Deliverable Number

D3.2

EMPOWER

deliverables

Deliverable name	**PLATFORM FEATURES AND ABILITIES**
Type	**DEM – Demonstrator, pilot, prototype.**
Dissemination level	**PU - Public**
Date	**Month 12**

This deliverable will reflect detailed descriptions of all the features and abilities of the platform.

–

Description

WP.3

Work Package. 3

Lead Beneficiary – (UV)

PLATFORM FEATURES AND ABILITIES

Executive Summary

This document describes recent work developed by EMPOWER partners involved in WP2 and WP3, which aims to use the definition of the games done in WP2 to develop all the features and abilities of the games platform.

Date	Version	Description	Authors
19.09.2023	1.0	Preparation of the first version of this deliverable	Lucia Vera
24.09.2023	1.1	Review and edits.	Lucia Vera, Gerardo Herrera
17.01.2023	1.2	Add platform link	Lucía Vera

Table of Content

#1. Introduction

As mentioned in other deliverables, the Empower project aims to develop different interactive games for children with neurodevelopmental disorders (NDDs) to help them reduce their emotional and behavioural problems. For that, in WP2, we were working on describing and defining the different games that will be included in the final platform. These games are based on two essential constructs: **Executive Functions** (EFs) and **Emotion Regulation** (ER) strategies.

In WP3, the team members will use the game definition to develop a fully interactive and gamified application that can keep the child completely engaged while the game is measuring and getting data about their performance, to feed the algorithms developed in WP4, to create a personalised intervention and self-adaption in the games.

In this deliverable, we will explain the most essential features and abilities of the platform being developed to support all the games and content described in WP2.

#2. Games contained in the second development cycle

The Empower project will develop an application with nine games in these executive functions and emotional regulation areas:

Executive functions:

1) Sustained Attention;
2) Working Memory;
3) Cognitive Flexibility;
4) Delayed Gratification; and
5) Behavioural Inhibition.

Emotional regulation:

6) Emotion naming
7) Emotion Intensity level rating
8) Emotion Understanding
9) Emotion Regulation Strategies

The Sustained Attention, Working Memory and Behavioural Inhibition games were developed in the prototype. In this second prototype, we added the Cognitive Flexibility and Delayed Gratification games to the platform, and with that, all the Executive functions games are completed. Also, the first Emotional regulation game is under development, the Emotion naming. These three games will be included in the second prototype and tested in this second stage of the development cycle.

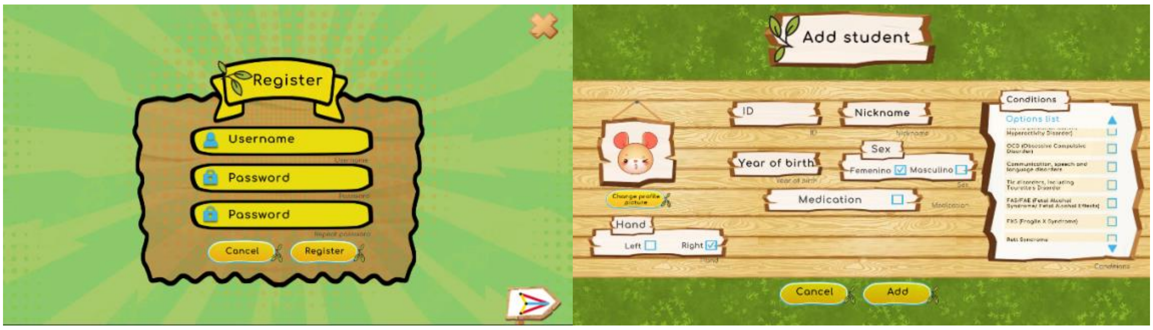
In the following sections, we will review the common features of all the games available and the platform abilities.

