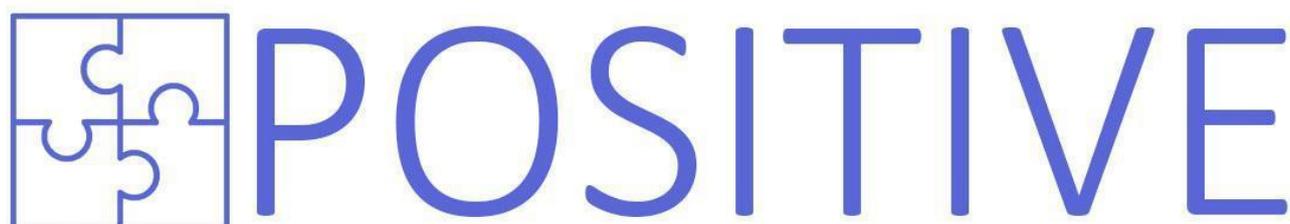


## D2.3 [System Architecture and Technical Specifications]



**Personalized platform assisting seniors in healthy, fulfilled and active life**

<b>Project No.</b>	<b>AAL-2018-2-118</b>
<b>Project Acronym</b>	<b>POSITIVE</b>
<b>Start date (duration)</b>	<b>01/05/2019</b>
<b>Deliverable Leader</b>	<b>BLP</b>
<b>Contributors</b>	<b>MAGG,REALL</b>
<b>Deliverable No.</b>	<b>2.3</b>
<b>Deliverable Title</b>	<b>System Architecture and Technical Specifications</b>
<b>Dissemination level</b>	<b>PU</b>
<b>Status</b>	<b>Final</b>
<b>Version</b>	<b>1.0.</b>
<b>Delivery date</b>	<b>31.10.2019</b>

Copyright © AAL-2018-2-118 POSITIVE Consortium

This document is proprietary of the POSITIVE Consortium. Neither this document nor the information contained herein shall be used, distributed or communicated, in any form or by any means, without the prior written consent of the POSITIVE Consortium.

## Revision History

Revision	Date	Organization(s)	Changes
Draft 1	30.10.2019	SILOR	Added architecture outline
Draft 2	31.10.2019	REALL	Consortium partner review
Draft 3	28.11.2019	MAGG	Consortium partner review
Draft 4	07.05.2020	BLP	Updated Architecture Diagram
Draft 5	14.05.2020	BLP	Analysis of Data Layer
Draft 6	21.05.2020	BLP , MAGG	Description of Functional Layer
Draft 7	28.05.2020	BLP	Updated part of data exchange and interoperability
Final	17.06.2020	BLP	Consolidated Final version

---

## Executive Summary

The present deliverable entitled “System Architecture and Technical Specifications” describes the architecture of POSITIVE platform as a complete solution for the pilot studies of the project. It is worth mentioning that the involvement of end-users from different EU countries is beneficial in predicting the evolution of needs in other countries not already represented in the consortium. The structure of the document provides an overview of the components of the system and the technologies/methodologies used to succeed the integration of them into a robust and fully functional solution.

Input from D2.1 “Co-Design and Requirements Analysis” has been used as a base to justify the specific needs of all stakeholders as well as on the evaluation of early design mock-ups presented to the end users and input from D2.2 “Report on End User Recruitment and Mobilization” has been used in order to receive an appropriate report for the mobilization and recruitment of end users for each period. The use case scenarios and the preliminary feedback received by the end users has been taken into consideration to adjust the modules and functionalities of the platform. Regarding the technical architecture and related specifications, detailed information about the POSITIVE system modules, including how they will be supported by the POSITIVE unified integrated cloud-based and smartphone platform, are presented in section 3.2 of this deliverable.

The architecture described in the document has been discussed and agreed between the partners, although, some deviations may exist in the final implementation of the system, due to the dynamic nature of requirement changes. Any major deviations will be described and justified in the deliverables of WP3 “POSITIVE platform”, and specifically in T3.5 “System Integration and Technical Testing”, where the integrated solution will be described.

## Table of Contents

<b>1.</b>	<b>REVISION HISTORY</b>	<b>2</b>
<b>2.</b>	<b>EXECUTIVE SUMMARY</b>	<b>3</b>
<b>3.</b>	<b>INTRODUCTION</b>	<b>6</b>
<b>3.1</b>	<b>ROLE OF THE DELIVERABLE</b>	<b>6</b>
<b>3.2</b>	<b>RELATIONSHIP TO OTHER DELIVERABLES</b>	<b>6</b>
<b>4.</b>	<b>OVERALL ARCHITECTURE</b>	<b>7</b>
<b>4.1</b>	<b>OVERVIEW</b>	<b>7</b>
<b>4.2</b>	<b>ARCHITECTURE DIAGRAM</b>	<b>7</b>
<b>5.</b>	<b>END USERS GROUPS</b>	<b>9</b>
<b>6.</b>	<b>POSITIVE DATA LAYER</b>	<b>11</b>
<b>6.1</b>	<b>SUPPORTED FUNCTIONALITY</b>	<b>12</b>
<b>6.2</b>	<b>TECHNICAL FUNCTIONALITY</b>	<b>13</b>
<b>7.</b>	<b>POSITIVE FUNCTIONAL LAYER</b>	<b>17</b>
<b>7.1</b>	<b>POSITIVE EVENTS AND FACILITATION MODULE</b>	<b>21</b>
<b>7.1.1</b>	<b>SUPPORTED FUNCTIONALITY</b>	<b>22</b>
	<i>7.1.2 Technical functionality</i>	<i>22</i>
	<i>7.1.3 Data exchange and Interoperability</i>	<i>25</i>
<b>7.2</b>	<b>POSITIVE WELLBEING TRACKER MODULE</b>	<b>26</b>
<b>7.2.1</b>	<b>SUPPORTED FUNCTIONALITY</b>	<b>27</b>
	<i>7.2.2 Technical functionality</i>	<i>29</i>
	<i>7.2.3 Data exchange and Interoperability</i>	<i>29</i>
<b>7.3</b>	<b>POSITIVE PRACTICAL INFORMATION AND E-LEARNING MODULE</b>	<b>30</b>
	<i>7.3.2 Technical functionality</i>	<i>31</i>
	<i>Moodle structure</i>	<i>33</i>
	<i>7.3.3 Data exchange and Interoperability</i>	<i>34</i>
<b>7.4</b>	<b>POSITIVE COLLABORATION &amp; GAMING MODULE</b>	<b>35</b>
	<i>7.4.2 Technical functionality</i>	<i>36</i>
	<i>7.4.3 Data exchange and Interoperability</i>	<i>36</i>
<b>8.</b>	<b>INTEGRATION PLAN</b>	<b>37</b>
<b>8.1</b>	<b>OVERVIEW</b>	<b>37</b>
<b>8.2</b>	<b>PURPOSE</b>	<b>37</b>
<b>8.3</b>	<b>INTEGRATION STRATEGY</b>	<b>37</b>
<b>9.</b>	<b>DEPLOYMENT</b>	<b>39</b>
	<b>MYSQL</b>	<b>39</b>
	<b>APACHE WEB SERVER</b>	<b>39</b>
	<b>PHP</b>	<b>39</b>
	<b>WORDPRESS</b>	<b>39</b>
	<b>MOODLE</b>	<b>39</b>
<b>10.</b>	<b>CONCLUSIONS</b>	<b>40</b>
<b>11.</b>	<b>REFERENCES</b>	<b>41</b>



## 1. Introduction

### 3.1 Role of the deliverable

This document is part of WP2 “End user involvement/co-creation” and describes the reference architecture of the solution provided by the POSITIVE project. The deliverable emphasizes on outlining the technical architecture of the POSITIVE platform, resulting for the initial identification of user’s needs. The specification of the technical architecture includes a detailed analysis of the modules (including third-party modules) comprising the platform, as well as of the interfaces between them. The architecture provides guidelines for the technical integration of the various modules into a robust and coherent platform able to respond to user needs. Current deliverable contains information related to the software design and security prerequisites of the system, along with data structures and data contracts for the communication of the sub-components of the platform.

The key inputs taken into account in this deliverable are: a) the description of the components of POSITIVE platform as described in the Description of Action (DOA) of the POSITIVE project and b) the requirements and use cases for the POSITIVE platform. The deliverable has been produced by adopting an iterative development approach. In this approach, the partners responsible for key components of the POSITIVE platform took responsibility for the development of the specification of their interfaces and the identification of the requirements which the component should contribute to the realization of. The overall architecture of the platform was specified during consortium meetings that considered in detail dependencies between components and the ways in which they could be addressed through the definition of proper component structures and interactions.

The architecture of the platform will be specified using the Unified Modelling Language (UML), such as activity diagrams, component interaction diagrams and/or activity diagrams in a later on updated version of the deliverable. It should be noted that although we have tried to provide a comprehensive specification of POSITIVE platform architecture, it is inevitable that the architecture presented in this deliverable will be subject to changes during the implementation. Our expectation is that deviations from the architecture as specified in this document will be minimal.

### 3.2 Relationship to other deliverables

This deliverable is related to D2.1 “Co-Design and Requirements Analysis”, which provides the requirements of the users, and input from D2.2 “Report on End User Recruitment and Mobilization” which describes the involvement of the end users of the system.

