Politecnico di Torino

Politecnico di Torino

Masterof Science in Computer Engineering

A.Y 2023/2024

Graduation Session OCTOBER 2024

Advancements in LC-Neurorehab

A Neurorehabilitation Tool to Support Professionals in Long COVID Therapy and Patients Evaluation

Supervisors

Prof. Luigi DE RUSSIS

Prof. Vito DE FEO

Candidate

Michele TATTI

Abstract

The widespread impact of Long COVID has driven significant research focused on understanding its causes and recovery processes, highlighting the urgent need for effective neurorehabilitation methods while also facilitating continuous data collection.

The application "LC Neurorehab", developed as part of the research at the University of Essex, addresses this need by offering a comprehensive platform for data collection and analysis. This thesis provides an in-depth overview of the development process of LC Neurorehab, detailing the modifications made to its previous version to better align with the current literature. The database system underwent a complete architectural redesign to optimise data collection processes, ensuring more robust and scalable handling of patient data, facilitating more detailed tracking and analysis of cognitive and behavioural outcomes.

Key enhancements include the introduction of a cognitive questionnaire, integration of EEG synchronisation with the existing behavioural tasks, and improved accessibility for healthcare professionals. These features enable practitioners to more accurately identify and assess patients' cognitive performance, allowing for a deeper understanding of the neurological implications of COVID-19.

The LC Neurorehab platform represents a significant advancement in both neurorehabilitation and the ongoing research into Long COVID.

To my family and friends.
"No one achieves anything alone."

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

The World Health Organization (WHO) has declared the end of the pandemic phase of COVID-19 on May 5, 2023. Even though the COVID-19 situation seems to have stabilized, its impact on social and economic life, as well as the persistence of some symptoms in some individuals, still requires adequate attention.

More than 24 thousand researches have been conducted to study Long Covid and its impact, making it one of the most studied health conditions on human history.

Despite these efforts, there is a significant lack of tools supporting professionals in the rehabilitation and collection of data. Studies, like the one in collaboration with the university of Essex, are trying to drive progresses in this field.

Aim of the project

The LC-Neurorehab project focuses on developing a comprehensive tool aimed at supporting neurorehabilitation, particularly for individuals dealing with long Covid. The core objective is to enhance patients' working memory and improve their attentional control by engaging them in tasks that are well-established in psychological and neurological research for their cognitive benefits while simultaneously collecting data to support the research in this field.

By assessing the patient's cognitive state upon first login, and continuously storing data as they engage with the tasks there is a systematic data collection that tracks the progression of a patient condition over time, offering valuable insights into the cognitive effects of long Covid. This knowledge can significantly contribute to a better understanding of the condition and help in the development of more effective treatment strategies.

Organization of the Thesis

The thesis is structured in different chapters:

- **Chapter 2** provides a context for Long Covid, the impact that it had on the life of different people and the neurological consequences.
- **Chapter 3** gives an overview of LC-Neurorehab, the functionalities it provided before the changes made in this thesis.
- **Chapter 4** details the design of the solution, the gathered requirements and the changes made to the application.
- **Chapter 5** describes the technology and the tools used in the work carried during this thesis.
- **Chapter 6** highlights the obtained results in different areas of the application.
- **Chapter 7** ends the thesis presenting the conclusions and possible future developments.

Chapter 2

Context and origin

2.1 Covid-19

In December 2019, the first cases of a novel infectious disease associated with pneumonia and acute respiratory distress were reported in the city of Wuhan, Hubei Province China.

It was quickly identified as SARS-CoV-2 which is a member of a large family of viruses called coronaviruses. SARS-CoV-2 is a single-stranded, positive-sense RNA virus closely related to the SARS and MERS viruses. It primarily infects the respiratory system by binding to the ACE2 receptors found on the surface of various cells, including those in the lungs. [1]

Once inside the body, the virus replicates, leading to widespread inflammation and potential damage to organs. The virus spreads from person to person through droplets released when an infected person coughs, sneezes, or talks. It may also be spread by touching a surface contaminated by the virus and then touching one's mouth, nose, or eyes, but this is considered to be less effective. [2]

Due to its rapid spread and the significant international concern it generated, the outbreak of the coronavirus was officially declared a pandemic on March 11, 2020. As of August 11 2024, the World Health Organization (WHO) reports that the cumulative total of global Covid-19 cases has reached 775 million, with more than 7 million reported deaths.

2.2 Symptomatology

The most common symptoms of Covid-19 are fever, often accompanied by a dry cough and fatigue. Studies indicate that 98% of hospitalised Covid-19 patients experience fever [3], while other respiratory symptoms might include nasal congestion, sore throat, and a loss of smell or taste. Beyond respiratory symptoms, Covid-19 can

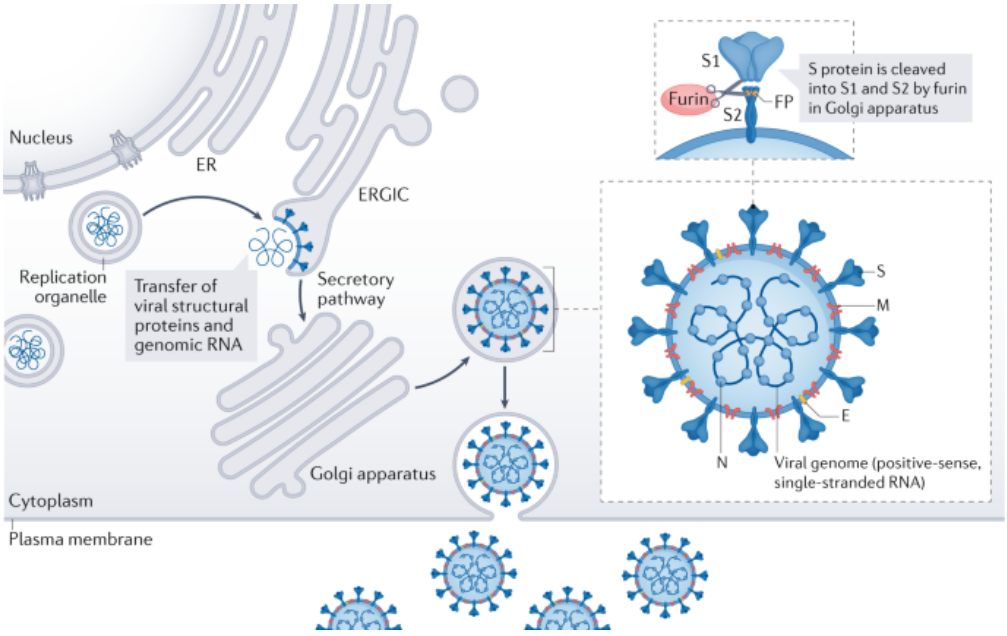

Figure 2.1: Coronavirus structure and maturation [1]

lead to severe pulmonary conditions like pneumonia and acute respiratory distress syndrome (ARDS), often necessitating ventilatory support and intensive care. [4] Covid-19 was primarily viewed as a pulmonary disease with some extra-pulmonary manifestations. However, over time, it became clear that the virus affected the entire body: in addition to respiratory symptoms, common symptoms include gastrointestinal issues and fever. Depending on the severity of the case, patients may also experience complications involving the kidneys, venous thromboembolism (VTE), heart, or central nervous system. Skin and ocular manifestations have also been reported. [4] As time progressed it was discovered that in some patients the symptoms lasted longer than expected with some developing neurologic symptoms after clearance of SARS-CoV-2 infection.

2.3 Long Covid

It is estimated that up to 25% of patients who have recovered from SARS-CoV-2, [5] regardless of age or severity of original symptom will experience persistent symptoms. [6]

This condition known as Long Covid, or Post Covid-19, is defined as the continuation or development of new symptoms 3 months after the initial SARS-CoV-2 infection, with these symptoms lasting for at least 2 months with no other explanation. [4]

Common symptoms of long Covid can include fatigue, shortness of breath and cognitive dysfunction. The list of complete symptoms shown in table 2.1 makes evident that Covid-19 has had a multi-domain impact on the human condition. [5]

Numerous studies are now focusing on understanding the effects on cognitive functions and potential recovery strategies. Epidemiological research indicates that approximately 20% of Covid-19 survivors experience objective cognitive impairment, with assessments typically based on standardised neuropsychological tests. [7] Studies have further demonstrated that individuals who have recovered from Covid-19, exhibited significant cognitive deficits versus controls group when controlling for age, gender, education level, income, racial-ethnic group, pre-existing medical disorders, tiredness, depression and anxiety. [8] Although it was expected for people that have been hospitalised, it is still unclear how people who only developed a milder case can still suffer from Long Covid.

Neurological Sequele	Symptoms and Presentation
Fatigue	Physical, mental, or emotional energy deficit that worsens after physical or mental exertion
Neuropsychiatric	Anxiety, post-traumatic stress disorder, pain disorder, delirium, mood swings, psychosis
Sleep disturbances	Insomnia, low sleep efficiency, nightmares, lucid dreaming
Sensorimotor deficits	Peripheral neuropathy, paresthesias, neuropathic pain, myalgia, persistent weakness
Brain Fog	Poor concentration, slowed thinking, difficulty paying attention, and focusing
Hyposmia/parosmia	Partial or total loss of sense of smell/misperceiving odors (often pleasant odors seem unpleasant)
Hypogeusia/dysgeusia	Partial or total loss of sense of taste/altered perception of taste
Hearing problems	Hearing loss, tinnitus
Ocular symptoms	Tearing, hyperemia, chemosis (conjunctival swelling), conjunctivitis, damage to ocular nerves

Table 2.1: Neurologic manifestations of long COVID and associated symptoms. [5]

Given the significant number of individuals affected by Long Covid and the evident lack of effective neurorehabilitation methods to support them, it is imperative to conduct further studies to assess the full extent of the impact and to develop strategies for recovery. Such research is crucial not only for improving the quality of life for those affected but also for ensuring the healthcare system is adequately equipped to address the long-term consequences of the pandemic.

Chapter 3

LC-Neurorehab

LC-Neurorehab is a desktop application created as part of the research made by the University of Essex and individual thesis contributions. It is thought as a cognitive training system to be used in a laboratory environment under the supervision of experts in the field of medicine and psychology for Long Covid neurorehabilitation.

Figure 3.1: LC-Neurorehab logo

It is centred around two tasks well-established in the literature for their usefulness on cognitive development, whilst also supporting psychologists in the assessment and evaluation of patients cognitive performance. Through the application professional can evaluate the data gathered during the sessions with the patients. The following paragraphs will present an overview of the application as it was prior to the changes introduced by this thesis work.

3.1 Log In and Registration

Upon lunching the application, a user is presented with a log in page. From this page, if already registered, the user can log in and access the homepage where a list of all the available tasks will be presented, start the registration process or access the admin area if registered as admin.

The registration process asks users to input personal information such as the first and last name and the date of birth. It also requires them to choose username and password. The username must be unique as it identifies a specific user.