#3. Common features and abilities in all games

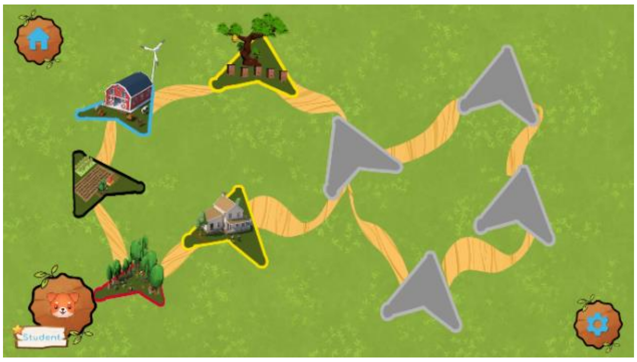
The common thread for all the games developed is the ECOFARM concept. The idea of this platform is to work in all the executive functions and emotional regulation aspects described in the previous section, but at the same time, teach important concepts to all users about the importance of ecological and sustainable behaviours, in this case, related to a farm. This critical concept is used in each game, not only as part of the descriptions and explanations but also as the main base for the content and main objectives of the game.

Technologically, the platform has several common features for all the games:

1. A primary interface connects all the games that support user creation and login, student creation and control, and game selection.



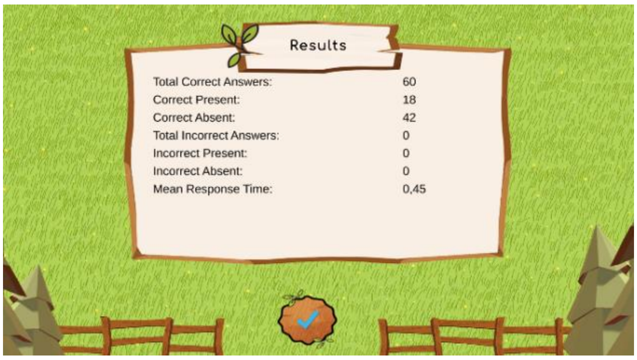
2. There is a map inside the game to connect all the areas of the farm and, therefore, all the games that will be included in the platform.



3. There is a local database where all the games can obtain information and save all the data of the user's performance inside each game. This database is also connected with the eye-tracking and wearable modules.
4. Each game has its parameters to control and measure the performance inside the game. These parameters are recorded during the game, and the final result is

saved inside the database associated with the specific user and game. All the games have as common variables the number of correct and incorrect answers, the response time for each repetition (trial) inside the game, and the mean response time. This response time is when the stimuli are presented (or question, depending on the game) until the user answers (correct or not).

5. The parameters controlled in each game are shown at the end of the game for the corresponding evaluation of the teacher.
   The algorithms can later use all these parameters to provide information or suggest adaptations inside the game to improve the performance of each specific user.



Results

Total Correct Answers:	60
Correct Present:	18
Correct Absent:	42
Total Incorrect Answers:	0
Incorrect Present:	0
Incorrect Absent:	0
Mean Response Time:	0,45

6. An eye-tracking module can be integrated into some games to allow it to follow where the user is looking during the game and save that information inside the database for its analysis and later use in the algorithms module. This module can be reviewed in detail in Deliverable D.3.3.
7. The platform also contains a module to control the user's physiological response using a wearable. This wearable is attached to the child's wrist and connected to the database to send all the data registered from the beginning until the end of the session with each child. This module can be reviewed in detail in Deliverable D.3.4.
8. All the modules are working together and, simultaneously, having as a common link the identifier of the user, session, and activity. The eye-tracking module is integrated into the activity, so you can exchange information directly with it. The wearable module works outside the main application because it is a device connected to the user's body but communicates with the main application through the database. They exchange the identifier of the active user in each moment, the session's identifier open and the ongoing activity's identifier, to link all the data registered with the correct person, session, and game.
9. There will be a common module containing the algorithms (under development) that can feed the games with information about what is wrong and what can be adapted for that specific user, also giving feedback to the teacher about which level must be selected next time the student plays the game and possible relation between the performance inside the game and the cognitive areas to work on.