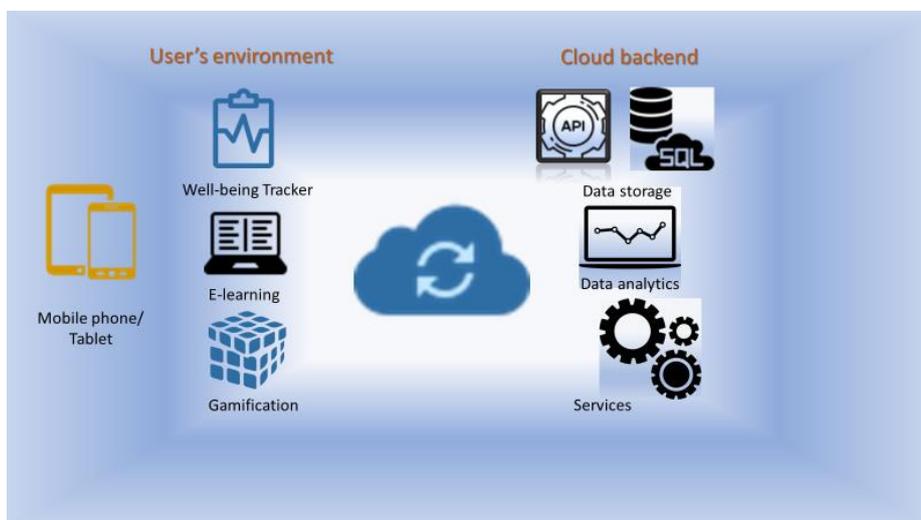
## 2. Overall Architecture

### 4.1 Overview

The details of POSITIVE architecture are summarized in the following sections. The architecture diagrams and flows are provided, along with details for the system modules. Dependencies between the components are described (input and output contracts), along with the flow of the information and the interaction between the components.

### 4.2 Architecture diagram

An outline of the architecture is presented in Figure 2. As illustrated, POSITIVE follows a layered approach which allows it to be modular and expandable. The bottom layer will constitute the POSITIVE Data Layer, which comprises of a set of modules that will host amongst others the user and event facilitator profiles and the corresponding management mechanisms, articles and posts and the training courses that will be delivered through the platform. The Data Layer will constitute the core layer through which the information exchange between the various layers and modules will take place. On the top of that, there is a Functional Layer, which consists of interconnected modules defining functionalities in different parts of the platform explained before. All the modules can be developed separately, some contain repeated functionalities that can be easily reused. On top of these layers, POSITIVE will incorporate the UI Layer which will consist of all parts that make it possible for the user to actually access, explore and engage with the platform’s content. POSITIVE adopts a cross-platform design in order to be accessible through the devices used in the project (mobile phone and tablet).



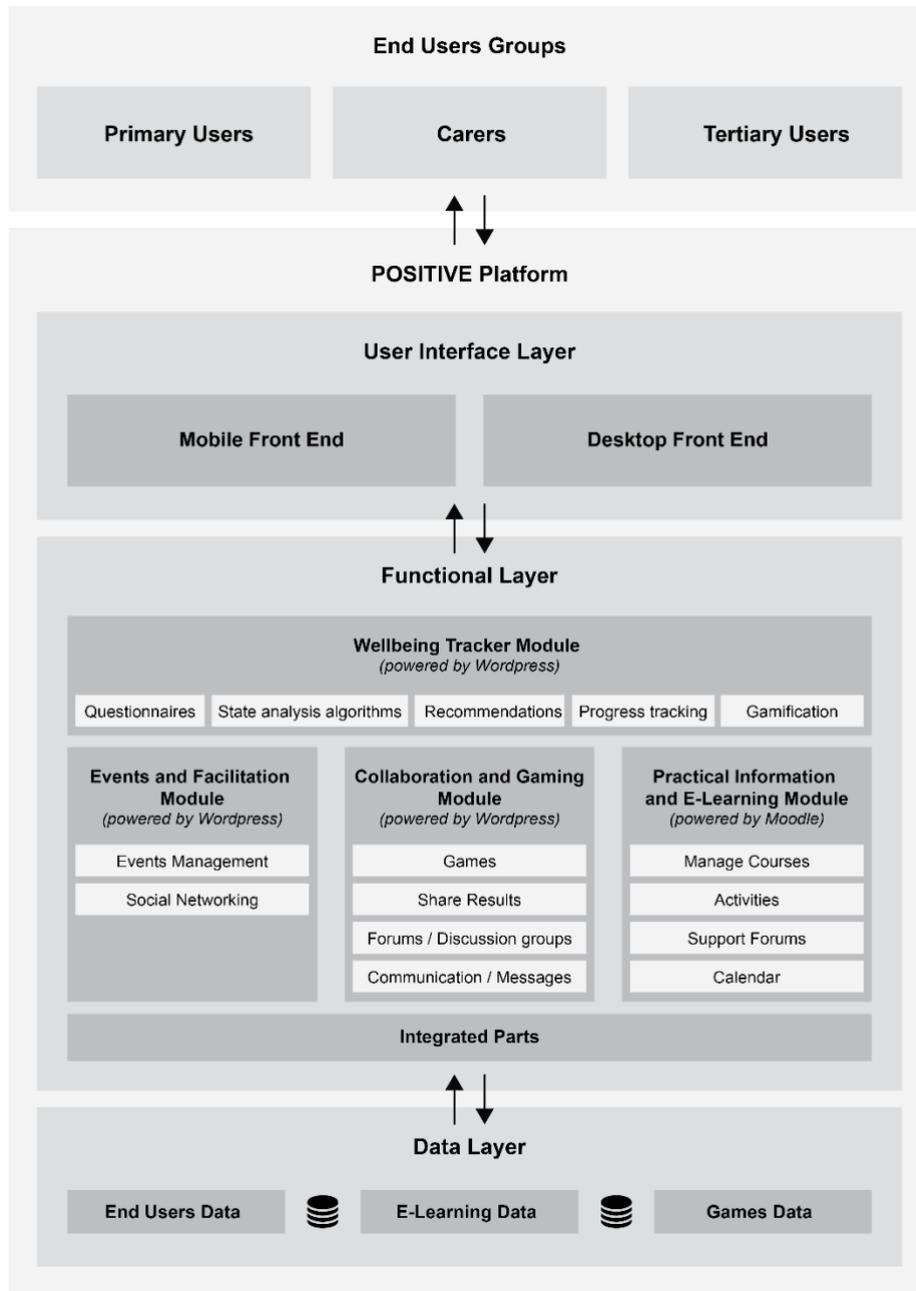
**Figure 1: Main concept and interactions**

The architecture can be decomposed into three layers, as following (Figure 2):

- The POSITIVE’s Data Layer which comprises of the modules that will host the user and event facilitator profiles and the corresponding management mechanisms and content that will be delivered through the platform.

- 
- The Functional Layer consists of many interconnected modules that are defining the functionalities of the different parts of the platform.
  - The User Interface Layer will be designed and implemented in a way to allow web and mobile interaction, accessible from all devices, providing a unique digital experience.

The initial conceptual architecture of POSITIVE includes the primary, secondary (carers) and tertiary users and their interaction with the User Interface layer as well as the Functional layer and the Data layer. The primary end users are active 55+ people split in two groups (minor experienced with Internet use and unexperienced), the secondary end users are their families, acquaintances, informal caregivers and the tertiary users are the organizations providing the events and content of POSITIVE.



**Figure 2: POSITIVE Overall Architecture**

### 3. End users groups

The end users will be involved in the evaluation of the project, regarding the design and requirements analysis and confirmation, the Evaluation of Pilot Operations and the continuous feedback loops validating and verifying the progress of the development work. In addition there will be involvement in the acquisition of champions, fans and co-creators and users training and supervision.

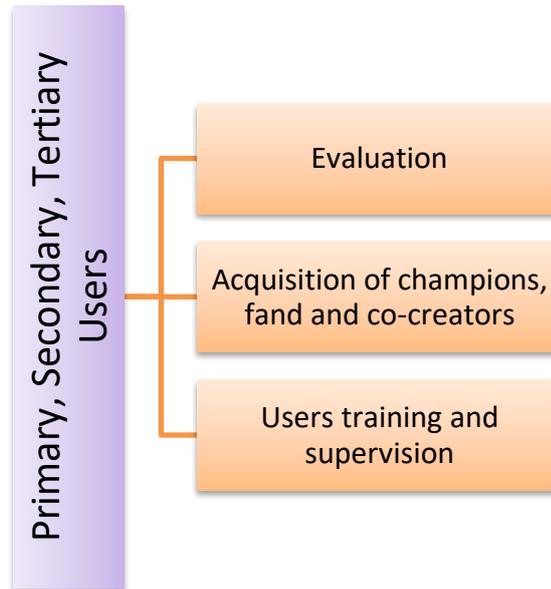


Figure 3: End user involvement

Primary end users will interact directly with the Front End, the User interface layer, either through a desktop or mobile device. From there they can have access to a series of sub-modules such as accessibility, personalization, gamification, smart notifications and collaboration module.

Secondary users, such as family, acquaintances, informal caregivers will interact with the functional layer, in terms of sharing the platform, via the social media.



Figure 4: Primary End user involvement

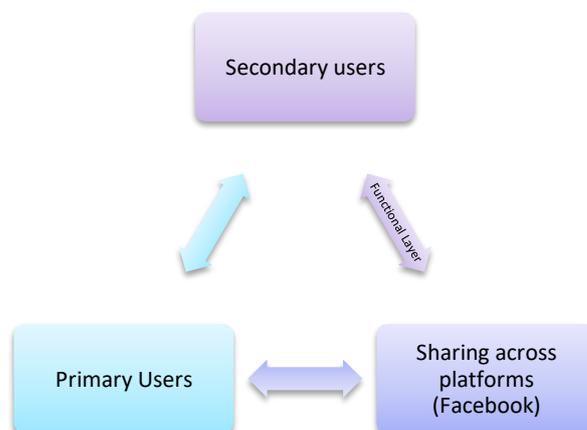


Figure 5: Secondary users involvement within the functional layer

Tertiary users, will have indirect access to the platform in terms of content / event provision and management. The following infographics depicts how each user of POSITIVE interacts with the platform (Figure 5).