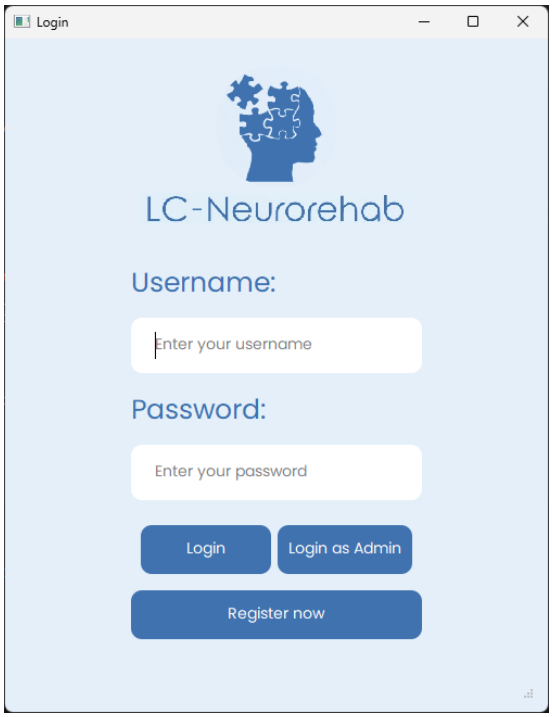

Figure 3.2: Log In display

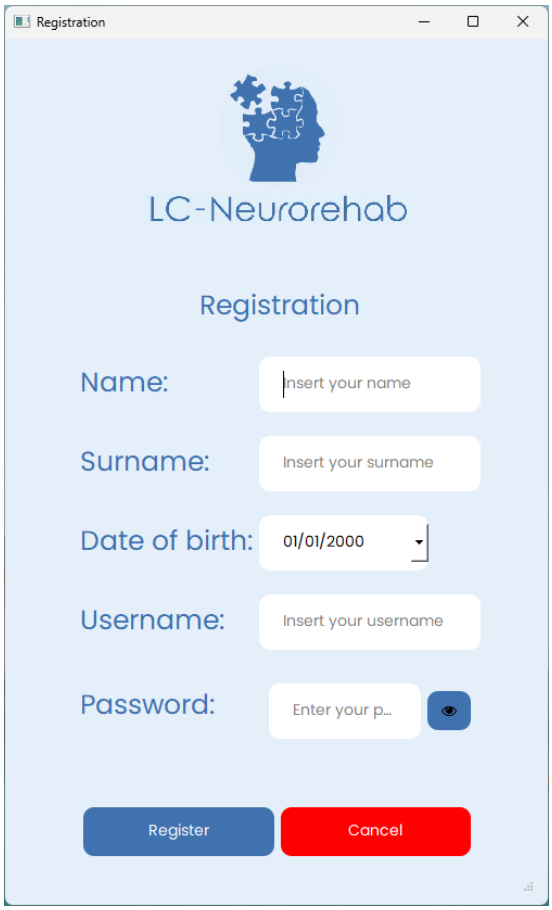

Figure 3.3: Registration display

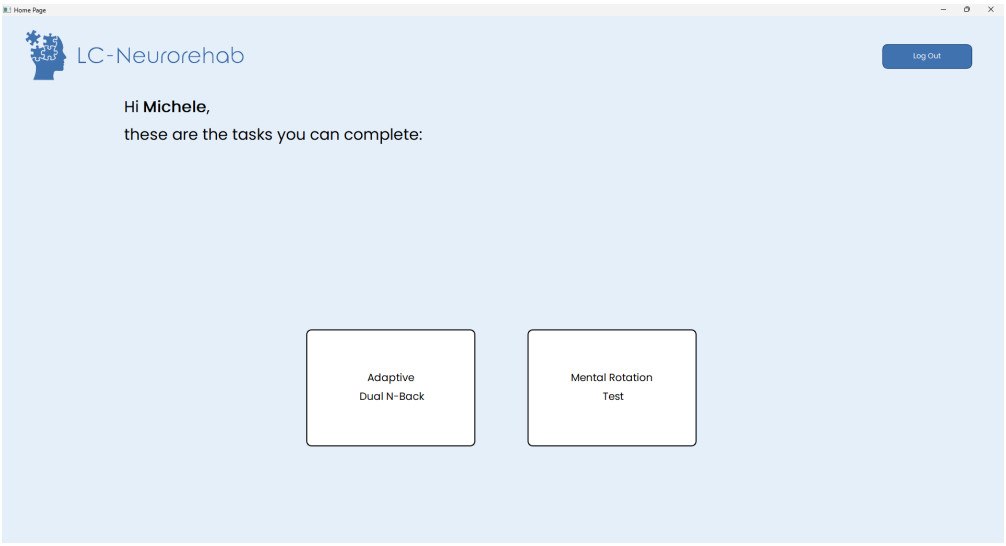

Figure 3.4: Homepage display

3.2 Tasks

3.2.1 Adaptive Dual N-Back

The Dual N-back task involves a sequence of trials presented at intervals of 2500 msec, each characterised by two distinct stimuli —hence, the term *dual*— with a 500 msec delay between the two. The two types of stimuli are visuospatial and auditory-verbal. [9]

The visuospatial stimulus consists of a blue square that appears in one of the external cells of a 3x3 grid, while the auditory-verbal stimulus is a random spoken letter from the English alphabet. The participant is required to process these two streams of information independently by comparing the current trial with the one that occurred n trials before.

If the participant detects a match, he must press the corresponding button for either the visual or auditory match, or both if both stimuli match. The task begins at level 1, where the participant has to compare the current trial with the immediately preceding one, the maximum level they can reach is 4.

The task is structured into 20 blocks, with each block consisting of 20+n trials, where n corresponds to the current difficulty level.

This adaptive nature of the task ensures that the difficulty adjusts based on the participant's performance at the end of each block, optimising cognitive engagement while minimising frustration. The highest level of difficulty will be compared with 4 trials back. The performance is checked against these accuracy levels:

- less or equal than 75% it will go to the previous level or stay in the same level on level one.
- between 75% and 95% included it will stay on the same level of difficulty.
- above 95% it will increase the difficulty of one level.

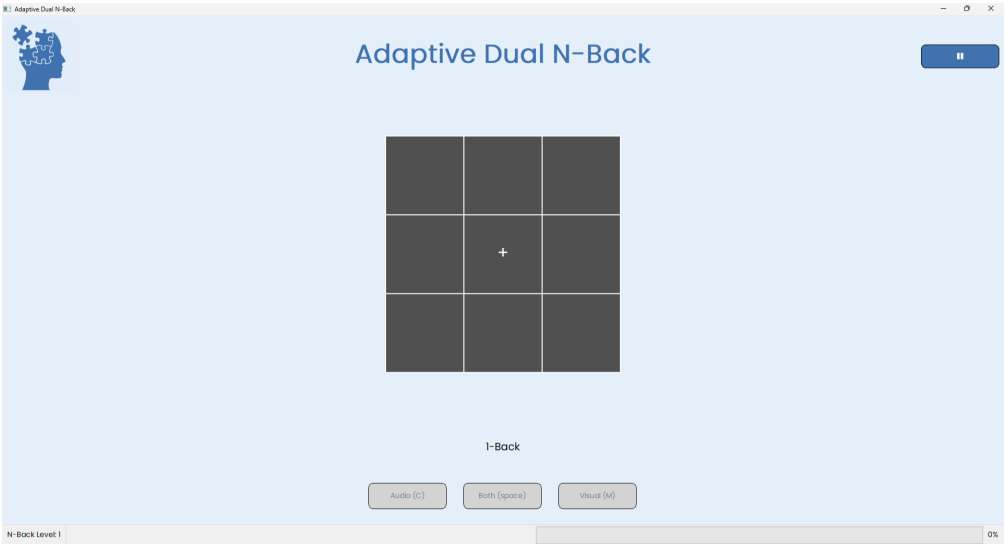

Figure 3.5: Adaptive Dual N-Back

Cognitive Benefits of Dual N-Back

Regular practice with the Adaptive Dual N-Back task enhances working memory, specifically the capacity to retain and manipulate information in short-term memory. These improvements translate into better attentional control, which is the capacity of focusing on relevant tasks while disregarding distractions. In addition research has shown that the benefits gained from training with the Adaptive Dual N-Back task can extend beyond the specific skills practised during the exercise and can be transferred to other cognitive activities that are not directly targeted by the task thus proving the wide range of application that dual n back has from cognitive rehabilitation to performance optimization in healthy individuals. [9]

3.2.2 Mental rotation test

The Mental Rotation Test (MRT) is a cognitive test used to assess an individual's ability to mentally visualise and manipulate objects in three-dimensional space.

It is used to measure spatial reasoning, which is the ability to understand and remember the spatial relations among objects.

It is one of the most influential paradigms in the history of cognitive psychology [10]. The participant is presented with two 3D objects: a baseline object that will be shown on the left and a target object that can be the same image or a mirrored image but rotated in its vertical and/or horizontal axes by 0, 50, 100, and 150 degrees.

Participants are asked to mentally rotate the target object to determine whether it can be aligned with the baseline object by rotation and they must press one of two buttons:

- **Same** when the rotated target object coincides with the baseline.
- **Different** when the rotated target object does not coincides with the baseline.

They have 7.5 seconds to detect it, otherwise the answer will be considered incorrect and they will move on to another set of two images. The images are taken from the repository given in the research paper [10].

To keep the cognitive engagement high, at the beginning of the task the participant's ability will be first assessed, with a block of 16 sets of images with a difficulty level going from 1 to 4.

Depending on the accuracy obtained the task will move to one of the following levels:

- level 1: accuracy less or equal than 62.5 %.
- level 2: accuracy 62.5 % to 75 % included.
- level 3: accuracy from 75 % to 87.2 % included.
- level 4: accuracy above 87.5 %.

After the initial assessment every 3 minutes the accuracy will be checked and the difficulty adjusted accordingly to the following:

- less or equal than 75 % it will go to the previous level or stay in the same level if on level one.
- between 75 % and 9 5% included it will stay on the same level of difficulty.
- above 95 % it will increase the difficulty up until level 4.

In total, the Mental Rotation Test takes 12 minutes.

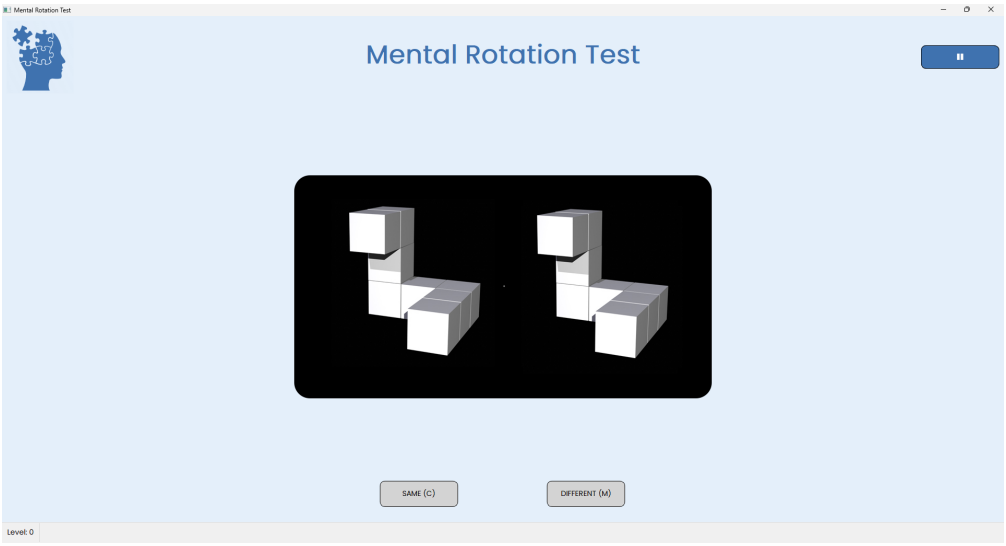

Figure 3.6: Mental Rotation Test

Cognitive Benefits of Mental Rotation Test

The Mental Rotation Test (MRT) is commonly used as a measure of mental rotation, a cognitive process involving covert simulation of motor rotation. [11] However, research indicates that MRT performance is not solely reliant on motor simulation; it also engages other analytic cognitive strategies that depend on visuospatial representation and visual working memory (WM). [11]

This suggests that MRT performance involves a flexible use of cognitive strategies, particularly balancing between a motor simulation-based approach and WM-intensive analytic strategies, depending on the task's difficulty resulting in great value for the neurorehabilitation of patients and can yield long term cognitive advantages in different contexts.

3.3 Admin Area and User Page

The admin area of the application allows psychologists to monitor patients progress, manage their tasks, and have an administrative view on the application.

In the Admin Area a list of users is shown. By clicking on *Show* the specific user page is displayed. It contains the information about the user, some notes that the professional might have left and from there one can see the progress of the patient on a specific task.

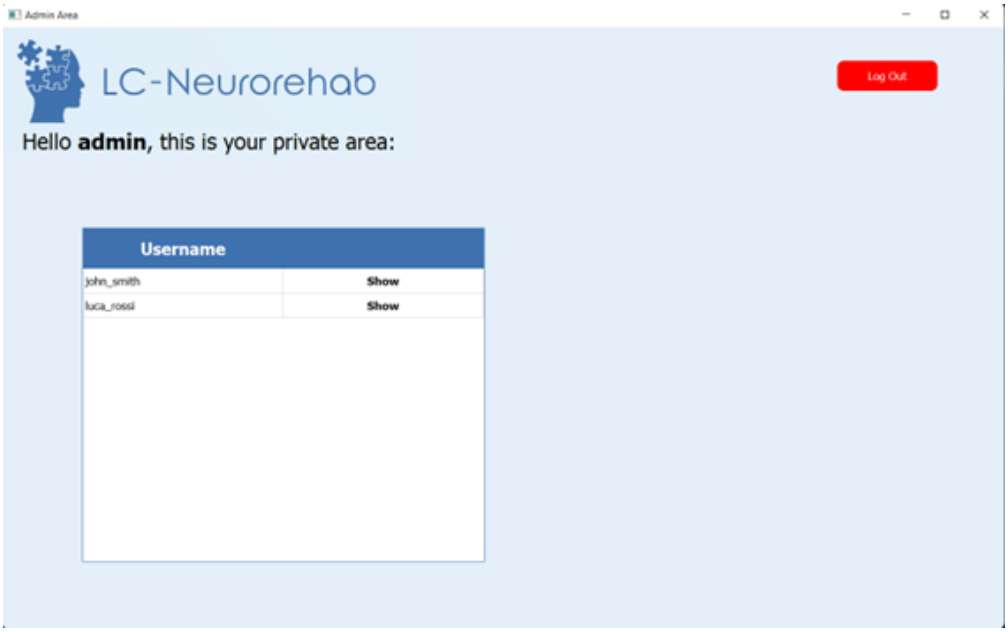

Figure 3.7: Admin Area

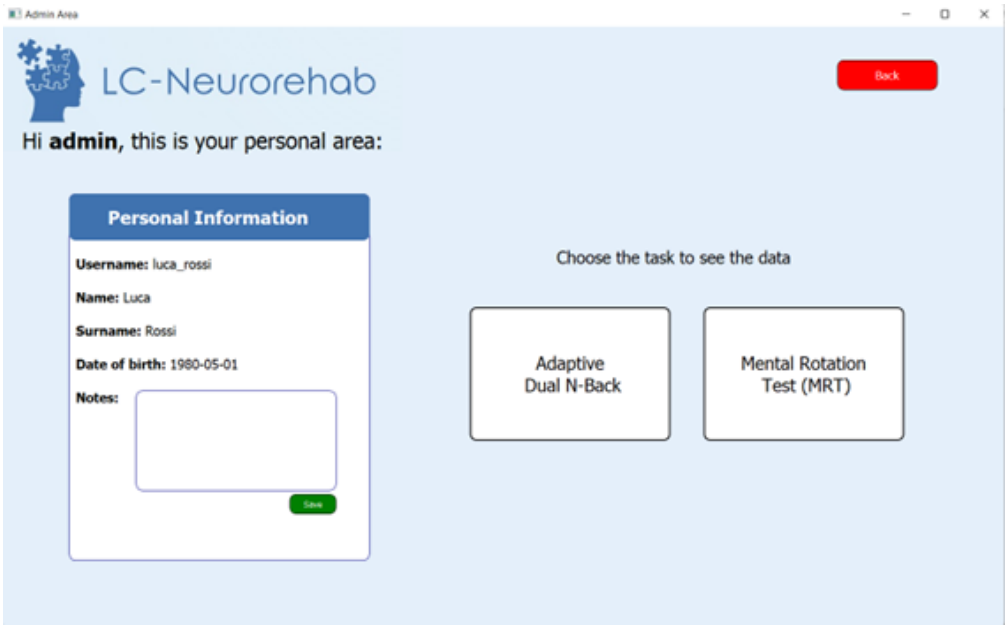

Figure 3.8: User Page

Chapter 4

Solution Design

The work conducted in this thesis aims to improve the existing application from which this project builds upon and to develop new features. While some improvements had already been planned at the time I started working on it, others still required clarification.