We will review in detail some of these aspects in the following subsections.

DATABASE CREATION AND CONNECTION

The platform includes a connection with a local database to exchange information between modules and save all the data registered while the child plays a game. This database has been developed using the MariaDB application and a local server implemented using NodeJS. For that, the devices used for the tests must do the following steps:

- Install MariaDB database software.
- Install NodeJS software.
- Import the database schema inside the instance of MariaDB in the device.
- Install all the modules needed in the server to support the connection.
- Copy the needed js files to a local directory with all the functionality required for the platform.
- Star the server locally before starting a session.

An API is created for the modules to interact with the server and work with the database. All the database classes and entities will be described in detail in Deliverable D.3.5.

CONNECTION WITH THE EYE-TRACKING MODULE

As mentioned in the previous section, the eye-tracking module can be integrated into a game, depending on its specifications. The most important limitation of that integration is the game interaction. We are using a Tobii Pro Nano device attached to a laptop. The team selected a laptop with a touch screen to give more flexibility to the user’s interaction, making the interaction more natural for all user profiles. But this kind of interaction is an important occlusion element for the eye tracker due to the movement of the hand in front of the head, making it impossible to continuously follow the user’s eyes when the interaction is happening (that also is the most crucial moment inside the game).

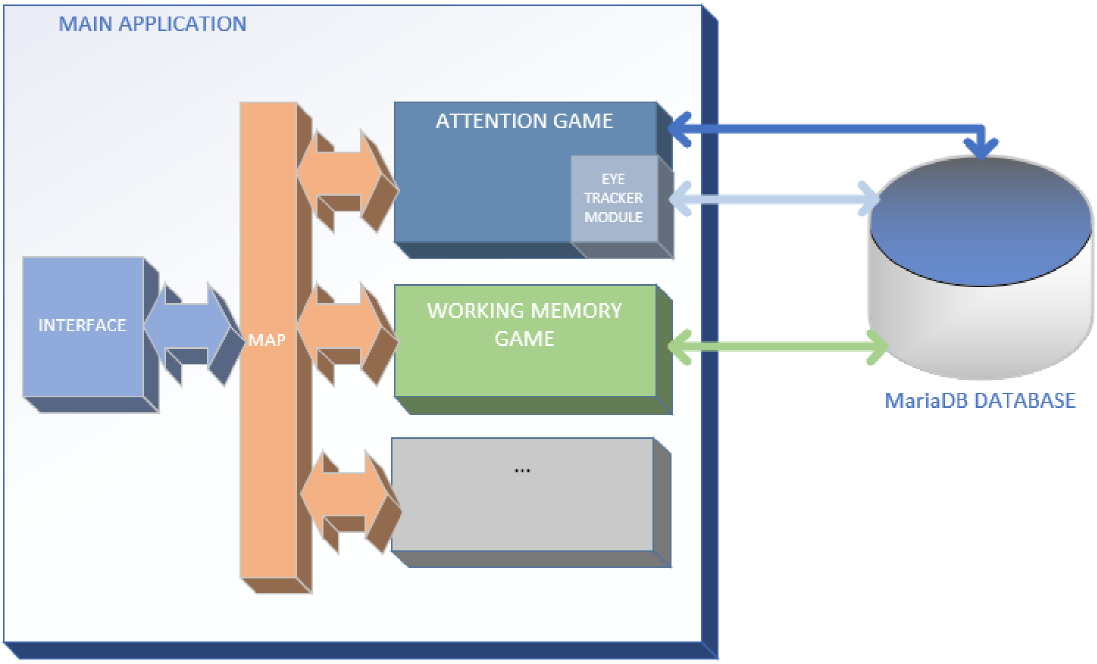
Different settings were tested to allow the use of the touch screen in combination with the eye tracker. But the most stable and suitable configuration is the one where the eye tracker is in the lower part of the screen.