**Figure 6: User Roles within the POSITIVE solution**

#### 4. POSITIVE Data Layer

In this section (section 6) we analyze the POSITIVE Data Layer. This layer is responsible for providing access to stored data required for the upper layers to be functional. The corresponding section of the holistic platform architecture is highlighted within the borders of the red box as illustrated in Figure 7.

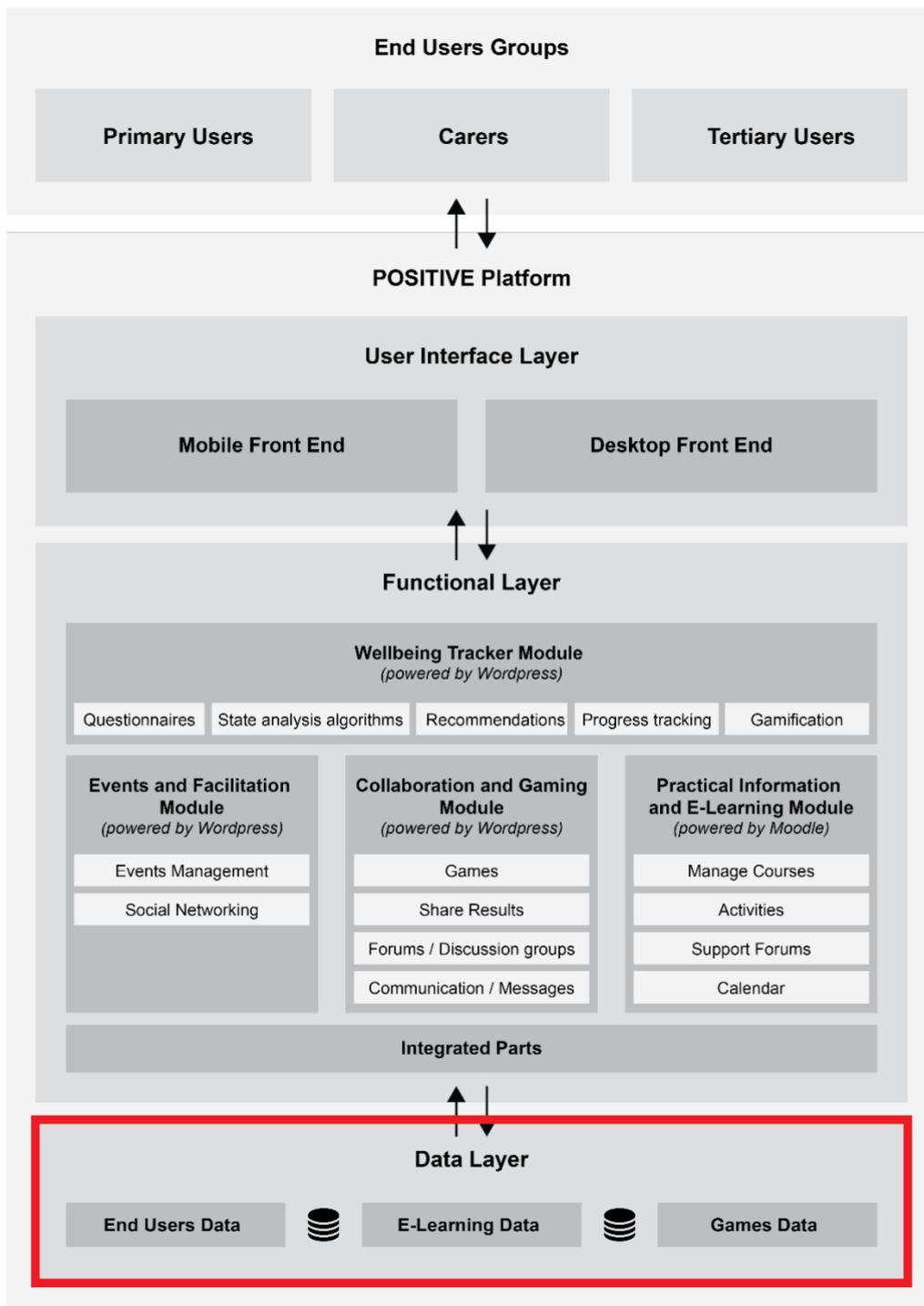


Figure 7: POSITIVE platform Data Layer

## 6.1 Supported functionality

This layer (Data layer) will be responsible to store all the persistent data shared between applications and services of the POSITIVE platform. This includes the following information:

- End user information used to authenticate the users,
- E-learning and courses data
- Content related meta-data defining the categorization of the material

The Data Layer will be the storage facility for storing user-generated content and the data required for the operation of the applications of the functional layer.

This layer is responsible for performing the actual data I/O operations for a database as well as enabling and enforcing certain feature sets that target a specific application need. A major benefit of using specific storage engines is that you are only delivered the features needed for a particular application, and therefore you have less system overhead in the database, with the end result being a more efficient and higher database performance.

Incorporated components of this layer include the Functional Layer Database, comprising from End users, e-Learning and Gaming data, which refer to the relational database management system (RDBMS).

## 6.2 Technical Functionality

In this section the analysis of the technical functionality of the POSITIVE Data Layer is performed.

### MySQL Features

The Data Layer of POSITIVE platform will be heavily based on MySQL database. MySQL supports various storage engine infrastructure components which will be utilized during the project. Below we summarize some of its key features that are going to be utilized within POSITIVE implementation [1]:

- **Concurrency:** *Some applications have more granular lock requirements (such as row-level locks) than others. Choosing the right locking strategy can reduce overhead and therefore improve overall performance. This area also includes support for capabilities such as multi-version concurrency control or “snapshot” read.*
- **Transaction Support:** *Not every application needs transactions, but for those that do, there are very well defined requirements such as ACID compliance and more.*
- **Referential Integrity:** *The need to have the server enforce relational database referential integrity through DDL defined foreign keys.*
- **Physical Storage:** *This involves everything from the overall page size for tables and indexes as well as the format used for storing data to physical disk.*
- **Index Support:** *Different application scenarios tend to benefit from different index strategies. Each storage engine generally has its own indexing methods, although some (such as B-tree indexes) are common to nearly all engines.*
- **Memory Caches:** *Different applications respond better to some memory caching strategies than others, so although some memory caches are common to all storage engines (such as those used for user connections or MySQL's high-speed Query Cache), others are uniquely defined only when a particular storage engine is put in play.*
- **Performance Aids:** *This includes multiple I/O threads for parallel operations, thread concurrency, database checkpointing, bulk insert handling, and more.*
- **Miscellaneous Target Features:** *This may include support for geospatial operations, security restrictions for certain data manipulation operations, and other similar features.*

Each set of the pluggable storage engine infrastructure components are designed to offer a selective set of benefits for a particular application. Conversely, avoiding a set of component features helps reduce unnecessary overhead. It stands to reason that understanding a particular application's set

of requirements and selecting the proper MySQL storage engine can have a dramatic impact on overall system efficiency and performance.

### **MySQL pluggable storage engine**

MySQL Server uses a pluggable storage engine architecture that enables storage engines to be loaded into and unloaded from a running MySQL server.

The MySQL pluggable storage engine architecture enables a database professional to select a specialized storage engine for a particular application need while being completely shielded from the need to manage any specific application coding requirements. The MySQL server architecture isolates the application programmer and DBA from all of the low-level implementation details at the storage level, providing a consistent and easy application model and API. Thus, although there are different capabilities across different storage engines, the application is shielded from these differences.

The pluggable storage engine architecture provides a standard set of management and support services that are common among all underlying storage engines. The storage engines themselves are the components of the database server that actually perform actions on the underlying data that is maintained at the physical server level.

This efficient and modular architecture provides huge benefits for those wishing to specifically target a particular application need—such as data warehousing, transaction processing, or high availability situations—while enjoying the advantage of utilizing a set of interfaces and services that are independent of any one storage engine.

The application programmer and DBA interact with the MySQL database through Connector APIs and service layers that are above the storage engines. If application changes bring about requirements that demand the underlying storage engine change, or that one or more storage engines be added to support new needs, no significant coding or process changes are required to make things work. The MySQL server architecture shields the application from the underlying complexity of the storage engine by presenting a consistent and easy-to-use API that applies across storage engines.

The POSITIVE platform modules of the Functional Layer will make an extensive use of the MySQL Database server in order to store data related to the use of each one of the modules.

## Deployment

The Data Layer will be deployed on a Linux server running Ubuntu Server. Functional layer will be using this Layer for data persistence.

Below you can find the initial schemes of the WordPress and the Moodle databases that will be used in the context of the Functional Layer.