Continuous collaboration with psychologists and experts, combined with some feedback obtained during the application showcase to the public and a rigorous study on the state of the art of the application led to further requirements gathering, a thorough plan of what needed to be implemented and its design.

The lack of proper separation of the architectural layers, which resulted in tightly coupled code that was difficult to understand and maintain, along with the complexity introduced by new requirements, called for the adoption of a defined architectural pattern.

The redesign of the codebase has been done by keeping in mind the core aspects of the application. It needed: to operate as a standalone system within a controlled clinical environment, to process data locally to minimize risks and ensure data security, to be flexible enough to accommodate new tasks and allow customization of the rehabilitation protocol.

This chapter will describe the gathered requirements, the choices made in the design of the application architecture and the database.

4.1 Requirements

When it comes to defining the requirements, we need to differentiate them into two categories: *Functional* and *Non-Functional*.

Functional requirements refer to the specific features, functionalities that the application must own, they describe what the app should do and how it should perform certain tasks.

On the other hand, *Non-Functional* requirements specify the qualities or characteristics that a software should possess and ensure certain standards and user expectations are met in terms of performance, security, usability, scalability, reliability and maintainability.

4.1.1 Functional Requirements

The collaboration with experts in the field of cognitive development and Long Covid and their feedback had a substantial value in the definition of the functional requirements reported in the table 4.1.

Table 4.1: List of Functional Requirements.

Identifier	Description
FR-1	Admin must be registered to the app.
FR-2	The application must let participants complete a daily questionnaire only once per day of login.
FR-3	The application must allow the participant to pause the task they are undertaking.
FR-4	The application must allow participants to resume the task they have paused.
FR-5	The application shall require participants to answer to a questionnaire at the end of each block.
FR-6	The system is required to store trial data.
FR-7	The system is required to store/update block data together with end of block questionnaire data.
FR-8	The system is required to store daily questionnaire data.
FR-9	The application must provide a tutorial for the Dual N-Back task.
FR-10	The Dual N-Back tutorial must show hints to guide the user.
FR-11	The application must provide examples for MRT.
FR-12	The system must allow the participant to go back to the homepage if the task is paused.
FR-13	The system must allow participants to restart a task from the start of the block where they stopped.
FR-14	The application will allow patients to pause/resume the task through specific keys on the keyboard.

FR-15	The system must send a signal to the EEG when the task starts, a stimulus is presented or the user responds to the task.
FR-16	The application must store for each task a session number to discriminate different sessions in a day.
FR-17	The system must allow admins to download users' questionnaire data in csv format .
FR-18	The system must allow admins to download users' trial data in csv format.
FR-19	The system must allow admins to download users' block data in csv format.
FR-20	The system must allow admins to modify settings related to the task.
FR-21	The system must allow admins to modify inter trial time and inter stimulus time for each user.
FR-22	In the User Page the system must allow admins to see the list of dates when the user took daily questionnaire.
FR-23	In the User Page the system must allow admins to download a specific daily questionnaire taken by a specific user.
FR-24	In the User Page the system must allow admins to download all daily questionnaires taken by a specific user.
FR-25	In the User Page admins can choose to display information for a specific task by selecting the task.
FR-26	The system will show data related to a single block.
FR-27	The system must allow admins to download data related to the blocks.
FR-28	The system must allow admins to download data related to trials.
FR-29	The system will store admin latest login or modification.
FR-30	The system will show data related to a session.
FR-31	Questionnaire will show if a question is not answered or not by coloring the square around it.

4.1.2 Non functional Requirements

Non-functional requirements are the criteria that define how a system should behave, rather than what it is supposed to do. These requirements focus on aspects such as performance, security, usability, reliability, and scalability.

While many of the *non-functional* requirements were already addressed in the previous version of the application, two critical areas were identified that required additional attention:

- Maintainability: it refers to the ease and speed with which a system can be updated, fixed, or modified to improve performance or adapt to changes in the environment.
- Responsiveness: it ensures that the application's user interface works seamlessly across different device sizes. Features should be available in all viewports, but their behaviour may adapt to different screen sizes.

The first requirement implied the redesign of the architecture, as the previous one had tightly coupled layers and lacked proper structure, making it difficult to modify or extend the system. The second requirement was already identified before the beginning of this work since the previous version of the application used fixed positions for widgets in the graphical user interface. This restriction prevented the application from being displayed properly in full-screen mode or across various screen sizes.

4.2 Architecture Design

With the introduction of new requirements, the increasing complexity of the application and the lack of clear separation of the three architectural layers, it became essential to redesign the architecture of the system. The redesign focused on establishing a clear division of responsibilities between layers by adapting a more modular approach, thus improving maintainability and flexibility while ensuring clarity within the codebase.

To achieve this goal, the Model-View-Controller architectural pattern was adopted. This facilitates the separation of the responsibility by dividing the system into distinct components, and ensures the independence among them as each one is responsible for solving a specific problem while contributing to the overall coherence of the application.

4.2.1 Model-View-Controller

Model-View-Controller (MVC) is a widely used software architectural pattern that separates an application into three interconnected components each with its own

distinct responsibility:

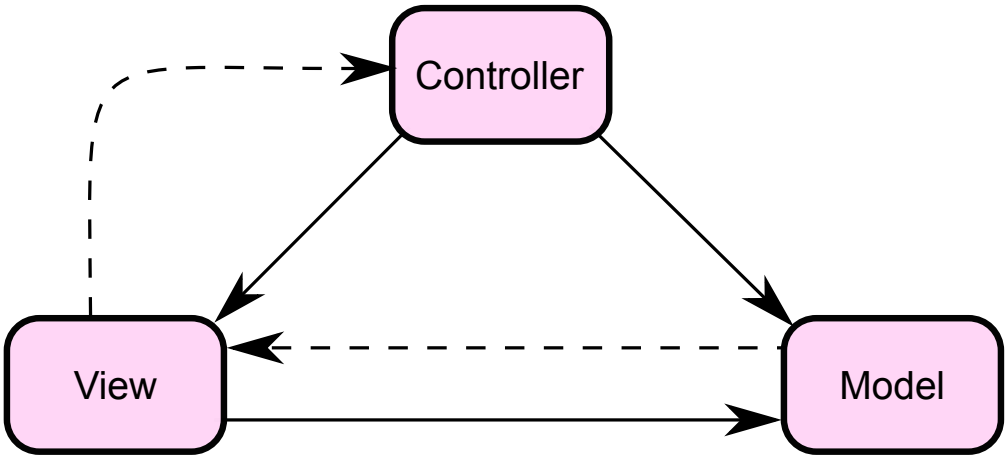

Figure 4.1: Representation of the Model-View-Controller

1. The **Model**: it defines what data the application should contain. If the state of this data changes, then the model will usually notify the view (so the display can change as needed) and sometimes the controller (if different logic is needed to control the updated view) [12].
2. The **View**: it is a visual representation of the model, it does not modify it but only reads its content. This is crucial to ensure that the application is user-friendly and accessible, making it a distinctive component focused on user experience and interaction.
3. The **Controller**: it contains the logic that updates the model and the view in response to the intercepted input from the users of the app. It acts as an intermediary between the view and the model. It extracts the parameters and invokes the service layer and chooses the view that will be populated with data obtained from the model. It is the core part of the interaction between view and model.

This choice brings different benefits to the application:

- **Separation of concerns**: Each component has a distinct responsibility, making the code easier to understand and maintain.
- **Modularity**: It is easier to modify or update one part of the application (e.g., changing how data is presented in the View) without affecting the others.
- **Testability**: Since components are separated, individual parts can be tested in isolation, improving the quality of the software.

- **Scalability**: The modular nature of MVC allows for easier scaling and expansion of the application.

4.2.2 Data access layer

The Model represents the data, but its structure does not directly correspond to how the database is organised. To manage interactions with the database, a new separate class needs to be developed.

This class will be responsible for handling all methods that interact with SQL, acting as a bridge, between the in-memory objects and the database. By separating the in-memory objects from the database, this class isolates them from each other, ensuring that the rest of the application remains unaffected by changes in the data storage layer.

All SQL queries and database access being centralized in this class will allow for a very seamless change if the underlying data storage technology changes in the future, as it will be the only class to be updated.

4.3 Database design

In this section, we will outline the changes made to the database and compare them to the previous structure (Table 4.2). After gathering all the new requirements, it became evident that the database structure needed to be modified to accommodate them. As a result, new tables were added and existing one were updated to hold additional information.

We applied database normalization principles, incorporating primary keys, foreign keys, and composite keys to ensure data integrity and efficient organization. The primary objective of database normalization is to simplify the database design by reducing data redundancy, eliminating anomalies, and structuring data in a systematic manner. This process divides the data into multiple related tables, ensuring that relationships between these tables are maintained through well-defined keys and constraints.

The database has been normalized to **Third Normal Form** (3NF), which involves the following key principles:

1. **First Normal Form (1NF)**: Ensures that each table contains only atomic values (no repeating groups or arrays), and each column contains values of a single type.

2. **Second Normal Form (2NF)**: Ensures that all non-key attributes are fully functionally dependent on the entire primary key. This eliminates partial dependencies in tables with composite primary keys.

3. **Third Normal Form (3NF)**: Ensures that there are no transitive dependencies, meaning that non-key attributes are dependent only on the primary key and not on other non-key attributes.

By applying these normalization rules, the database structure was optimized avoiding redundancy, streamlining data access, and maintaining consistency across the system.

Table 4.2: Representation of old database.

Tables	**Fields**
admins This table contains information about the administrators of the desktop app.	• username • password • salt
users This table contains information about the patients of the desktop app.	• username • password • name • surname • dateofbirth • notes • salt

Tables	Fields
dualnback This table records user performance metrics in an Adaptive Dual N-Back cognitive training task.	• username • level • accuracy • n_correct_audio • n_correct_visual • n_correct_both • score • date
mrt This table records user performance metrics in a Mental Rotation Test cognitive training task.	• username • level • accuracy • n_images • n_correct • n_wrong • date

User related tables

One of the first tables modified was the User table with the addition of these new fields:

- **user_id**
- **is_admin**

The admin table in the old database has been incorporated by the User table with the addition of the field **is_admin**. This ensures that an admin is also a fully registered user before having this role granted.

Table	Fields
User This table contains information about the users of the application.	• user_id • username • password • name • surname • dateofbirth • notes • salt • is_admin

Table 4.3: User table

The other change is the definition of a new integer field: **user_id**. This is also a change on the primary key of table that uniquely identifies the user, since the previous one was *username*, which will still remain a unique entry. The reason behind this choice is that by using integer as a primary key a consistent, immutable identifier for efficient lookups, indexing, and table relationships, is provided while still allowing flexibility in managing usernames that may change over time.

As soon as a new user will be registered a trigger, which is procedural code automatically executed in response to certain events on a particular table, will insert in the UserTimeTask table two records with the user_id, the task and the corresponding inter trial time and inter stimulus time (if present) default values found in the papers that have been used to create the two tasks.

```
1 CREATE TRIGGER IF NOT EXISTS new_user_registered
2 AFTER INSERT ON user
3 BEGIN
4     INSERT INTO UserTaskTime(user_id,task,latest_update,
    inter_trial_time,inter_stimulus_time)
5     values(NEW.user_id,"DualNBack",CURRENT_TIMESTAMP,2500,500);
6     INSERT INTO UserTaskTime(user_id,task,latest_update,
    inter_trial_time)
```

```
7       values(NEW.user_id,"Mrt",CURRENT_TIMESTAMP,7500);
8   END;
```

Table	Fields
UserTimeTask	• user_id • task • latest_update • inter_trial_time • inter_stimulus_time

Table 4.4: This table holds the time settings for each user's task

This table has been added to give the possibility to admins to tailor the timing of the tasks to the participant's needs.

Task related tables

For what it concerns the two main tasks new fields have been added to the Dual N-Back and MRT tables:

- **task_id**: unique identifier for each task session.
- **user_id**: foreign key to link to the User table.
- **session**: value that specifies how many attempts at that task have been made that specific date.
- **block**: a block is made up of tot trials is used to identify the specific block within the session.
- **entertainment, tiredness, difficulty**: 3 key areas in the subjective perception of the task by the user.
- **session_timestamp**: variable date renamed to better convey its value.

