לנו פרספקטיבה רחבה על UI ונתן נקודות מבט מקצועית בפרויקט ששיפרו את המוצר הסופי.

SYNC ALONG

הנדסת תוכנה

B.Sc. ספר פרויקט גמר לתואר ראשון בוגר במדעים

מאת: שני לוי ואימן אלוהבאני
מנחה:ד"ר מרסלו שייכמן ומר אייל נוסבאום

25/08/2022

מוגש כחלק מהדרישות לקבלת תואר ראשון בוגר במדעים (B.Sc)