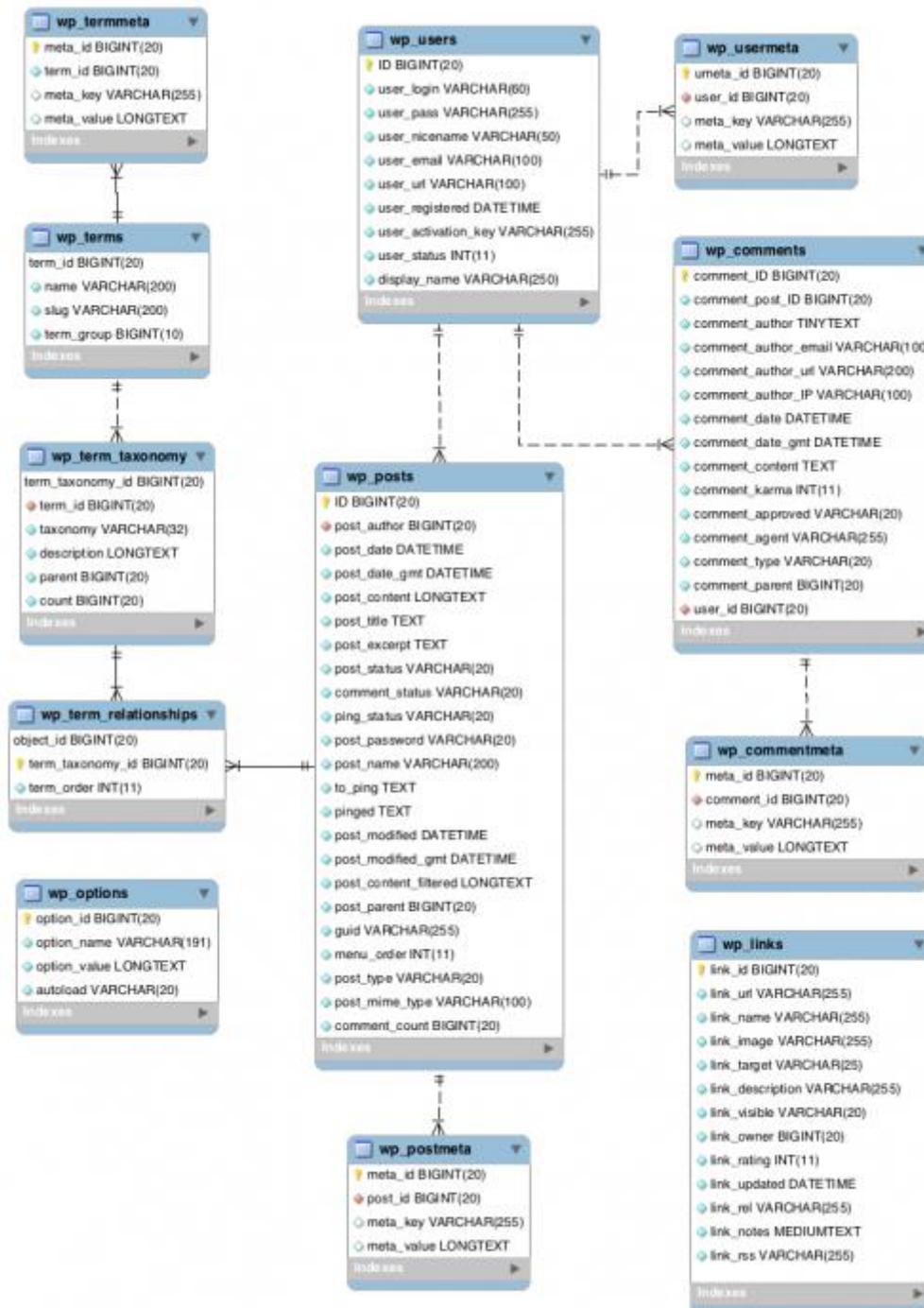


Figure 8: Wordpress Database Schema

Database chosen for Moodle is MariaDB, a fork of MySQL more suitable for Unix systems.

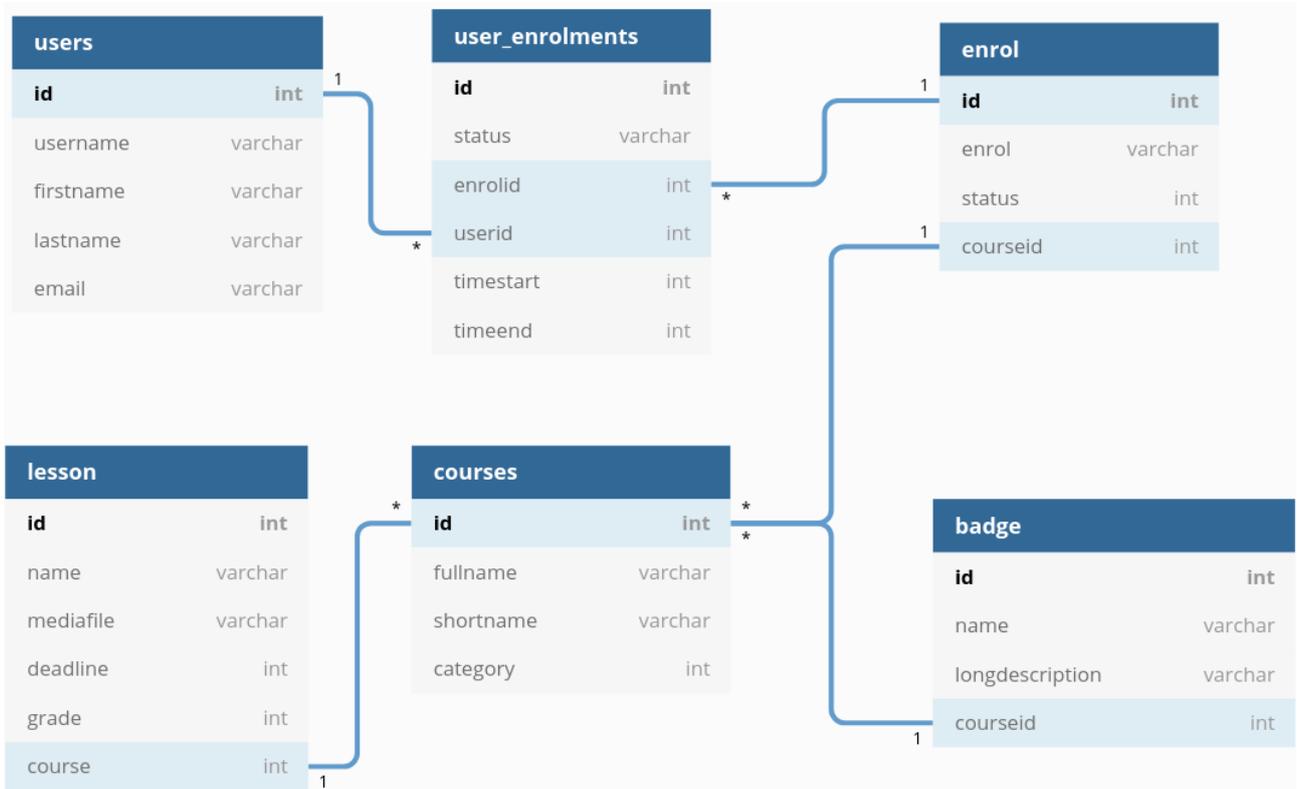


Figure 9: Moodle Database Schema

## 5. POSITIVE Functional Layer

In this section (section 7) we analyze the POSITIVE Functional Layer. This layer is responsible for providing all the required services to the POSITIVE platform user groups. POSITIVE system modules corresponds to the different services to be offered and validated in the scope of the project, including how they will be supported by the POSITIVE unified integrated platform.

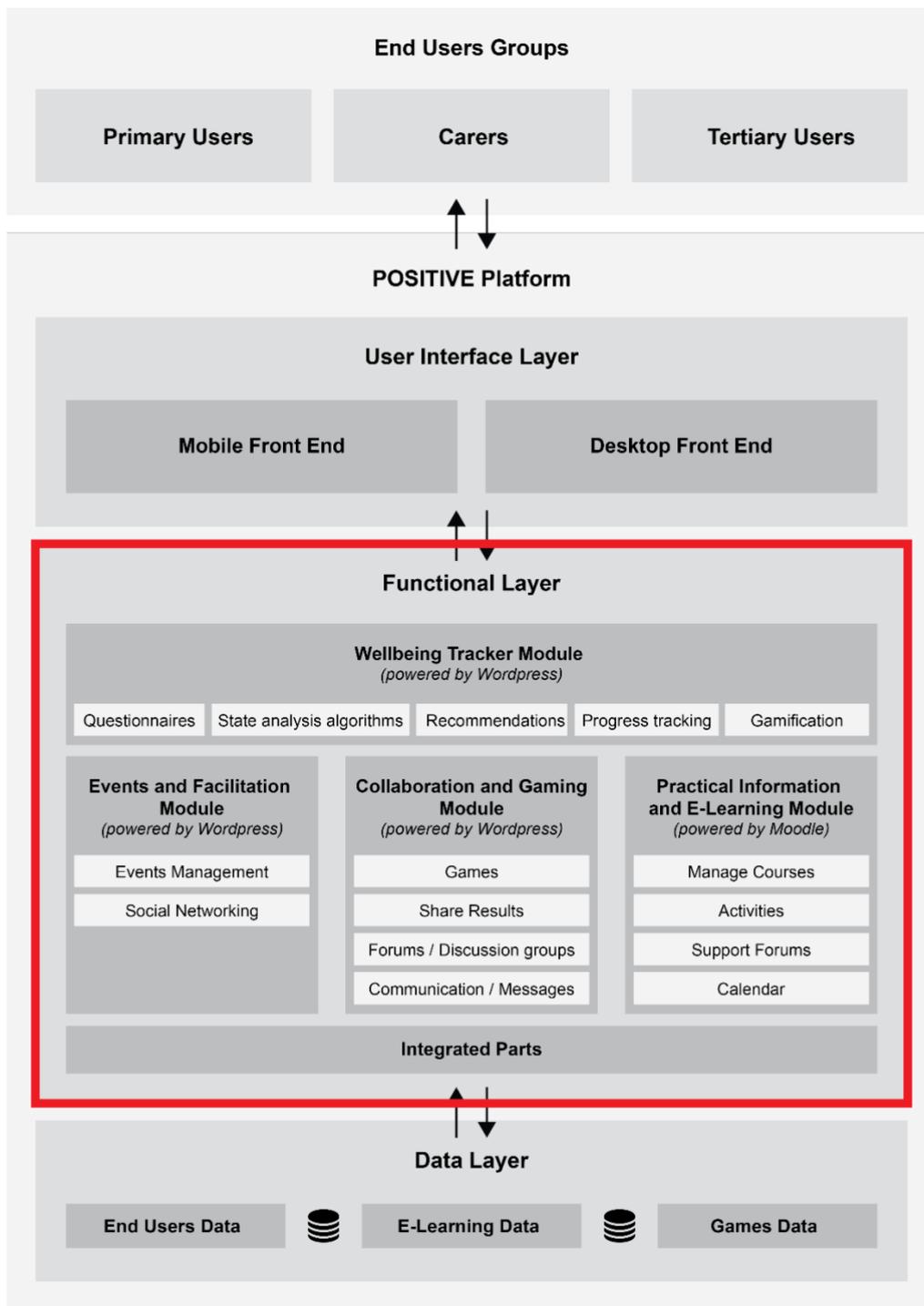


Figure 10: POSITIVE Functional Layer

The modules of POSITIVE functional layer are the following:

- 1) Service / Module I: Events and Facilitating module
- 2) Service / Module II: Wellbeing Tracker module
- 3) Service/Module III: Practical Information and E-learning module
- 4) Service / Module IV: Collaboration & Gaming Module

After analyzing the input of requirements analysis we proceeded with a prioritization of the features requested per module. You can see analysis on the table below.