For that reason, it is essential to select the integration of the eye-tracking module inside the games, where the interaction can be done only with a mouse click or a key in the keyboard. Due to the user profiles that we can have, not all children can control and move the mouse efficiently to perform complicated tasks. But most can use the mouse or the keyboard for a single click. Because of that, the first approach is to use this module only in games with single-click interaction. We will study how to improve that and integrate the eye-tracking module into more games if we can solve this interaction problem.

In those games where eye-tracking is present, all the functionality is integrated inside the game. With that, there is a calibration part for the eye tracker, then an acceptance of that calibration and the module will start recording where the user is looking inside the game. All this data is saved directly to the database, labelled with a timestamp, and used inside the application to determine which 3D visual element the child is looking at each moment.

The games save data about when the game starts and ends, using the same timestamp type as the other modules. Also, each activity registers the moment when the interaction with the game starts and ends to map data in that time slot and analyse if something different has happened.

This will give important information to the teacher to improve the performance inside the game and also to the algorithm's module to understand how to improve the game or decide which cognitive areas are essential to work more with that user.



In deliverable D3.3., it is possible to see in detail this module, the steps for the calibration inside the games and the functions needed to make it work inside the main application.

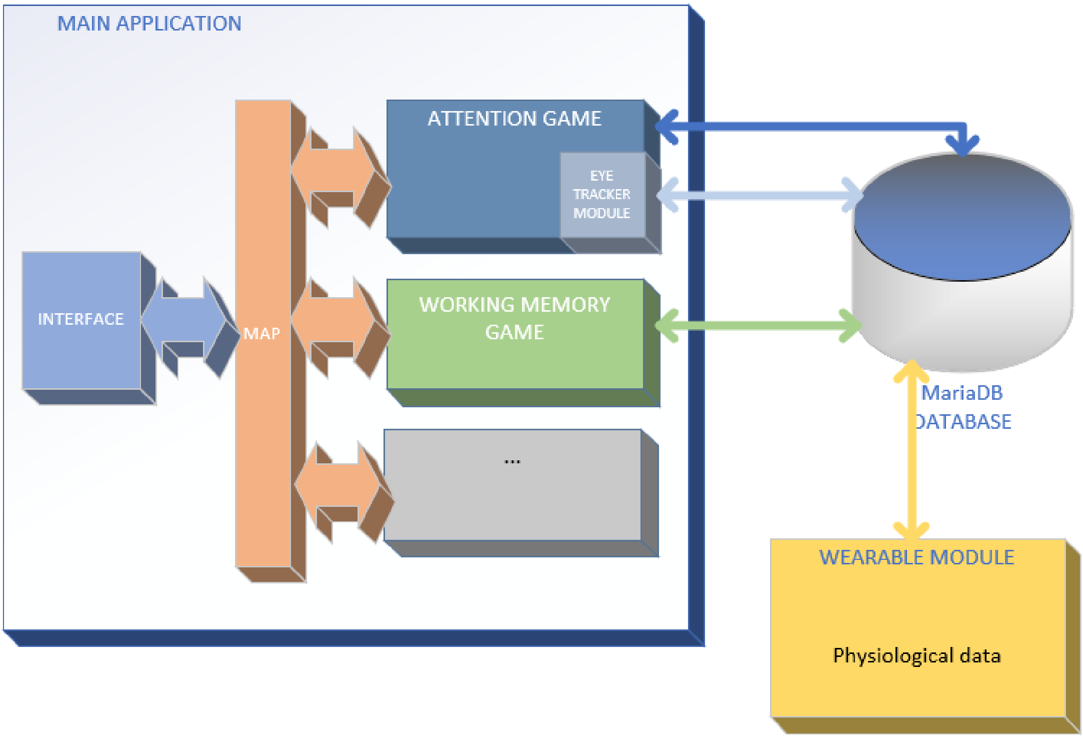
CONNECTION WITH THE WEARABLE MODULE

Another critical module is the wearable module. As the wearable device must be attached directly to the user's body, this module cannot be integrated into the main application. Therefore, the wearable has its operating system and application connected wirelessly to the database. All data registered by the device is saved directly inside the database. But for that, it is essential to keep some synchronisation. We will have the following elements that can be used later to map all the data inside the game:

- First, the module gets the student active and registers their identifier to link the data with it.
- Then, the module searches for the open session inside the database to map all the data with that session.
- Finally, the module searches for the open activity, the game active at that moment. When a game is dynamic, all the physiological data will be recorded associated with that activity and labelled with a timestamp.
- One of the main rules in this communication is that, inside that specific computer, only

one student, session and activity are open and working each time to guarantee that the data is associated with the correct student, session, and training.

The game will save data about when the activity starts and ends using the same timestamp type to allow synchronisation between the data. Also, the application records the timestamp of all the interaction slots when the user must answer, as it is an essential moment in each game. This will allow mapping of the physiological data inside the activity and inside the slot of time when the user must do the interaction. This will give us information about the child's emotional state during the game and in each moment when they must answer.



It is possible to see a complete description of the wearable module in D.3.4.

CONNECTION WITH THE ALGORITHMS MODULE

There will be a module with AI working to help the teacher decide the level of the game for the next time the student plays it and to give information for possible adaptation inside the game. For that, this module will have in the first version an offline part to use all the variables from the games, wearable and eye tracker to give feedback to the teacher for the following sessions. For this part of the module, the connection with the platform will be through the database because they will need all the information registered after the sessions.

Later, we will want to use all the information we have in real time to feed the algorithms, give feedback to the teacher while the session is playing, and adapt the games in real-time, depending on what is happening during the session. For this part, the module can be integrated inside the platform and connected directly with the games or can be connected again through the database to keep the same communication method. This part is under development.

All the essential elements of this module are described in D.4.2.

To give the algorithms as much information as possible, it is essential to register as many elements as possible inside the game to feed them complete data about what is happening in each game. For that, we will record inside the database the appearance of each essential element in the scene, the interactions of the child, the position of essential elements, and the response condition for each trial inside each game. All this data will be associated with a timestamp and can be seen as a description of what happens in each moment inside the game. This can also be later correlated with the child's physiological measurements and looking pattern.

#4. Hardware specification

For the correct usage of all the platforms, specific hardware is needed:

- Convertible Laptop HP Pavillion x360. The application will run on any Windows computer, but a laptop like this is necessary if the user can have the touch screen functionality.

- Tobii Pro Nano eye tracker to control where the child is looking.

- Samsung Galaxy Watch 5

- Internet wireless connection

#5. User profiles

Empower platform can be used by teachers, tutors or parents who want to work with children with neurodevelopmental disorders. In this case, the teacher/tutor/parent creates the account inside the application (as a child representative), only providing a username (email account) and a password. Inside the application, each user can create several students, who will be the final users who will play the games. For each student, the application asks for some specific data. Still, it is internal for the teacher/tutor/parent because all the data is anonymised for analysis purposes to keep the child's identity safe.

Each time a student plays a game, that game registers data about the performance and saves that data inside the database. These results can be used to understand better which cognitive areas it is essential to work with that child to improve their executive functions and emotional regulation strategies.

#6. Latest demonstrator

The last version of the current platform can be found at this link (version 1.6.11):

https://drive.google.com/drive/folders/1f9AwyQZQq-9Z8Q572rbIdxwwhzaxidhD?usp=drive_link

In this folder, it is possible to find the main application with the games in its current state and an EmpowerDBService.zip file with the instructions to install the local server and database to be able to play with the application and games. Also, it is possible to see the state of the application in a YouTube video following this link: https://www.youtube.com/watch?v=bOFbaOH3-1E