As shown in tables 4.5 and 4.6, an auto incremented integer id has been defined as primary key to uniquely identify the block, with the change of the primary key in the User table also the field *username* has been changed to *user_id*.

For more complex test structures with multiple blocks per session two new fields have been introduced: session and block. While the three new fields entertainment, tiredness, difficulty have been added to track the subjective perception of the users.

The new structure allows for more detailed analysis of patients' performance. Considering subjective factors like fatigue, engagement, and perceived difficulty provides an outlook to the context in which the task was undertaken.

DualNBack	
Field	**Description**
`dualnback_id`	This is the primary key.
`user_id`	References the user to which the block data corresponds.
`session`	Indicates the session to which the block belongs.
`level`	Indicates the level of difficulty of the block.
`accuracy`	Indicates the level of accuracy of the user during the block.
`n_correct_audio`	Indicates the number of correct answers for audio stimulus.
`n_correct_visual`	Indicates the number of correct answers for visual stimulus.
`n_correct_both`	Indicates the number of correct answers when both stimuli are presented.
`n_correct_total`	Indicates the total number of correct answers.
`session_timestamp`	Indicates the timestamp of the session.
`block`	Identifies the block number within a session.
`entertainment`	Indicates the level of entertainment of the user.
`tiredness`	Indicates the level of tiredness of the user.
`difficulty`	Indicates the level of perceived difficulty of the user.

Table 4.5: This table records user performance metrics in an Adaptive Dual N-Back cognitive training task.

MRT	
Field	**Description**
`mrt_id`	This is the primary key.
`user_id`	References the user to which the block data corresponds.
`session`	Indicates the session to which the block belongs.
`level`	Indicates the level of difficulty of the block.
`accuracy`	Indicates the level of accuracy of the user.
`n_images`	Indicates the number of sets of images shown during the block.
`n_correct`	Indicates the number of correct answers.
`n_wrong`	Indicates the number of wrong answers.
`isTimeBased`	Indicates whether the task is constructed by block or time.
`session_timestamp`	Indicates the timestamp of the session.
`block`	Identifies the block number within a session.
`entertainment`	Indicates the level of entertainment of the user.
`tiredness`	Indicates the level of tiredness of the user.
`difficulty`	Indicates the level of perceived difficulty of the user.

Table 4.6: This table records user performance metrics in a Mental Rotation Test (MRT) cognitive training task.

Besides the two preexisting task tables, to record each single trial inside a block two more tables have been added (table 4.8 , 4.7).

The tables will store the times that characterize the trial such as response time, inter trial and inter stimulus time, the given answer and the correct answer, facilitating in depth performance analysis.

DualNBackTrial	
Field	**Description**
`dualnback_id`	References the DualNBack block.
`trial_number`	Indicates the number of the trial within the block.
`inter_trial_time`	Indicates the time between each trial.
`inter_stimulus_time`	Indicates the time between each stimulus.
`visual_timestamp`	Indicates the time when the visual stimulus is shown.
`audio_timestamp`	Indicates the time when the auditory stimulus is played.
`response_time`	Indicates the time when the user provides the answer, -1 otherwise.
`given_answer`	Indicates the answer given by the user.
`correct_answer`	Indicates the correct answer.
`start_resume_game`	Indicates the time when the task started or was resumed.

Table 4.7: This table records user performance for each trial within a block of the Adaptive Dual N-Back cognitive training task.

MrtTrial	
Field	**Description**
`mrt_id`	References the MRT block.
`trial_number`	Indicates the number of the trial within the block.
`inter_trial_time`	Indicates the time between each trial.
`visual_timestamp`	Indicates the time when the visual stimulus is shown.
`response_time`	Indicates the time when the user provides the answer, -1 otherwise.
`given_answer`	Indicates the answer given by the user.
`correct_answer`	Indicates the correct answer.
`start_resume_game`	Indicates the time when the task started or was resumed.

Table 4.8: This table records user performance for each trial within a block of the Mental Rotation Test cognitive training task.

Questionnaire related tables

A structured design with five new tables has been developed to support flexible questionnaire creation and response management. The system allows for predefined questions and answers, facilitating the creation of questionnaires by selecting from an existing pool of questions.

UserAnswer (table 4.13) references the specific questionnaire session (table 4.12), the questions presented, and the corresponding answers provided. This design ensures the questionnaires remain customizable while enabling efficient tracking and organization of user responses.

In order to fill 4.9 , 4.10 and 4.11 a script has been created to run the first time the tables are created.

Question	
Feature	**Description**
`question_id`	This the primary key.
`question`	This is the text of the question.
`area`	This categorizes the question's area (e.g., functional, memory).
`position`	This indicates the order of the question.
`question_type`	This indicates the format of the question (e.g., single, multiple).
`notes`	This provides additional information related to the question.

Table 4.9: Question stores individual questions that can be used in questionnaires.

Answer	
Feature	**Description**
`answer_id`	This is the primary key.
`answer`	This is text of the answer.
`value`	This is the value of the answer.

Table 4.10: Answer stores possible answers that can be associated with questions.

QuestionOptions	
Feature	**Description**
`question_id`	refers to a question.
`answer_id`	refers to an answer.
`position`	determines the order of the answer.

Table 4.11: QuestionOptions links a question to its possible answers.

Questionnaire	
Feature	**Description**
`questionnaire_id`	This is the primary key of the questionnaire table.
`user_id`	Refers to the user that is taking the questionnaire.
`timestamp`	Indicates the time and date when the questionnaire was taken.

Table 4.12: Questionnaire stores an individual questionnaire sessions completed by users.

UserAnswer	
Feature	**Description**
`questionnaire_id`	Refers to a specific questionnaire.
`question_id`	References a specific question.
`answer_id`	References the chosen answer.
`user_value`	This is a custom value for questions that require user input.

Table 4.13: UserAnswer stores for each question in a specific questionnaire the answer given by the user.

Chapter 5

Technology

Powerful and efficient desktop applications rely on a diverse set of technologies. These technologies range from programming languages and frameworks to libraries and tools that ensure the applications are fast, scalable, and user-friendly. This chapter will describe the key technologies behind LC Neurorehab.

5.1 Python

Python is an interpreted, object oriented, high-level language with dynamic semantics. Its adaptability, versatility and the vast range of libraries makes it the ideal language to write clean, organised and maintainable code for a variety of applications. The next two paragraphs will describe two important libraries used to develop this application: PyQt and PyLSL.

5.1.1 PyQt

PyQt is a powerful toolkit for building graphical user interfaces (GUIs) in Python. It provides official Python bindings for the Qt framework, making it easy to create Qt-based applications. Qt itself consists of various components and modules tailored for different development needs, such as creating command-line tools, managing files, handling network connections, or designing a user interface like in the case of LC Neurorehab.

The main reasons to use this library are:

- **QtWidgets Module**
  PyQt's QtWidgets module offers a vast selection of built-in widgets, such as buttons, labels, menus, and more. These widgets allow developers to build fully functional user interfaces across multiple platforms without needing to modify code for different operating systems.

- **Adaptive Layouts**
  PyQt makes it easy to manage the layout of widgets, ensuring that applications adapt to different screen sizes and resolutions. This guarantees that the UI remains responsive and functional across a range of devices, providing an optimised user experience.

- **Event Handling via Signal-Slot Mechanism**
  PyQt uses a signal-slot mechanism to manage events such as button clicks, form submissions, and other user interactions.

  – *Signal*: A signal is an event or notification emitted by an object when a particular action occurs. For example, when a user clicks a button, it emits a signal (such as clicked()).

  – *Slot*: A slot is a function designed to respond to a signal. In Python, any function can serve as a slot. Once a signal is emitted, the connected slot function is automatically executed.

The signal-slot mechanism enables loose coupling between objects, meaning the object emitting the signal doesn't need to be aware of which object will handle it. This flexible communication system is essential for building interactive, user-friendly applications in Qt. PyQt integrates seamlessly into this structure by maintaining a clear separation between the interface and the underlying logic.

This makes it well-suited for the MVC architecture, as signals from the View (such as button clicks) can easily notify the Controller, which then updates the Model. In turn, the Model can emit signals to update the View, all without requiring tight coupling between the components.

5.1.2 pyLSL

PyLSL is a Python library designed to interact with the Lab Streaming Layer (LSL), a protocol widely used for real-time acquisition, synchronisation, and recording of time-series data. LSL is particularly popular in fields like neuroscience, bio-signal processing, and brain-computer interface (BCI) research. It provides a unified interface for transmitting data from a variety of devices, such as EEG systems and heart rate monitors, ensuring that all streams are accurately time-synchronised.

PyLSL allows for a direct interaction with LSL streams within Python, making it easy to interface with physiological sensors, capture real-time data, and either store or analyse that data instantly. One of the main advantages of this library is that many EEG systems currently on the market are compatible with LSL. This compatibility simplifies the process of synchronising behavioural tasks with EEG signals.

A common use case for PyLSL, specifically also for LC Neurorehab, involves creating streams for transmitting event markers or application states, which are essential for stimulus-presentation programs. In such cases, the stream may be a single-channel, with an irregular sampling rate, and the value per channel being a string. This makes PyLSL particularly useful for applications where time-sensitive event marking is required, such as experiments involving cognitive or behavioural tasks.

```
1    from pylsl import StreamInfo, StreamOutlet
2
3    def set_up_pylsl(self):
4        info = StreamInfo("EEGStream", "Markers", 1,
5                          0, "string", "myuidw43536")
6
7        self.outlet = StreamOutlet(info)
8
9    def send_marker(self, marker):
10       self.outlet.push_sample([marker])
```

Table 5.1: LSL stream for EEG data.

The code in table 5.1 shows how to set up and send data on LSL stream. First, all of the necessary classes from the pylsl library need to be imported. StreamInfo is used to define the stream's metadata while StreamOutlet is responsible for sending the data

In the setup function an instance of StreamInfo is created with the following parameters:

- **"EEGStream"**: The name of the stream.
- **"Markers"**: The type of stream (could be "EEG", "Markers", etc.).
- **1**: The number of channels (1 for a single marker stream).
- **0**: The sampling rate (0 indicates that the stream does not have a specific rate).
- **"string"**: The data type of the stream (in this case, a string for markers).
- **"myuidw43536"**: A unique identifier for the stream.

The StreamOutlet instance is created using the info object enabling the data transmission through the LSL, the send_marker function will be used to send a string through the stream.

5.2 Database

The database of the application is created using SQLite, a C library that provides a lightweight disk-based database that does not require a separate server process and allows accessing the database using a nonstandard variant of the SQL query language.

It provides several benefits:

- **Serverless Architecture**: SQLite does not require a separate server process or management system. The database is directly integrated into the application, reducing complexity and eliminating the need for database administration.
- **Zero Configuration**: With no setup required, SQLite can be used immediately. There are no configuration files, and the database operates without needing to install or manage database servers.
- **Single Database File**: The entire database is stored in a single file, which can be easily transferred, backed up, or integrated into an application. This portability makes SQLite particularly useful for applications that require offline storage or local databases.
- **ACID Compliance**: SQLite is fully ACID-compliant (Atomicity, Consistency, Isolation, Durability), ensuring that all database transactions are reliable and secure, even in the event of a system crash or failure.
- **SQL Standard**: SQLite supports most of the SQL-92 standard, allowing developers to write familiar SQL queries for data manipulation and retrieval. Cross-Platform: SQLite works on virtually any operating system, including Windows, macOS, Linux, iOS, and Android, making it versatile and adaptable for many use cases.

Given the scale of the project and the environment in which the application will be used, there was no need to switch the current database library. However, if in the future a need for synchronising the database across multiple instances of the application arises a more suitable database solution should be considered.

5.3 JSON

JSON (JavaScript Object Notation) is a lightweight, human-readable format for data exchange, and it's also easy for machines to parse and generate. It works on two primary structures:

- **Name/Value Pairs**: Represented as objects, dictionaries, or associative arrays in various languages.
- **Ordered Lists of Values**: Commonly realised as arrays or sequences.

These structures are universal and supported across most modern programming languages, making JSON a natural choice for data interchange. In the scope of this project it has been used to define some tasks constraints, allowing for their modification without impacting the code of the application. By decoupling application logic from business rules, the JSON file ensures that the core logic remains unaffected by changes in system settings.

5.3.1 JSON Schema

In order to describe how the JSON file should be structured the JSON Schema reported below has been created.

A JSON Schema precisely defines how a record should be organised, offering a structured vocabulary to establish clear rules for the data architecture.

The schema provides a way to describe the expected structure of the data, including the types of data (e.g., strings, numbers, objects, arrays) and constraints on the values (e.g., minimum or maximum values, required fields).

The JSON schema is made of different components:

- **$schema**, it defines the version of the JSON schema specification from which features and rules for writing the schema are taken.
- **type**, it specifies the data structure such as objects, numbers, integers, boolean.
- **properties**, it defines the properties of the JSON object. In the case of this application: dualnback and mrt. Each of them defined as an object containing other nested properties.
- **constraints**, each type of data structure might come with some constraints that a specific data type must have (e.g., minimum, maximum, exclusiveMaximum and/or required).