Functionalities mentioned in the proposal	Suggested adjustments based on user interviews and research during the project
<b>Events facilitating module</b>	
<ul style="list-style-type: none"> <li>- display of all events personalized to the user</li> <li>- search engine</li> <li>- events being able to be created by seniors</li> </ul>	<ul style="list-style-type: none"> <li>- display of all events personalized to the user</li> <li>- search engine</li> <li>- events being able to be created by seniors</li> <li>- possibility of reporting an abusive content</li> <li>- preliminary verification of abusive content</li> </ul>
<ul style="list-style-type: none"> <li>- events being able to be created by third party organizations (different interface)</li> </ul>	<ul style="list-style-type: none"> <li>- events being able to be created by third party organizations (different interface)</li> </ul>
<ul style="list-style-type: none"> <li>- logistics help (Uber integration)</li> </ul>	<p><i>Deprioritized - not necessary to fulfill the holistic idea</i></p>
<ul style="list-style-type: none"> <li>- one-click payment</li> </ul>	<p><i>Deprioritized - not necessary to fulfill the holistic idea</i></p>
<ul style="list-style-type: none"> <li>- weather condition integration</li> </ul>	<p><i>Deprioritized - not necessary to fulfill the holistic idea</i></p>
<ul style="list-style-type: none"> <li>- calendar</li> </ul>	<p><i>Deprioritized - not necessary to fulfill the holistic idea</i></p>
<b>Practical Information and E-learning module</b>	
<ul style="list-style-type: none"> <li>- participating in e-learning courses</li> </ul>	<ul style="list-style-type: none"> <li>- participating in e-learning courses</li> </ul>
<ul style="list-style-type: none"> <li>- quizzes, support sessions, planning of learning activities</li> </ul>	<ul style="list-style-type: none"> <li>- quizzes, support sessions, planning of learning activities</li> </ul>
<ul style="list-style-type: none"> <li>- giving seniors ability to create content</li> </ul>	<ul style="list-style-type: none"> <li>- giving seniors ability to create content</li> </ul>
<ul style="list-style-type: none"> <li>- giving tertiary users ability to add content (different interface)</li> </ul>	<ul style="list-style-type: none"> <li>- giving tertiary users ability to add content (different interface)</li> </ul>
<ul style="list-style-type: none"> <li>- practical information services/webinars ("cook dumplings with me" or "fill tax statement with me")</li> </ul>	<ul style="list-style-type: none"> <li>- practical information services/webinars ("cook dumplings with me" or "fill tax statement with me")</li> </ul>
<ul style="list-style-type: none"> <li>- webinars for young people that requested help in a certain topic - connecting seniors with youngsters</li> </ul>	<p><i>Deprioritized - not necessary to fulfill the holistic idea</i></p>

<b>Collaboration &amp; Gaming Module</b>	
- playing games alone or through video chat with others	- playing games alone or through video chat with others - possibility to share results with others - increasing difficulty level
- forum groups of common interest - discussion rooms (Philosophy, Psychology, Politics, Technology, etc.)	- forum groups of common interest - discussion rooms (Philosophy, Psychology, Politics, Technology, etc.) - the interface preferably should be similar to other well-known communicators like Facebook
- a place to share one-liners (e.g., mix salt with lemon juice to get rid of a coffee stain) and be able to add picture, send to family	<i>Deprioritized - not necessary to fulfill the holistic idea</i>
- a place to gather towards a goal e.g., signing a petition	<i>Deprioritized - not necessary to fulfill the holistic idea</i>
<b>Wellbeing Tracker module</b>	
- tracking measures, like happiness, wellbeing, fresh air intake, good diet, physical activity, medicine intake, regular doctors' visits - inputted by the senior through a gamified experience	- tracking measures, like happiness, wellbeing, fresh air intake, good diet, physical activity, medicine intake, regular doctors' visits - inputted by the senior through a gamified experience - recommendations of certain activities in each module based on the users' input given while setting up the account - progress tracking (the most feasible options, like "your reaction time has increased by ...")
- state analysis algorithm	- state analysis algorithm
- gamifying the experience with scoring points and recommendation based on constantly learning analysis of user inputs	- gamifying the experience with scoring points and recommendation based on constantly learning analysis of user inputs
- integration with smart devices, health centers	<i>Deprioritized - not necessary to fulfill the holistic idea</i>
- fortune cookie (horoscope or proverb)	<i>Deprioritized - not necessary to fulfill the holistic idea</i>
<b>Integrated parts</b>	
- signing up, logging	- signing up, logging
- user membership payments	- user membership payments
- password reminder	- password reminder
- clear and visible font	- clear and visible font

	- just enough space to fill the text gaps, large blank fields intimidate seniors
	- a tutor icon at every step with manual in a balloon help (basically a friendly help button that imitates a real human)
	- maximum simplicity (preferably drop-down lists, multiple-choice questions rather than open text fields)

**Table 1: Updates on functionality after analysis of user requirements**

Each one of the abovementioned services/modules are described in the following sections.

## 7.1 POSITIVE Events and Facilitation module

The events and facilitation module provides the necessary mechanisms for the POSITIVE platform users to manage events. Both user groups (primary end users and tertiary users) can create events and include specific attributes.

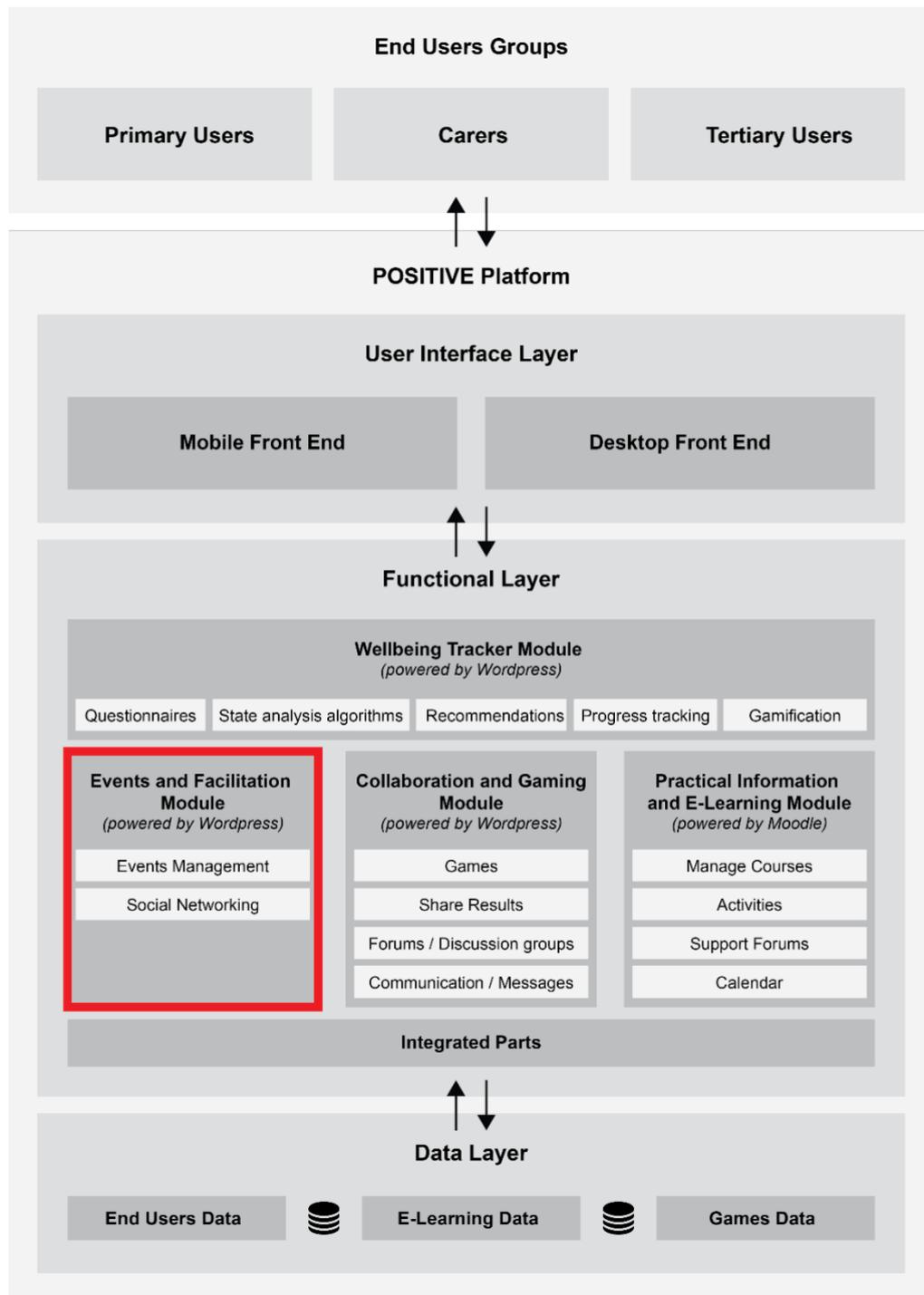


Figure 11: POSITIVE Events and Facilitation Module

### 7.1.1 Supported Functionality

More specifically through this module the following functionalities will be available:

#### Events management

Through this module both primary end users and tertiary users will be able to create events and ask people to join. These events can be physical or virtual ones. The organizer will be able to share information about the event. The users that would like to participate can indicate that they will join and also ask/propose to their friends to join as well.

Events Attribute
Title
Description
Start time
Approximate End Time
Location
Users Invited
Users Going
Category
Tags

**Table 2: Events Attributes**

Each user will be prompted to join suggested events according to his physical and mental profile. The users of POSITIVE platform will be also able to report the use of abusive content to the POSITIVE platform Administrators.

#### Social Networking

The necessary mechanisms will be implemented to enable the users to create connections with other users of the platform. In that way they will be enabled to communicate and invite them to events they joined.

### 7.1.2 Technical functionality

The implementation of the Events facilitating module be based on the well-known, free and open source platform WordPress. WordPress includes a variety of features that make it efficient, fast and easy to use [2] and follows a layers approach. The layered structure is presented below on figure 7.

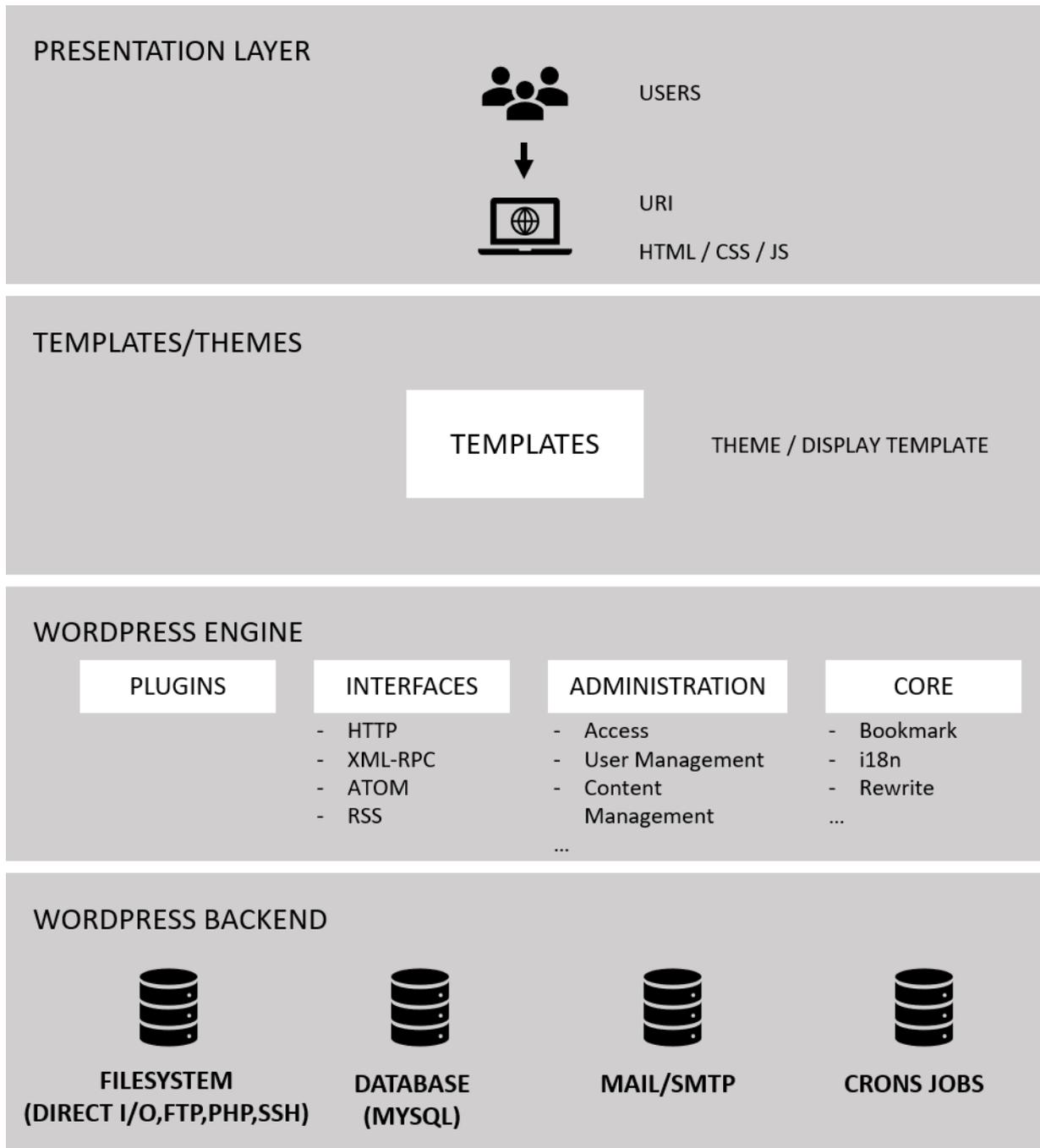


Figure 12: WordPress Layered Structure

Beginning from the WordPress Back-End layer (at the bottom) it included the filesystem and the database parts. On the next layer we have the WordPress Engine including: a) the plugins that are integrated within the WordPress installation, b) the interfaces, c) the administration as well as the other core services provided (e.g. user and content management). On the templates and themes layers we have the design of the look and feel to be located. Last on the actual presentation layer provided through a web interface to the users.

### WordPress API Specifications

The WordPress API stands for the WordPress Application Programming Interface. It can be separated into multiple API sections/topics. Each covers the functions involved in and use of a given set of functionality. Together they form what might be called the WordPress API, which is the plugin/theme/add-on interface created by the entire WordPress project. The WordPress APIs make it possible to create plugins to extend WordPress. WordPress' extensibility lies in the thousands of hooks at the disposal of developers. The WordPress API can be separated into multiple API sections/topics. Each covers the functions involved in and use of a given set of functionality. Such sections include but are not limited to: Database API, HTTP API, XML-RPC WordPress API and more.

Below is a list of WordPress APIs:

Dashboard Widgets	Database	HTTP	File Header
Filesystem	Metadata	Options	Plugin
Quicktags	Rewrite	Settings	Shortcode
Theme Modification	Theme Customization	Transients	Widgets

**Table 3: Events Attributes**

For instance, the Database API includes three separate APIs: Options API, Transients API and Metadata API. The Options API is a simple and standardized way of storing data in the database. The API makes it easy to create, access, update, and delete options. All the data is stored in the wp\_options table under a given custom name.

Options API ADD/DELETE OPTIONS:

- add\_option()
- delete\_option()
- add\_site\_option()
- delete\_site\_option()

Options API GET/UPDATE OPTIONS:

- get\_option()
- update\_option()
- get\_site\_option()
- update\_site\_option()

Example:

```
// Create an option to the database
add_option( $option, $value = , $deprecated = , $autoload = 'yes' );
// Removes option by name.
delete_option( $option );
// Fetch a saved option
get_option( $option, $default = false );
// Update the value of an option that was already added.
update_option( $option, $newvalue );
```

### Core Structure

The core structure of a WordPress site represents the main containers which hold the page's content. The core structure of a WordPress site features, at a minimum, are i) Header, ii) Sidebar / Menu, iii) Content, and iv) Footer.

### Interior Structures

Within these core structural containers are smaller building blocks that hold the specific content within the parent container. The header is the structure that traditionally sits at the top of a web page. It contains the title of the website. It may also be referred to as the masthead, head, title, and banner. In all WordPress Themes, the header is found within the header.php template file. The header will be used to host the sub-menu of the WordPress installation for the POSITIVE platform. This may contain links-pages about contact information or newsfeed.

The sidebar, in general, can be simple or complex. WordPress Themes often feature information within the sidebar in nested lists. The sidebar will be reserved for placing categories or/and sub-categories of the information within the POSITIVE platform.

The Pages and Links category, labelled "Blogroll", uses the `<?php get_links_list(); ?>` and `<?php wp_list_pages(); ?>` template tags which automatically generates a heading. The other sidebar section titles, categories, archives, meta, and others, do not use template tags which generate their own titles. These are set inside of PHP statements which "print" the text on the page. While these could be put inside of heading tags, WordPress uses the `_e()` function to display or "echo" the text titles while also marking the text as a possible target for language translation.

The content container in WordPress plays the most important role. It holds the WordPress Loop which dictates the generation of content on the page depending upon the request by the user.

The search form is found within the searchform.php. It may be found in different locations within the sidebar. To style the overall search form, use the search ID. Search is vital for the POSITIVE platform. Via search, the user will be able to locate information quicker which otherwise he might have missed.

The footer is found within the footer.php template file. The footer will be designed so that the user will have an overall view of the site structure and the categories of the information available.