The schema ensures that the JSON data adheres to a defined structure, it helps catch errors and defines documentation. The schema allows also for adaptability, with its extremely flexibility new tasks can be added without compromising the existing application.

```
1    {
```

```
        "$schema": "http://json-schema.org/draft-07/schema#",
        "type": "object",
        "properties": {
            "dualnback": {
                "type": "object",
                "properties": {
                    "blocks": {
                        "type": "integer",
                        "minimum": 5,
                        "exclusiveMaximum": 15
                    },
                    "base_trials": {
                        "type": "integer",
                        "minimum": 20
                    },
                    "accuracy": {
                        "type": "object",
                        "properties": {
                            "low": {
                                "type": "number",
                                "minimum": 0,
                                "exclusiveMaximum": 100
                            },
                            "high": {
                                "type": "number",
                                "minimum": 0,
                                "maximum": 100
                            }
                        }
                    }
                }
            },
            "mrt": {
                "type": "object",
                "properties": {
                    "blocks": {
                        "type": "integer",
                        "minimum": 5,
                        "exclusiveMaximum": 15
                    },
```

```
                    "trials": {
                        "type": "integer",
                        "minimum": 10,
                        "exclusiveMaximum": 30
                    },
                    "assess_user": {
                        "type": "object",
                        "properties": {
                            "level_1": {
                                "type": "number",
                                "minimum": 0,
                                "maximum": 100
                            },
                            "level_2": {
                                "type": "number",
                                "minimum": 0,
                                "maximum": 100
                            },
                            "level_3": {
                                "type": "number",
                                "minimum": 0,
                                "maximum": 100
                            }
                        }
                    },
                    "trial_duration": {
                        "type": "number",
                        "minimum": 0,
                        "maximum": 100
                    },
                    "level_duration": {
                        "type": "integer",
                        "minimum": 0,
                        "maximum": 100
                    },
                    "game_duration": {
                        "type": "integer",
                        "minimum": 0,
                        "maximum": 100
                    },
                    "isTimeBased": {
```

```
                        "type": "boolean"
                    },
                    "accuracy": {
                        "type": "object",
                        "properties": {
                            "low": {
                                "type": "integer",
                                "minimum": 0,
                                "maximum": 100
                            },
                            "high": {
                                "type": "integer",
                                "minimum": 0,
                                "maximum": 100
                            }
                        }
                    }
                }
            }
        }
    }
```

To verify that a JSON instance complies with its schema, a JSON schema validator is required. The validation is obtained with the use of the *jsonschema* library which is an implementation of the JSON Schema specification for Python.

These validators can easily be integrated into projects of varying sizes, ensuring data consistency, automating testing, long-term maintainability and scalability of the system.

5.4 Support to development

GitHub

GitHub is an online software development platform. It's used for storing, tracking, and collaborating on software projects. GitHub is built around Git, a distributed version control system that enables developers to track changes in their code over time. This allows to experiment, revert to earlier versions, and understand the history of their codebase. Git's branching and merging features make it easy to develop new features or fix bugs without disrupting the main project.

Chapter 6

Results obtained

In this chapter all the new developed features and enhancements made to LC-Neurorehab will be presented. It will examine in detail the new functionalities, to which I personally contributed, introduced for both psychologist and patients.

6.1 Questionnaire

6.1.1 Daily Questionnaire

The questionnaire is designed with a precise set of questions, focusing on key cognitive areas of Long Covid symptomatology, such as functioning, fatigue, memory, concentration, motivation and quality of life, while also investigating physical symptoms and the mood of the participants.

It is a crucial part for psychologists wanting to obtain a comprehensive understanding of the patients' overall condition on the day the experiment is carried out, as it provides the context for comparing their performance on each cognitive tasks with how they felt on that specific day. The questions give a detailed insight into participants' experiences, thoughts, and perceptions, offering a deeper understanding of their journey with Long Covid and their rehabilitation.

Most of the questions are multiple choice single answer. They allow participant to pick only one answer from a list. Each answer is linked to a specific numerical value, ranging from 1 to the total number of answers, in order to capture the general state of the participant in each area. Only the Long Covid physical symptoms question is multiple choice and multiple answer, where users are asked to choose the symptoms they are experiencing and then rank them from worst to least bad.

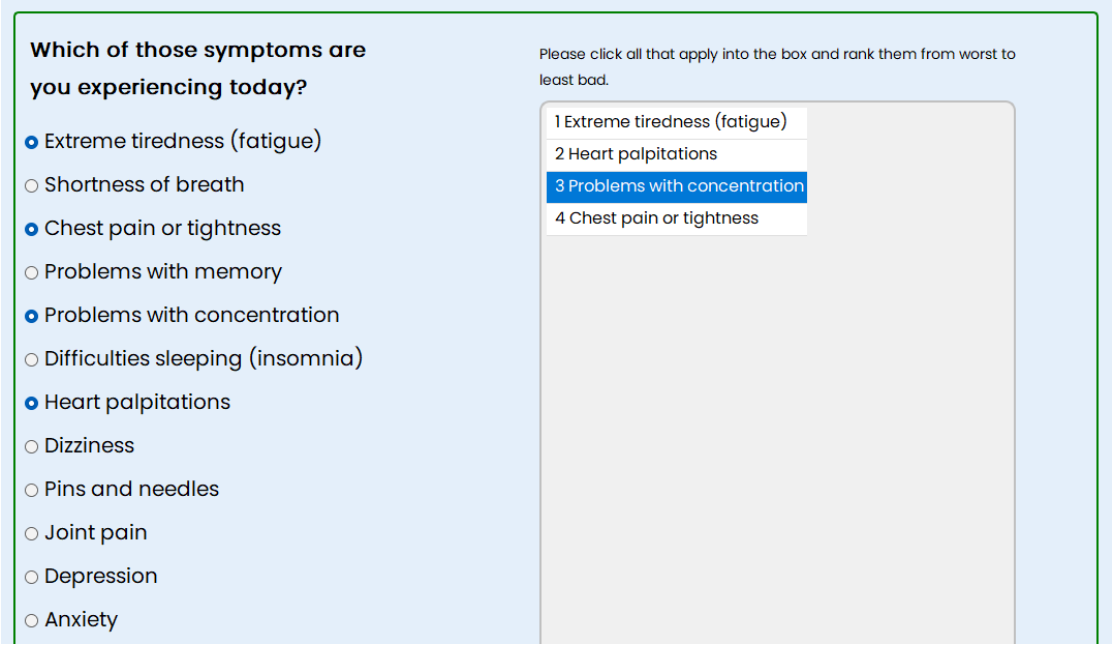

Figure 6.1: Ranking question

This ranking is facilitated by a ListView and a delegate item, using the Model-View (MV) pattern. Through this pattern, a model is linked to the ListView, allowing the delegate to modify the view.

The Mood section is designed to assess how the participant is acting or feeling that week. It consists of a series of words that describe emotions for each one of them the participant needs to answer on a scale from 1 to 5 ranging from "Very slightly or not at all" to "Extremely."

It is mandatory to answer all the questions before patients can go to the homepage and access the list of tasks. When a user answers a question the corresponding box turns green, if some question goes unanswered when the submit button is pressed, a pop up will appear asking to answer all the question, all the boxes contained unanswered questions will turn red and the user will be redirected to the first unanswered question in the list.

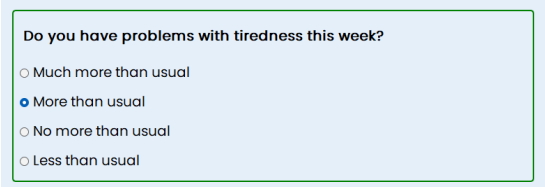

Figure 6.2: Question Answered

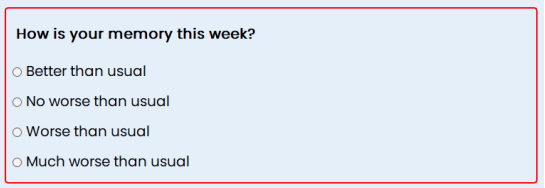

Figure 6.3: Unanswered questionnaire question

The list of the questions used in the questionnaire, their answer and the values assigned can be found in the appendix.

6.1.2 End of block questionnaire

Besides the daily questionnaire, participants will be required to complete a follow-up assessment at the end of each task block where they will be asked to rate, on a scale from 1 to 10, how they feel about three key areas:

- **Engagement**: This is a critical metric in psychological research. Engagement refers to a set of mindful, goal-directed states driven by motivation, often arising from positive emotions. It is essential for sustaining cognitive and motor functions, typically requiring effort. Research has shown that engagement is predictive of academic proficiency, motivation, and task persistence. The balance between a patient's interest and the demands of the task influences the strength of their engagement.

- **Perceived difficulty**: This metric assesses the patient's perception of task difficulty, particularly important when tasks are repeated or adjusted in difficulty. Increased perceived difficulty has been shown to correlate with higher levels of fatigue.

- **Mental fatigue**: It refers to a psycho-biological state characterised by tiredness resulting from extended periods of engaging in cognitively demanding activities. This fatigue diminishes cognitive performance efficiency.

The chosen values will be recorded alongside each block data, providing a subjective context to the result of the performance.

After completing the block questionnaire, patients are given the option either to continue with their ongoing task or to stop the experiment.

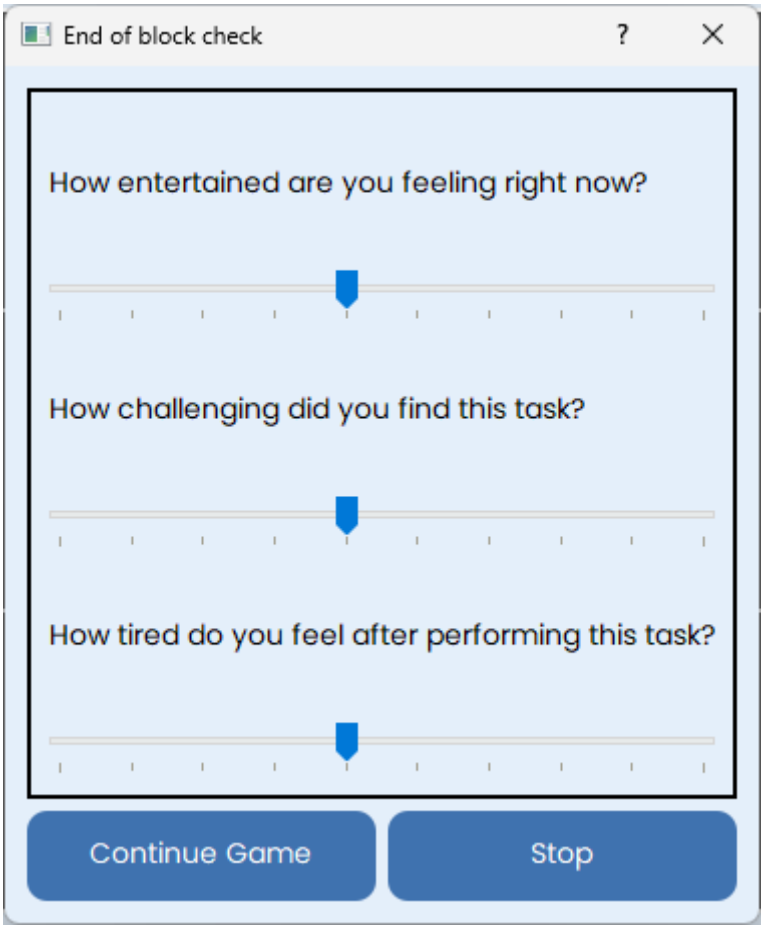

Figure 6.4: This dialog is presented at the end of each block.

6.2 User Experience

User experience is a critical aspect of any application, especially in desktop environments where users interact with the interface for extended periods.

Some issues were addressed in order to have a well-designed user interface with new functionalities to minimize frustration and cognitive load for the user, allowing them to focus on the task at hand without being blocked by the application's design.

The first problem to tackle was the lack of responsiveness in the user interface of the desktop application, as already stated while talking about the nonfunctional requirements.

The original design had fixed dimensions, the application could not be resized or maximized to fit the user's screen. This rigid layout, besides affecting the usability of the app, might have an impact directly on patients, as they might be distracted

by other elements in the background. The correct use of Qt's layout classes, such as *QVBoxLayout*, *QHBoxLayout*, and *QGridLayout* has solved the issue allowing the application to dynamically adjust to different screen sizes and resolutions allowing them to focus solely on the task without distractions.

New developments, described in the next paragraphs, have been added to the application to reduce the cognitive load and ease the experience of patients using the app: task pausing and resuming and tutorial and examples.

6.2.1 Task Pausing and Resumption

A new feature introduced during this thesis project was the possibility to pause an ongoing task.

A patient can pause the task whenever they want by either clicking the pause button located in the upper-right corner of the screen or by pressing the P key on the keyboard.

When the button is clicked, it will emit the *clicked()* signal connected to a slot function in the controller that will execute the pause logic. The controller will call the correct view to display, the pause button will be replaced with the back button, the displayed task will be hidden, and a resume button will appear.

All the active timers will be stopped, and the elapsed time of each timer will be assigned to a corresponding variable.

If the user decides to resume the task he will have to click the resume button or press the R key on the keyboard, timers will be set with the elapsed time values, and the task will resume from the point where it was paused

If the patients wishes to quit the task, once they have paused it they can press the back button, which will redirect them to the homepage.

If during the day the user is required by the professional to resume the task they will be presented with two options:

- Start a new session.
- Resume the previous session.

By resuming the previous session, the user will start from the beginning of the latest block reached when he quit the session.