### 7.1.3 Data exchange and Interoperability

Events and Facilitation module	
<b>Input Data</b>	<ul style="list-style-type: none"> <li>● Configure Events</li> <li>● Configure Social Networking</li> <li>● Invite Friends</li> <li>● Report Events</li> </ul>
<b>Output Data</b>	<ul style="list-style-type: none"> <li>● List of available events</li> <li>● List of events in which a user is enrolled</li> <li>● Information about an event</li> <li>● List of friends of a user</li> </ul>

## 7.2 POSITIVE Wellbeing tracker module

Below we analyze the POSITIVE Wellbeing tracker module. The wellbeing tracker module will be responsible for tracking the state of the user, both for mental and physical parameters and provide suggestions to the users in terms of platform content. These suggestions can include events to participate, courses to follow and games to play.

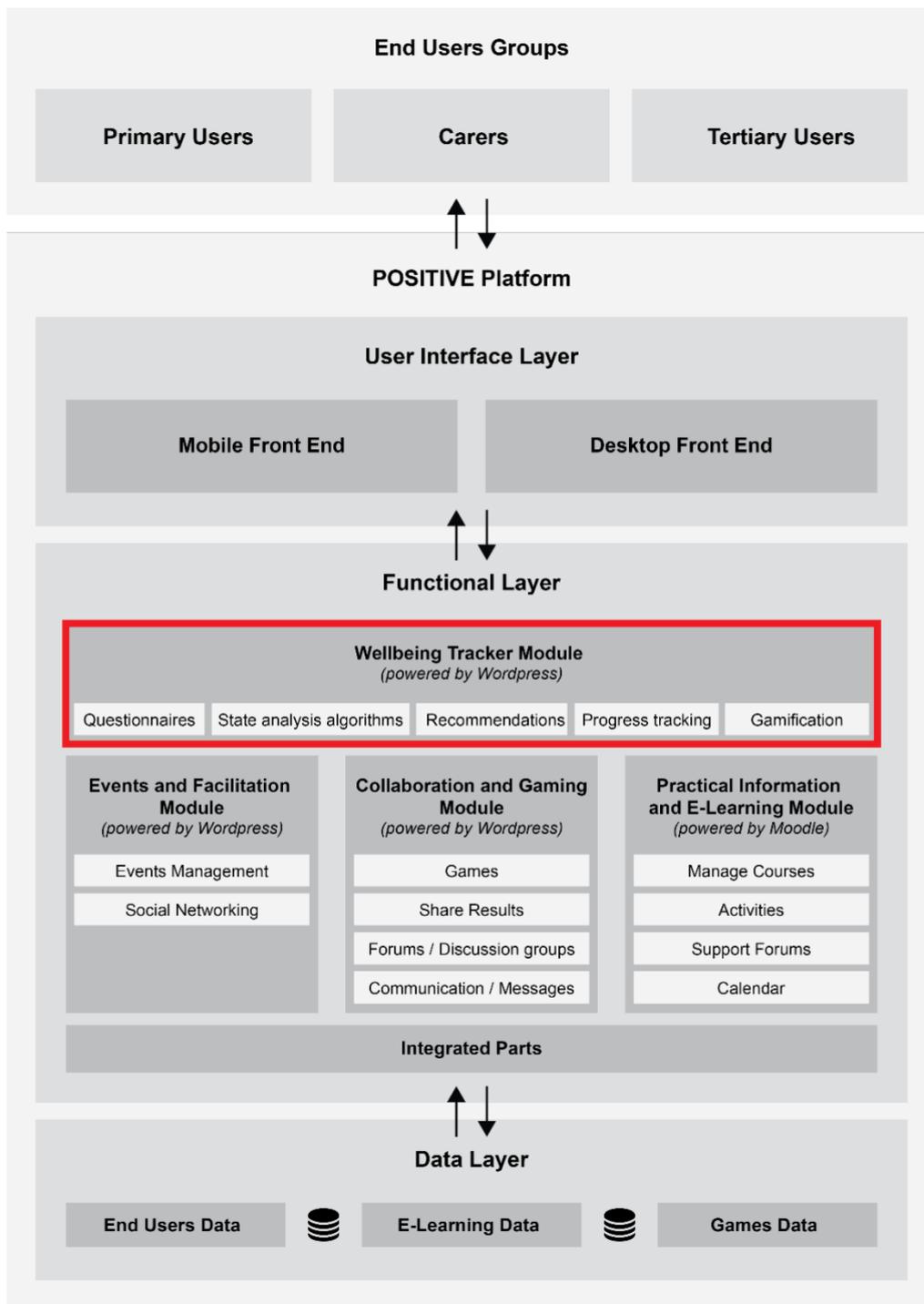


Figure 13: POSITIVE Wellbeing Tracker Module

### 7.2.1 Supported Functionality

More specifically through this module the following functionalities will be available:

#### Tracking measures / Questionnaires

The primary end user (elderly) will be prompted to fill in a specific questionnaire. The questions will consist of two main pillars, physical and mental wellbeing. Physical wellbeing questions will emphasize on the physical parameters of the person, that allow calculating useful statistics together with current physical lifestyle habits.

Mental wellbeing questions will focus on the user’s ability to develop interpersonal relations, socializing and learning new things or practicing existing ones. It will also emphasize on mental activities that are relatively easy for him to perform and to the ones that he phases difficulties.

Below you can find some indicative questions to be included in the questionnaires (both for physical and mental wellbeing parts).

<b>Mental Wellbeing Questions</b>
1. I generally feel happy.
2. I experience a lot of anxiety in my life.
3. My social life is satisfying.
4. My sleep quality is good. I have no trouble falling asleep and wake up refreshed.
5. I spend a lot of time with my family
6. I have a lot of excess energy.
7. I engage in activities related to spirituality/relaxation (such as prayer, meditation, breathing exercises, etc).
8. I am free from addictions and destructive behaviors.
9. I easily find ways to spend my free time.
10. I have hobbies and activities that bring me joy.
11. I have a lot of dreams and plans for the future.
12. I feel that I am growing as a person.

**Table 4: Physical Wellbeing Questionnaire**

<b>Physical Wellbeing Questions</b>
1. How old are you?
2. What is your weight?
3. What is your height?
4. How often do You exercise? - <75 minutes a week - 75-150 minutes a week - 150-225 minutes a week - >300 minutes a week
5. What kind of physical activities do you carry out? -jogging -walking -dancing -gardening -cycling -nordic walking -community activities -household chores -swimming -other -none
6. What is your favorite physical activity? -jogging -walking -dancing -gardening -cycling -nordic walking -community activities -household chores -swimming -other -none
7. Do you suffer from any physical disease? -Cardiovascular Disease -Hypertension -Cancer -Osteoarthritis -Diabetes Mellitus -Osteoporosis -Multiple Chronic Conditions -other -none

**Table 5: Physical Wellbeing Questionnaire**

**State analysis algorithm**

Based on the state analysis algorithm the user answers and progress will be analyzed and he will be assigned to a specific user category and provide to him specific suggestions for events to participate, e-learning content to enroll and games to play. In that way a unified constantly learning analysis will be performed. A specific

**Recommendations**

Recommendations of certain activities in each module based on the users' input given while setting up the account( the user fills up the survey and based on that is presented with different paths of growth, which help him to level up and get harder at each level. It would require us to group activities from different modules into a block, that would be recommended to the user based on the initial interview

**Progress Tracking - Gamification**

The user profile will include all the summary of the user progress in terms of participation of platform activities. Following a gamified experience he will be assigned with scoring points and badges based on his participation in events, e-learning courses and games. He will also have the ability to share his progress with his family members and/or informal carers on Facebook.

**7.2.2 Technical functionality**

The implementation of the wellbeing tracker module will also be based on WordPress open source platform. Analysis about its features are presented above on section **7.1.2** .

**7.2.3 Data exchange and Interoperability**

Wellbeing tracker module	
<b>Input Data</b>	<ul style="list-style-type: none"> <li>• Configure users responses about mental wellbeing</li> <li>• Configure users responses about physical wellbeing</li> </ul>
<b>Output Data</b>	<ul style="list-style-type: none"> <li>• Information on users score</li> </ul>

### 7.3 POSITIVE Practical Information and e-Learning module

The aim of POSITIVE practical information and e-learning module is to provide to users access to webinars and e-learning courses. The practical information and e-learning module will be based on a flexible and easily configurable learning management system. This management system will enable users to access course-based training material. The corresponding section of the holistic platform architecture is highlighted within the borders of the red box as illustrated in Figure 14.