It is worth reminding that this application is to be used in a laboratory environment under the supervision of psychologists and experts. The decision to quit or pause the task might come directly from them after evaluating the state of the patient.

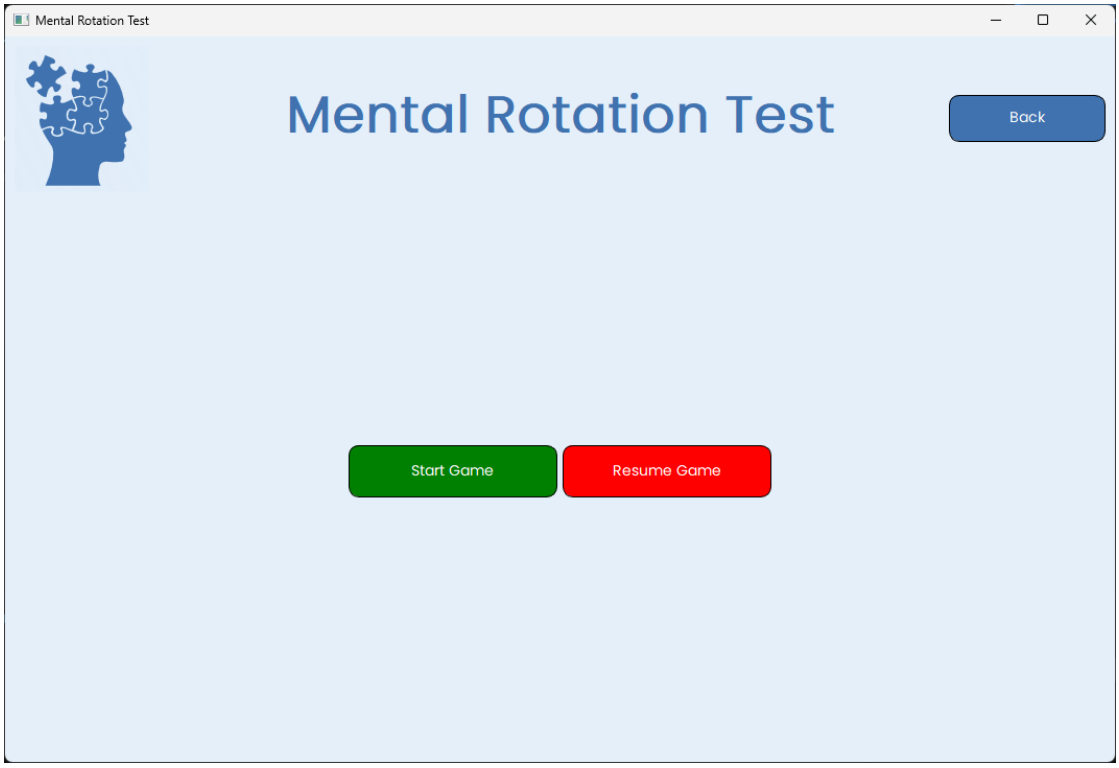

Figure 6.5: This screen is shown when the participant has quit a MRT session.

6.2.2 Tutorials and Examples

Due to the inherent difficulty of the tasks, a set of written instruction was not enough for users to understand how to perform during the task. So two different approaches have been chosen: for the dual n-back task, a tutorial has been developed to improve user understanding, while for the MRT a set of visual examples have been introduced.

Dual N-Back Tutorial

The tutorial is divided into two sections:

- **Guided Introduction**: This section introduces the user to the game with simpler versions of the task, such as 1-back and 2-back. During this phase, a series of pop-up prompts will guide the user, explaining whether there is no match, an audio match, a visual match, or no match for each stimulus. If the user does not fully understand the example, they have the option to replay it. Additionally, a panel on the left-hand side will display the audio and visual

stimuli for each trial, providing visual support to help users better grasp the concept.

- **Practice Mode**: After completing the guided introduction a pop up appears telling the user is in practice mode. In this section, users can freely perform the task without having their responses tracked. This untracked practice mode allows users to familiarise themselves with the game mechanics without the pressure of performance evaluation.

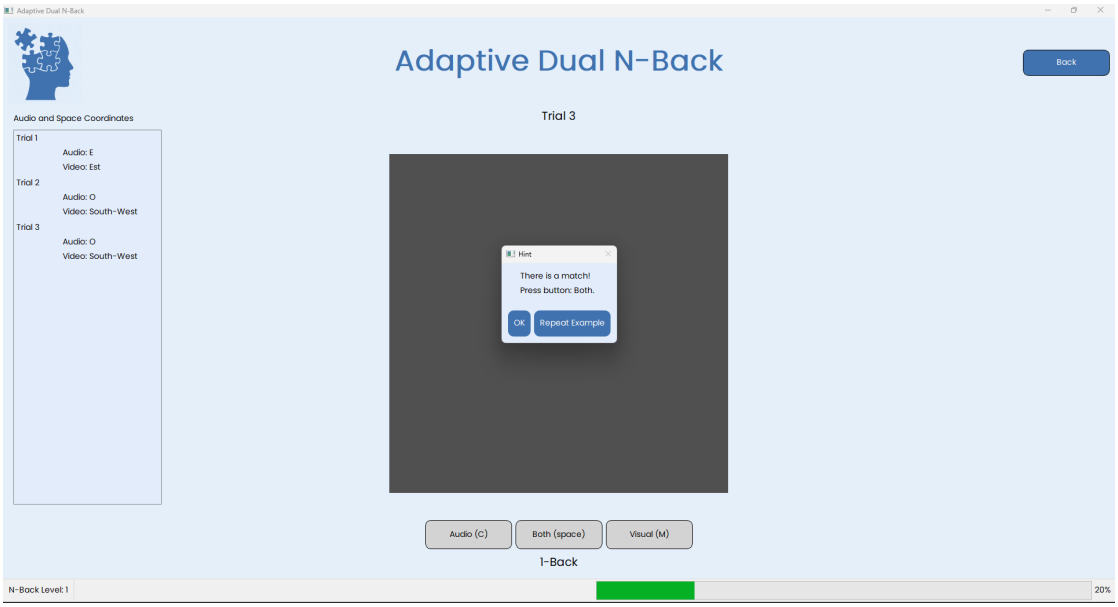

Figure 6.6: Dual N-Back Tutorial

Mental Rotation Test Examples

In the context of MRT, examples are needed in order to simplify and improve the understanding of the task. Without a visual aid, the user may have some difficulties in identifying the criteria that determines the similarity of two images.

By showing concrete examples that highlight the distinction in angle rotation and perspective between identical and non-identical images, patients will understand the task without worrying about its rules or nuances, resulting in a more reliable and meaningful outcome.

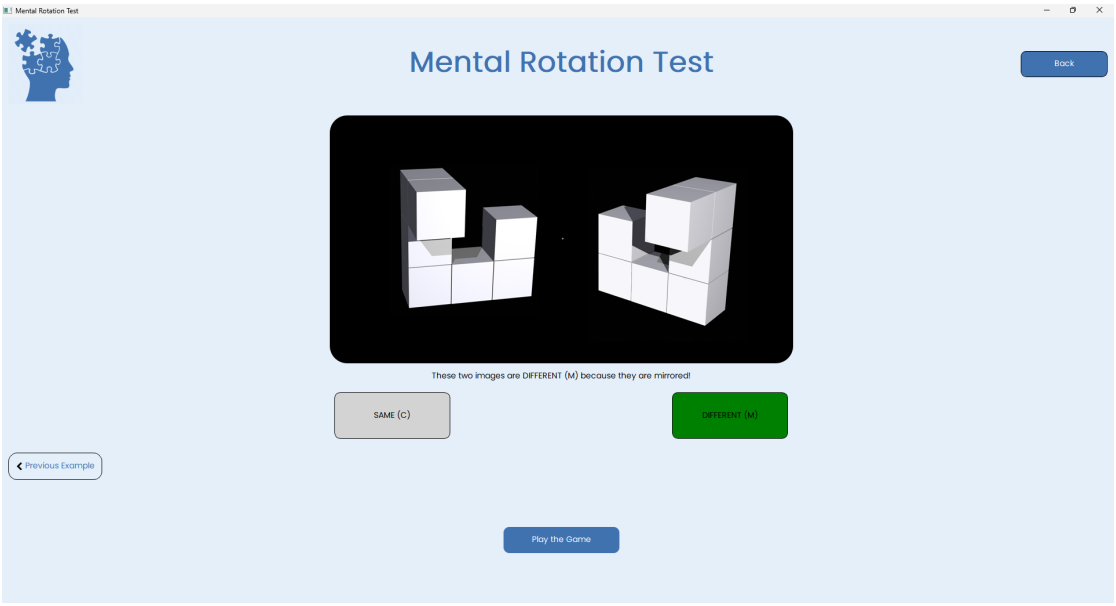

Figure 6.7: This screen show the example for MRT.

Another modification made to the MRT was the alignment of the task behavior with existing literature. While the assessment level remained unchanged, the previous time-based structure was replaced with a block format. The task now consists of five blocks in addition to the assessment level, with each block made of 20 pairs of images. The difficulty of each block is determined by the corresponding level. From the admin area it will still be possible to enable the old behavior.

6.3 Admin Area Enhancements

One of the main objective of the thesis was to have a tool that not only benefited patients affected by Long Covid but also could supported professionals in understanding the effects and the response to the therapy by giving them the possibility to tailor the experiments to the type of patients by modifying some task parameters, to download the data collected or visualize it with a finer granularity.

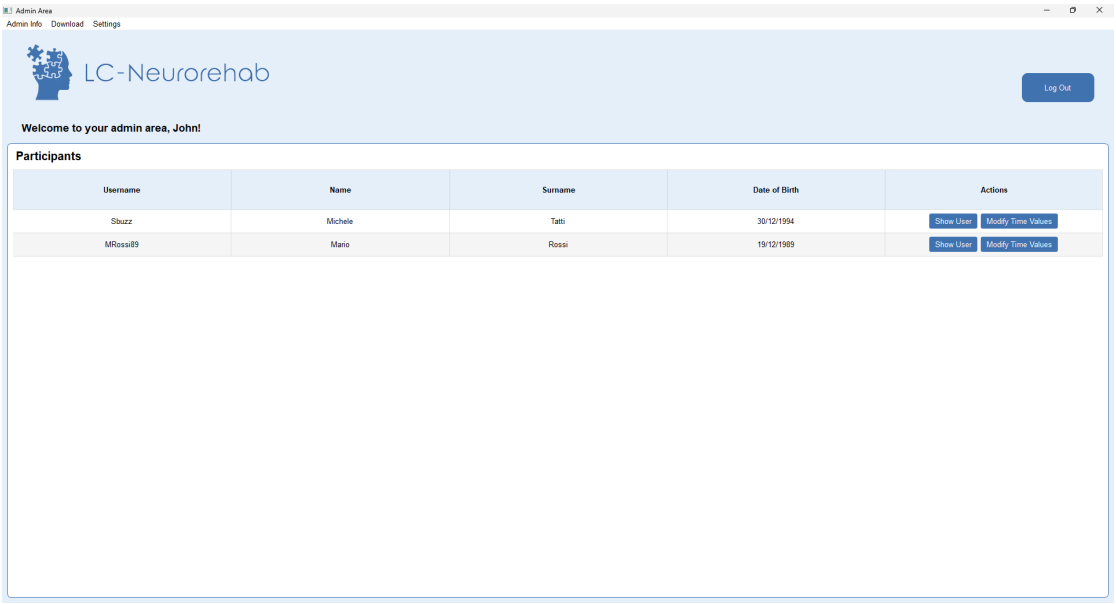

Figure 6.8: Admin area.

The following paragraphs will describe all the changes made to the admin area and the user page.

6.3.1 Menu Bar

As shown in figure 6.9 a menu bar has been added to the admin area. This menu bar is made of 3 different buttons:

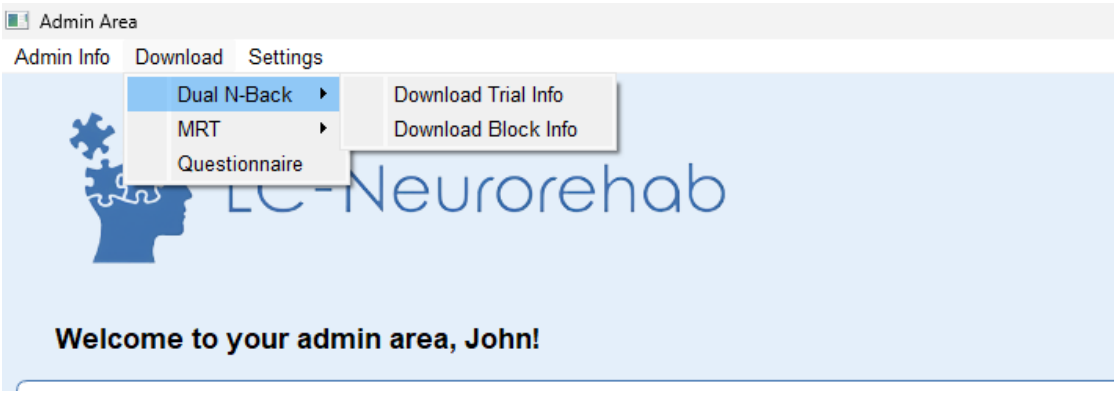

Figure 6.9: Menu Bar.

- Admin Info

- Download
- Settings

When clicking over the *Admin Info* section, a dialog window listing all the registered admin users within the application will be shown. This list also acts as a log, tracking each admin last login time and the most recent modification time.

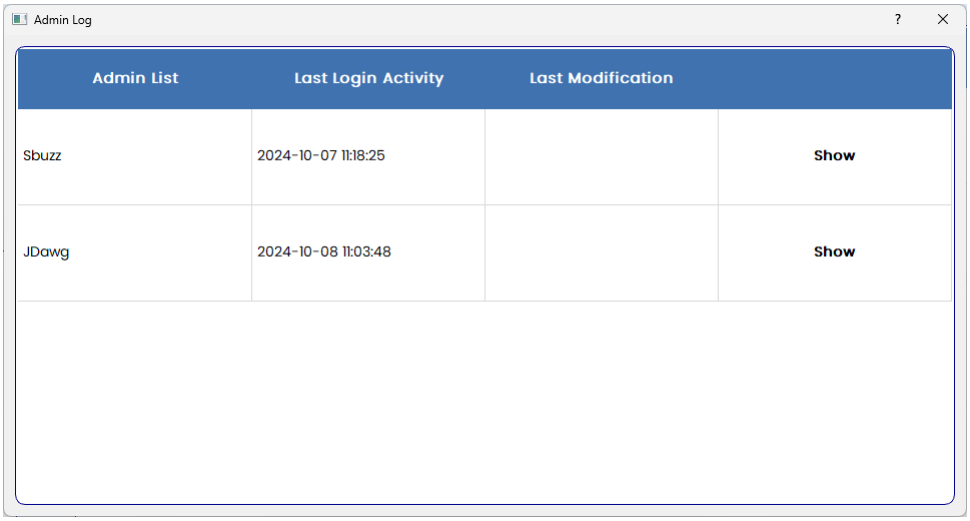

Figure 6.10: Admin Log.

The *Download* option in the menu bar opens a submenu that allows professionals to export aggregated user data for a specific task or questionnaire in CSV format. Data for the tasks can be downloaded either by trial or by block, depending on the required level of granularity, enabling experts to track user performance and progression over time.

```
user_id ,timestamp                ,Q1_Functioning ,Q2_Fatigue ,Q3_Memory ,Q4_Concentration ,Q5_Motivation ,Q6_QoL-Socialising
      1 ,2024-09-11 22:22:10.729419 ,             5 ,         4 ,         4 ,                4 ,            4 ,                 1
      1 ,2024-09-12 12:52:53.521106 ,             3 ,         3 ,         3 ,                2 ,            2 ,                 4
```

Figure 6.11: Questionnaire CSV File.

The *Settings* option opens a dialog where task-related parameters can be adjusted. This is where the JSON schema comes into play, enabling administrators to adapt tasks without altering the underlying code while adhering to predefined boundaries. This flexibility allows admin users to easily modify settings, such as task duration or difficulty, based on their assessments throughout the course of the experiments.

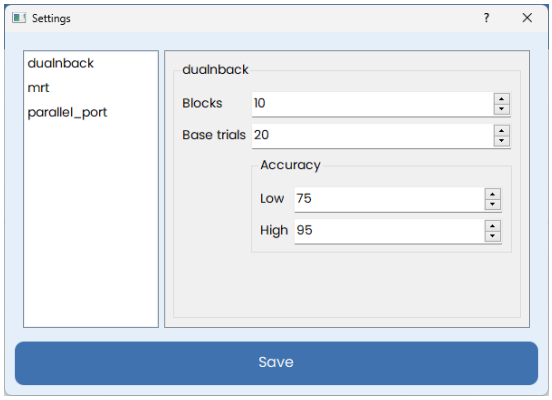

Figure 6.12: Settings Dialog

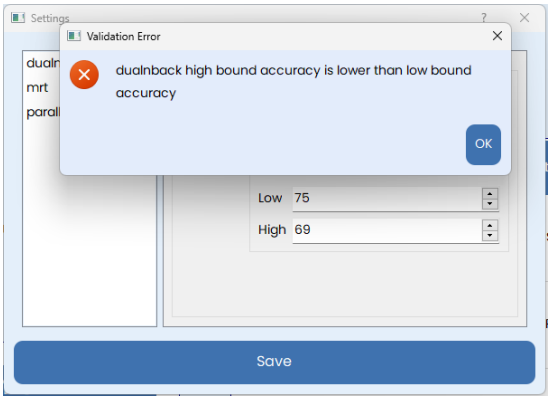

Figure 6.13: Warning coming from settings validation

6.3.2 Task Times Modification

The Admin Page also allows for the modification of specific task time parameters for individual users, enabling the adaptation of task difficulty to meet each user's needs. For instance, a patient with above-average performance can be challenged by reducing time constraints, pushing them to perform at a higher difficulty level. For those facing difficulties instead, extending the time can make the task more manageable, improving the likelihood of success and reducing pressure. This adaptive approach ensures that each user is appropriately challenged according to their abilities.

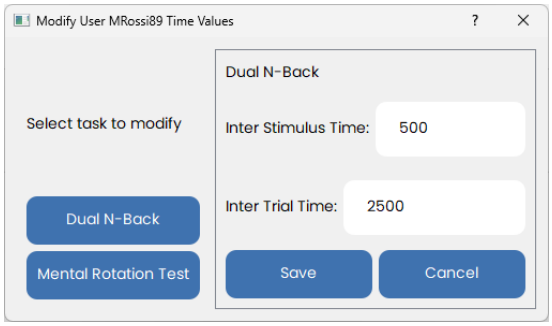

Figure 6.14: Dual N-Back Time Values Modification

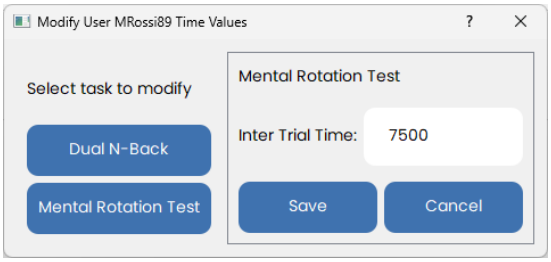

Figure 6.15: MRT Time Values Modification

6.3.3 Participant Page

Several enhancements have been made to the participant page to improve functionality and user experience. Since we added questionnaires, professionals are enabled to download questionnaire data either by specific day by choosing a specific questionnaire from the panel under the participant information or retrieve all questionnaires related to a single participant by pressing the “download all” button. This offers greater flexibility in data analysis, allowing for more focused or comprehensive data extraction.

Additionally, task data is now more accessible with a streamlined interface. Instead of navigating between multiple pages to view task-related information, users can now simply select a task from a list and see the corresponding data displayed directly in a widget on the same page. This eliminates the need to switch back and forth between pages, offering a more efficient workflow and making it easier to analyse task performance at a glance.

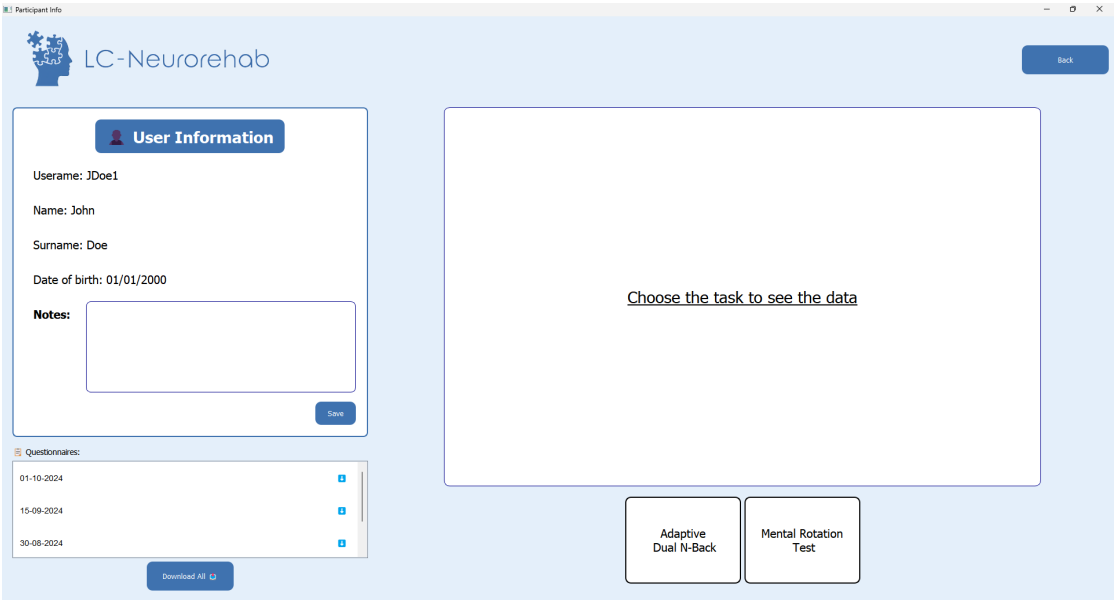

Figure 6.16: Participant info page.

The table includes an index, the date of task completion, a “Show Data" button that allows the administrator to access the patient’s data related to that specific activity session and a download button to download the data related to that session.

Session Info

When the admin selects a specific session, a table containing the information relate to that activity session of that specific patient is displayed.

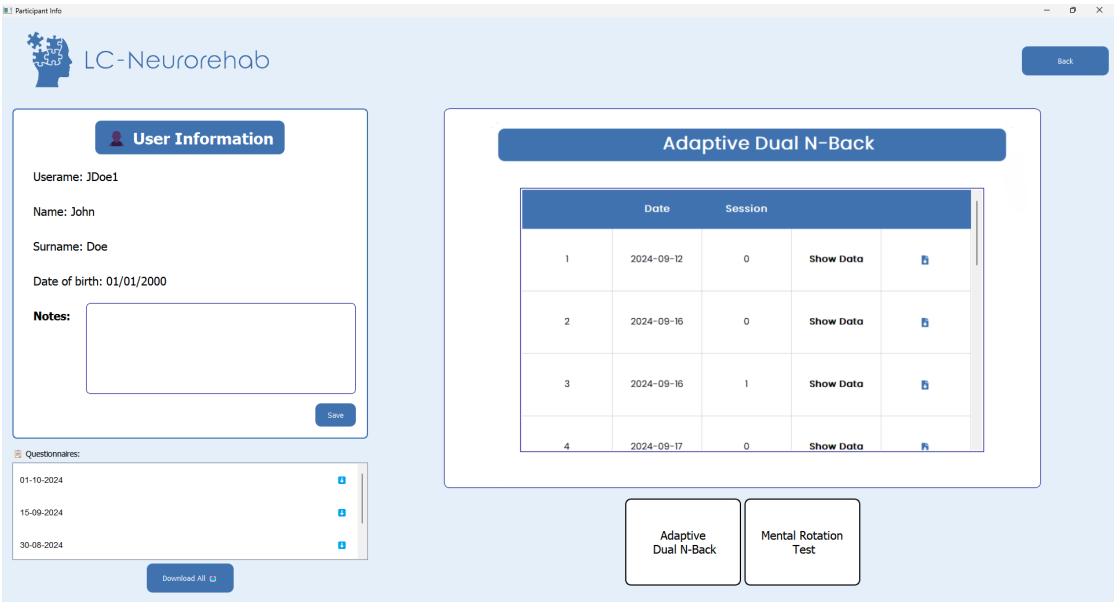

Figure 6.17: Trial Visualization.

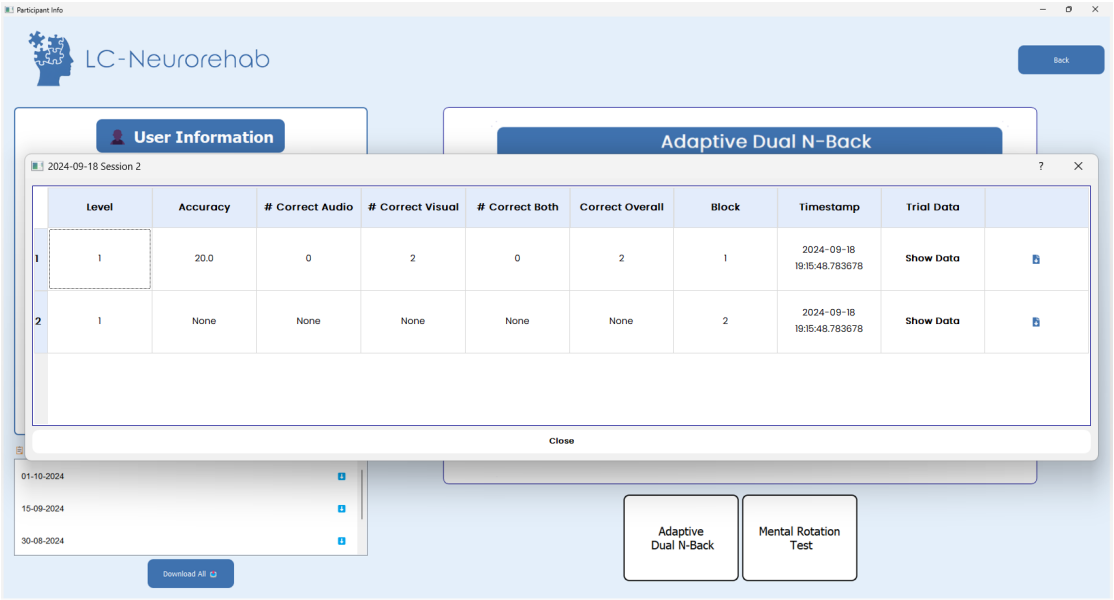

Figure 6.18: Block Visualization.

For the Dual N-Back the data displayed will be: level,accuracy, number of user correct audio, number of user correct visual, number user both correct answer, the total number of correct answers, the session timestamp, the block, and the user

level of entertainment, tiredness and difficulty.

For the Mental Rotation task the displayed data will be: level,accuracy, total number of shown images, number of correct answers, number of wrong answers, the block, the session timestamp and the user level of entertainment, tiredness and difficulty.

Trial Info

For each activity session it is possible to display and/or download in a CSV format all the information related to trials of that session block. The download button is in the last column in figure 6.18.

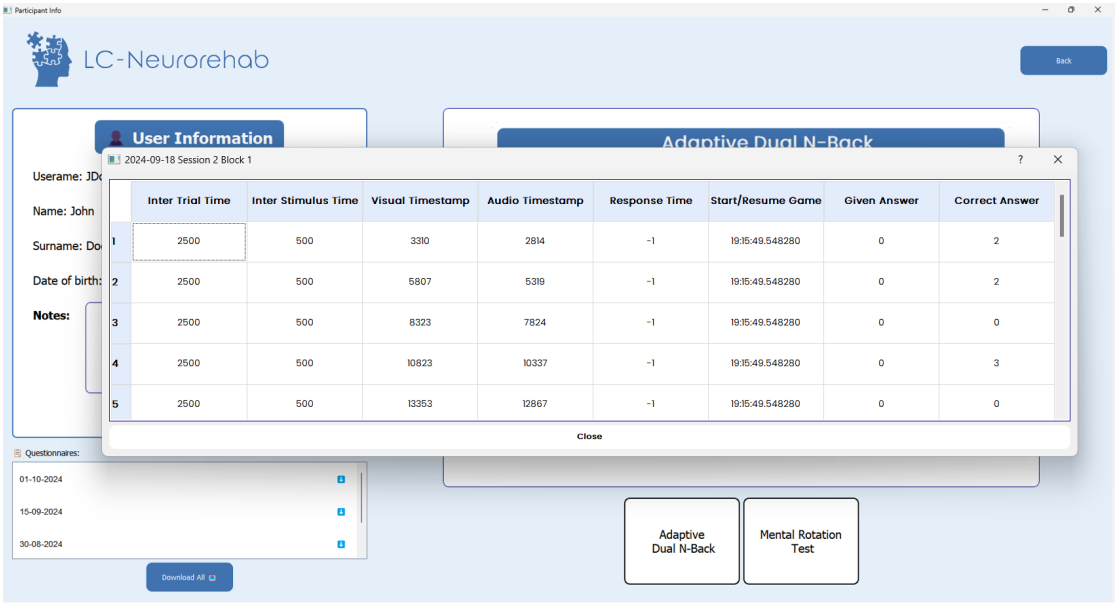

Figure 6.19: Trial Visualization

The information shown is the inter trial time and if present inter stimulus time in ms, the visual and if present audio timestamp in ms from the beginning of the session, the response time from the beginning of the session, the given answer and the correct answer, and the start resume time.

Chapter 7

Conclusion

LC-Neurorehab is an ongoing and still evolving project. We now have a tool that is fully developed and ready to be used in this medical field where nowadays lots of research is being made without proper equipment to support it.

This current prototype serves as a solid foundation, and once it will be deployed, it will facilitate the collection of valuable information on how patients respond to therapy and which factors contribute to it.

Alongside data coming from the application there will also be a collection of EEG signal data that combined with the temporal markers provided by the application, will help identify the changes in the brain activity of the patients responding to the therapy.

Studying these signals and extrapolating their most important features will be vital for gaining a deeper understanding of the patients' experiences and recovery dynamics. It will be beneficial for the application itself, as that could be incorporated in the automatic adjustment of the difficulty of the task.

Indeed, after having collected these data, an AI system can be created to adjust automatically the task to the patient status, this will also have no impact on the existing database structure, as it is already thought to cover these possible changes. This can aid doctors and psychologists giving them the possibility to solely focus on the patients status and their evaluation. The gathered data could be displayed by means of graphs to visually examine the course of the patient state in search of common trends.

Another potential enhancement is the implementation of rigorous automated testing to ensure that the application continues to behave as expected, even with the introduction of new functionalities. Automated testing is crucial, particularly for applications that have only been manually tested, as it helps identify bugs and issues early in the development process. Manual testing can be time-consuming and prone to human error, leading to inconsistencies in testing coverage. By incorporating automated tests, we can ensure comprehensive validation of the

application's behavior across various scenarios, improving the reliability of the software.

This application is a testament of how problems nowadays must be challenged with a multidisciplinary approach, by bringing together experts of different domains we can create tools that allow for quicker identification and analysis of problems leading to more innovative solutions and improved outcomes in tackling complex issues.

Appendix A

Daily Questionnaire

The following is the complete questionnaire administered to every patient using the application.

Question 1. [Functioning]

How much have you been affected in your everyday life by long-lasting symptoms of Covid-19?

Answers		**Value**
1	I have no limitations in my everyday life and no symptoms, pain, depression or anxiety related to the infection.	5
2	I have negligible limitations in my everyday life as I can perform all my usual duties/activities, although I still have persistent symptoms, pain, depression or anxiety.	4
3	I suffer from limitations in my everyday life as I occasionally need to avoid or reduce usual duties /activities or need to spread these over time. due to symptoms, pain, depression or anxiety. I am, however, able to perform all activities without any assistance.	3
4	I suffer from limitations in my everyday life as I am not able to perform all usual duties/activities	2

	due to symptoms, pain, depression or anxiety. I am, however, able to take care of myself without any assistance.	
5	I suffer from severe limitations in my everyday life: I am not able to take care of myself and therefore I am dependent on nursing care and/or assistance from another person due to symptoms, pain, depression or anxiety.	1

Question 2. [**Fatigue**]

Do you have problems with tiredness this week?

Answers		**Value**
1	Much more than usual	4
2	More than usual	3
3	No more than usual	2
4	Less than usual	1

Question 3. [**Memory**]

How is your memory this week?

Answers		**Value**
1	Better than usual	4
2	No worse than usual	3
3	Worse than usual	2
4	Much worse than usual	1

Question 4. [**Concentration**]

Have you had any difficulty concentrating this week?

Answers		**Value**
1	Much more than usual	4
2	More than usual	3
3	No more than usual	2
4	Less than usual	1

Question 5. [**Motivation**]

Have you had problems starting things this week?

Answers		**Value**
1	Much more than usual	4
2	More than usual	3
3	No more than usual	2
4	Less than usual	1

Question 6. [**Quality of Life**]

How would you rate your ability to socialise this week?

Answers		**Value**
1	Terrible	1
2	Unhappy	2
3	Mostly dissatisfied	3
4	Mixed	4
5	Mostly satisfied	5
6	Pleased	6
7	Delighted	7

Question 7. [**Long Covid Symptoms**]

Which of these symptoms are you experiencing today?

Extreme tiredness (fatigue)
Chest pain or tightness
Problems with concentration
Heart palpitations
Pins and needles
Depression
Tinnitus
Feeling sick
Stomach aches
High temperature
Headaches
Shortness of breath
Problems with memory
Difficulties sleeping (insomnia)
Dizziness
Joint pain
Anxiety
Ear ache
Diarrhoea
Loss of appetite
Cough
Sore throat

Changes to sense of smell or taste	Rashes
Other	None

Patients are asked to select which one of these symptoms are experiencing and rank them from worst to least bad.

Question 8. [Mood]

This scale consists of a number of words that describe different feelings and emotions. Read each item and then mark the appropriate answer in the space next to that word. Indicate to what extent you GENERALLY feel this way, that is how you feel ON AVERAGE.

	Very slightly or not at all	**A little**	**Moderately**	**Quite a bit**	**Extremely**
Interested	1	2	3	4	5
Enthusiast	1	2	3	4	5
Proud	1	2	3	4	5
Distressed	1	2	3	4	5
Excited	1	2	3	4	5
Irritable	1	2	3	4	5
Upset	1	2	3	4	5
Alert	1	2	3	4	5
Strong	1	2	3	4	5
Guilty	1	2	3	4	5
Inspired	1	2	3	4	5
Ashamed	1	2	3	4	5
Scared	1	2	3	4	5
Nervous	1	2	3	4	5
Determined	1	2	3	4	5
Hostile	1	2	3	4	5
Attentive	1	2	3	4	5
Jittery	1	2	3	4	5
Active	1	2	3	4	5
Afraid	1	2	3	4	5

Bibliography

[1] Cody B. Jackson, Michael Farzan, Bing Chen, and Hyeryun Choe. «Mechanisms of SARS-CoV-2 entry into cells». In: *Nature Reviews Molecular Cell Biology* 23.1 (Jan. 2022), pp. 3–20. ISSN: 1471-0080. DOI: 10.1038/s41580-021-00418-x. URL: https://doi.org/10.1038/s41580-021-00418-x (cit. on pp. 3, 4).

[2] Rajiv Dhand and Jie Li. «Coughs and Sneezes: Their Role in Transmission of Respiratory Viral Infections, Including SARS-CoV-2». In: *American Journal of Respiratory and Critical Care Medicine* 202.5 (2020). PMID: 32543913, pp. 651–659. DOI: 10.1164/rccm.202004-1263PP. eprint: https://doi.org/10.1164/rccm.202004-1263PP. URL: https://doi.org/10.1164/rccm.202004-1263PP (cit. on p. 3).

[3] Wei-jie Guan et al. «Clinical Characteristics of Coronavirus Disease 2019 in China». In: *New England Journal of Medicine* 382.18 (2020), pp. 1708–1720. DOI: 10.1056/NEJMoa2002032. eprint: https://www.nejm.org/doi/pdf/10.1056/NEJMoa2002032. URL: https://www.nejm.org/doi/full/10.1056/NEJMoa2002032 (cit. on p. 3).

[4] Islam H. Elrobaa and Karl J. New. «COVID-19: Pulmonary and Extra Pulmonary Manifestations». In: *Frontiers in Public Health* 9 (2021). ISSN: 2296-2565. DOI: 10.3389/fpubh.2021.711616. URL: https://www.frontiersin.org/journals/public-health/articles/10.3389/fpubh.2021.711616 (cit. on p. 4).

[5] Allison B. Reiss, Caitriona Greene, Christopher Dayaramani, Steven H. Rauchman, Mark M. Stecker, Joshua De Leon, and Aaron Pinkhasov. «Long COVID, the Brain, Nerves, and Cognitive Function». In: *Neurology International* 15.3 (2023), pp. 821–841. ISSN: 2035-8377. DOI: 10.3390/neurolint15030052. URL: https://www.mdpi.com/2035-8377/15/3/52 (cit. on pp. 4, 5).

[6] World Health Organization. *Post COVID-19 Condition*. 2023. URL: https://www.who.int/europe/news-room/fact-sheets/item/post-COVID-19-condition (cit. on p. 4).

[7] Sandra Arbula, Elisabetta Pisanu, Giulia Bellavita, Alina Menichelli, Alberta Lunardelli, Giovanni Furlanis, Paolo Manganotti, Stefano Cappa, and Raffaella Rumiati. «Insights into attention and memory difficulties in post-COVID syndrome using standardized neuropsychological tests and experimental cognitive tasks». In: *Scientific Reports* 14.1 (Feb. 2024), p. 4405. ISSN: 2045-2322. DOI: 10.1038/s41598-024-54613-9. URL: https://doi.org/10.1038/s41598-024-54613-9 (cit. on p. 5).

[8] Adam Hampshire et al. «Cognitive deficits in people who have recovered from COVID-19». In: *eClinicalMedicine* 39 (Sept. 2021). ISSN: 2589-5370. DOI: 10.1016/j.eclinm.2021.101044. URL: https://doi.org/10.1016/j.eclinm.2021.101044 (cit. on p. 5).

[9] Susanne M. Jaeggi, Martin Buschkuehl, Alex Etienne, Christoph Ozdoba, Walter J. Perrig, and Arto C. Nirkko. «On how high performers keep cool brains in situations of cognitive overload». In: *Cognitive, Affective, & Behavioral Neuroscience* 7.2 (June 2007), pp. 75–89. ISSN: 1531-135X. DOI: 10.3758/CABN.7.2.75. URL: https://doi.org/10.3758/CABN.7.2.75 (cit. on pp. 9, 10).

[10] Giorgio Ganis and Rogier Kievit. «A New Set of Three-Dimensional Shapes for Investigating Mental Rotation Processes: Validation Data and Stimulus Set». In: *Journal of Open Psychology Data* 3 (Mar. 2015). DOI: 10.5334/jopd.ai (cit. on p. 11).

[11] Aaron L. Gardony, Marianna D. Eddy, Tad T. Brunyé, and Holly A. Taylor. «Cognitive strategies in the mental rotation task revealed by EEG spectral power». In: *Brain and Cognition* 118 (2017), pp. 1–18. ISSN: 0278-2626. DOI: https://doi.org/10.1016/j.bandc.2017.07.003. URL: https://www.sciencedirect.com/science/article/pii/S0278262616302147 (cit. on p. 12).

[12] MDN Contributors. *Model-View-Controller (MVC)*. 2023. URL: https://developer.mozilla.org/en-US/docs/Glossary/MVC (cit. on p. 18).