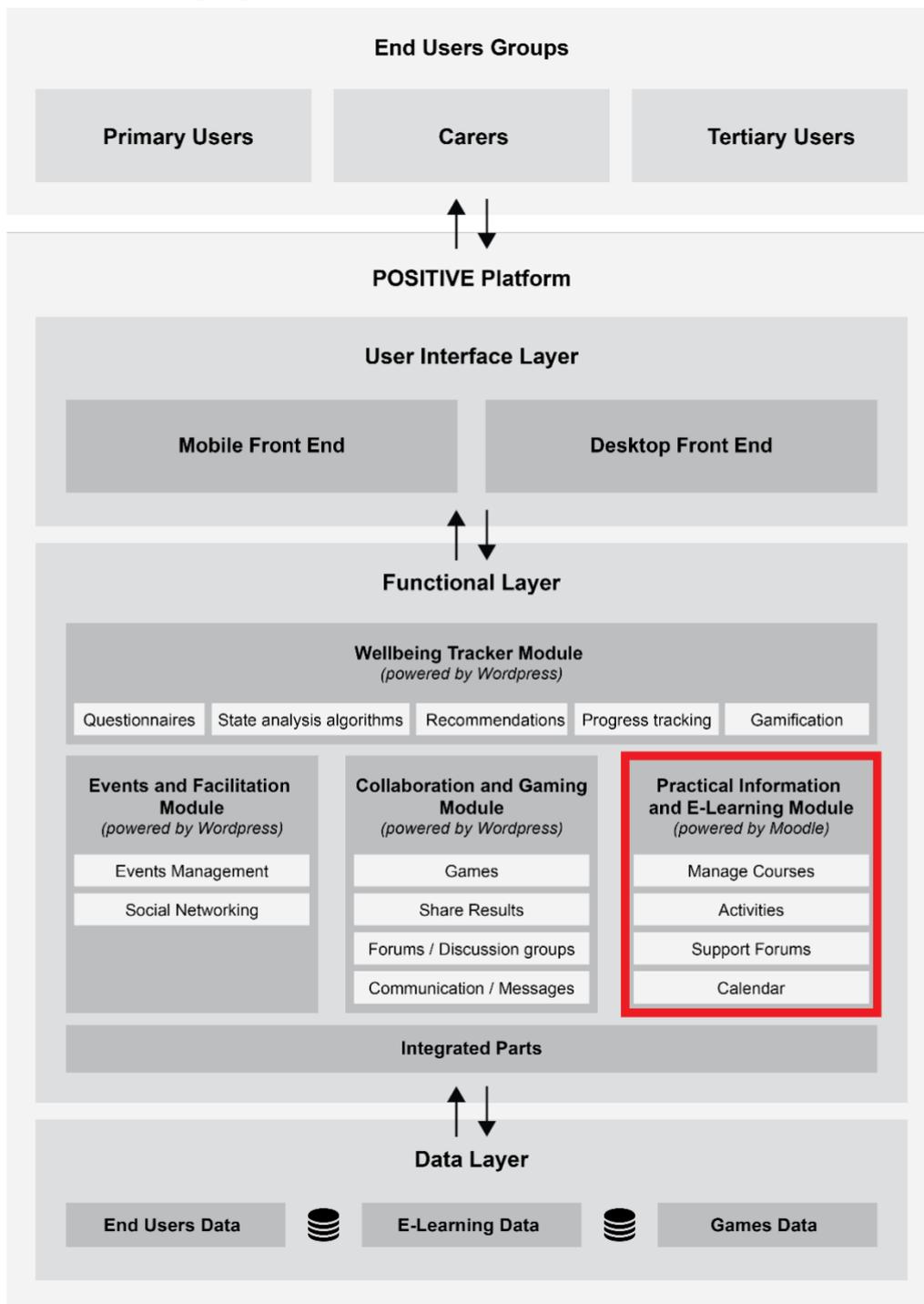


Figure 14: POSITIVE Practical Information and e-Learning module

### **7.3.1 Supported Functionality**

More specifically through this module the following functionalities will be available:

#### **Create and Manage Courses**

Tertiary users will have the ability to create courses and provide content to primary users to consume. They can monitor the participants within the course and their progress. They can provide activities for the user to participate. And they can also publish announcements for the participants of each course.

#### **Support forums**

The users will have the ability to see a list of suggested courses and his courses along with them already involved.

#### **Calendar**

Calendar functionality also is available and enable the user to provide planning of learning activities and set reminders. The users as they will participate in eLearning courses they will have the ability to follow quizzes, participate in support sessions.

### **7.3.2 Technical functionality**

The implementation of the Practical Information and e-Learning module will be based on the well-known open source learning management system (LMS) named Moodle. Moodle is a free, online Learning Management system providing a set of features enabling the development of dynamic courses. Provides functionalities that enable all user groups (tertiary users, elderly) to provide content, share material and participate in practical information sharing and e-learning activities.

#### **General Features**

The non-exclusive list below indicates most of the major general features of Moodle supporting the technical functionalities of the practical Information and e-Learning module [3].

- **Modern, easy to user interface.** Moodle provides an accessible user interface. On the same time it supports responsiveness and it is easy to navigate on both desktop and mobile devices.
- **Personalized Dashboard.** Moodle provides a personalized dashboard displaying the current, past and future courses, along with tasks due.
- **Collaborative tools and activities.** Moodle provides collaborative tools and enables the users to participate in respective activities (e.g. forums, wikis, glossaries, database activities).
- **All-in-one calendar.** Moodle provides a calendar tool helps that enables the users to keep track of their learning calendar including course deadlines, group meetings, and other personal events.
- **Convenient file management.** Moodle supports integration with cloud storage services (e.g. including MS OneDrive, Dropbox and Google Drive.) with drag and drop functionalities.

- **Simple and intuitive text editor.** Moodle provides a text editor enabling the users to format text and conveniently add media and images. The editor is compliant with all the web browsers and devices.
- **Notifications.** When enabled, users can receive automatic alerts on new assignments and deadlines, forum posts and also send private messages to one another.
- **Track progress.** Educators and learners can track progress and completion with an array of options for tracking individual activities or resources and at course level.

### Administrative Features

The non-exclusive list below indicates most of the major administrative features of Moodle supporting the technical functionalities of the Practical Information and e-Learning module [3].

- **Customizable site design and layout.** Moodle supports easy customization with the use of themes enabling to include your logo, desired color schemes and other customizations.
- **Secure authentication and mass enrolment.** Moodle provides over 50 authentication and enrolment options to enable users to add and enroll their users to the Moodle site and the courses they include.
- **Multilingual capability.** Moodle enables the users to view course content and learn in their own language or set it up for multilingual users and organizations.
- **Manage user roles and permissions.** Moodle addresses security concerns by defining roles to specify and manage user access.
- **Supports open standards.** Readily import and export IMS-LTI, SCORM courses and more into Moodle.
- **High interoperability.** Freely integrate external applications and content or create your own plugin for custom integrations.
- **Simple plugin management.** Install and disable plugins within a single admin interface.
- **Regular security updates.** Moodle is regularly updated with the latest security patches to help ensure your Moodle site is secure.
- **Detailed reporting and logs.** View and generate reports on activity and participation at course and site level.

### Course development and Management Features

The non-exclusive list below indicates most of the major course development and management features of Moodle supporting the technical functionalities of the Practical Information and e-Learning module [3].

- **Direct learning paths.** Design and manage courses to meet various requirements. Classes can be instructor-led, self-paced, blended or entirely online.
- **Encourage collaboration.** Built-in collaborative publishing features foster engagement and encourage content-driven collaboration.
- **Embed external resources.** Teach materials and include assignments from other sites and connect to the gradebook in Moodle.
- **Multimedia Integration.** Moodle's built-in media support enables you to easily search for and insert video and audio files in your courses.

- **Group management.** Group learners to share courses, differentiate activities and facilitate team work.
- **Marking workflow.** Conveniently assign different markers to assignments, manage grade moderation and control when marks are released to individual learners.
- **In-line marking.** Easily review and provide in-line feedback by annotating files directly within browser.
- **Peer and self assessment.** Built-in activities such as workshops and surveys encourages learners to view, grade and assess their own and other course members' work as a group.
- **Integrated Badges.** Fully compatible with Mozilla Open Badges, motivate learners and reward participation and achievement with customized Badges.
- **Outcomes and rubrics.** Select from advanced grading methods to tailor the gradebook to your course and examination criteria.
- **Competency based marking.** Set up competencies with personal learning plans across courses and activities..
- **Security and privacy.** Teach and share in a private space only you and your class can access.

### **Moodle structure**

In the section below the basic structure of Moodle is presented [3].

#### **The Front page**

The Front page of a Moodle site - the page you reach from your browser - usually includes information about the establishment itself and can be highly customised. (Note that it is also possible to lock the front page down so that all a user see when they click on the Moodle URL is a log in screen.) The way that end users join a Moodle site depends on the establishment: they might be given logins; they might be allowed to make accounts themselves, or they might be signed in automatically from another system.

#### **Inside Moodle**

Moodle's basic structure is organised around courses. These are basically pages or areas within Moodle where teachers can present their learning resources and activities to students. They can have different layouts but they usually include a number of central sections where materials are displayed and side blocks offering extra features or information.

Courses can contain content for a year's studies, a single session or any other variants depending on the teacher or establishment. They can be used by one teacher or shared by a group of teachers. How students enrol on courses depends on the establishment; for example they can self-enrol, be enrolled manually by their teacher or automatically by the admin.

Courses are organised into categories. Physics, Chemistry and Biology courses might come under the Science category for instance.

#### **Teachers, students and other Moodle users**

You don't enter Moodle with the "teacher" or "student" role. Everyone who logs into Moodle has no special privileges until they are allocated roles by the administrator according to their needs in individual courses or contexts.

**Finding your way around**

A logged in user can access areas of Moodle such as their courses or profile from the Navigation block and Administration block. What a user sees in these blocks depends on their role and any privileges granted them by the administrator. Each user has their own customisable Dashboard.

**7.3.3 Data exchange and Interoperability**

Moodle/Practical Information and e-Learning module	
<b>Input Data</b>	<ul style="list-style-type: none"> <li>• Configure Courses</li> <li>• Configure Storage Module (Files)</li> <li>• Configure Communication Module (Forums, Calendar, etc.)</li> <li>• Configure Evaluation Module (Assignments, Quizzes, Surveys)</li> </ul>
<b>Output Data</b>	<ul style="list-style-type: none"> <li>• List of available courses (even filtered by category)</li> <li>• List of courses in which a user is enrolled</li> <li>• Information on course activity of a user (completion tracking, grades)</li> <li>• List of badges of a user</li> </ul>

## 7.4 POSITIVE Collaboration & Gaming Module

In this section (section 7.4) we analyze the POSITIVE Collaboration & Gaming module. This module is responsible for providing all the required services to the POSITIVE platform user groups about playing games and about the engagement of users on the exchange of information and communication.

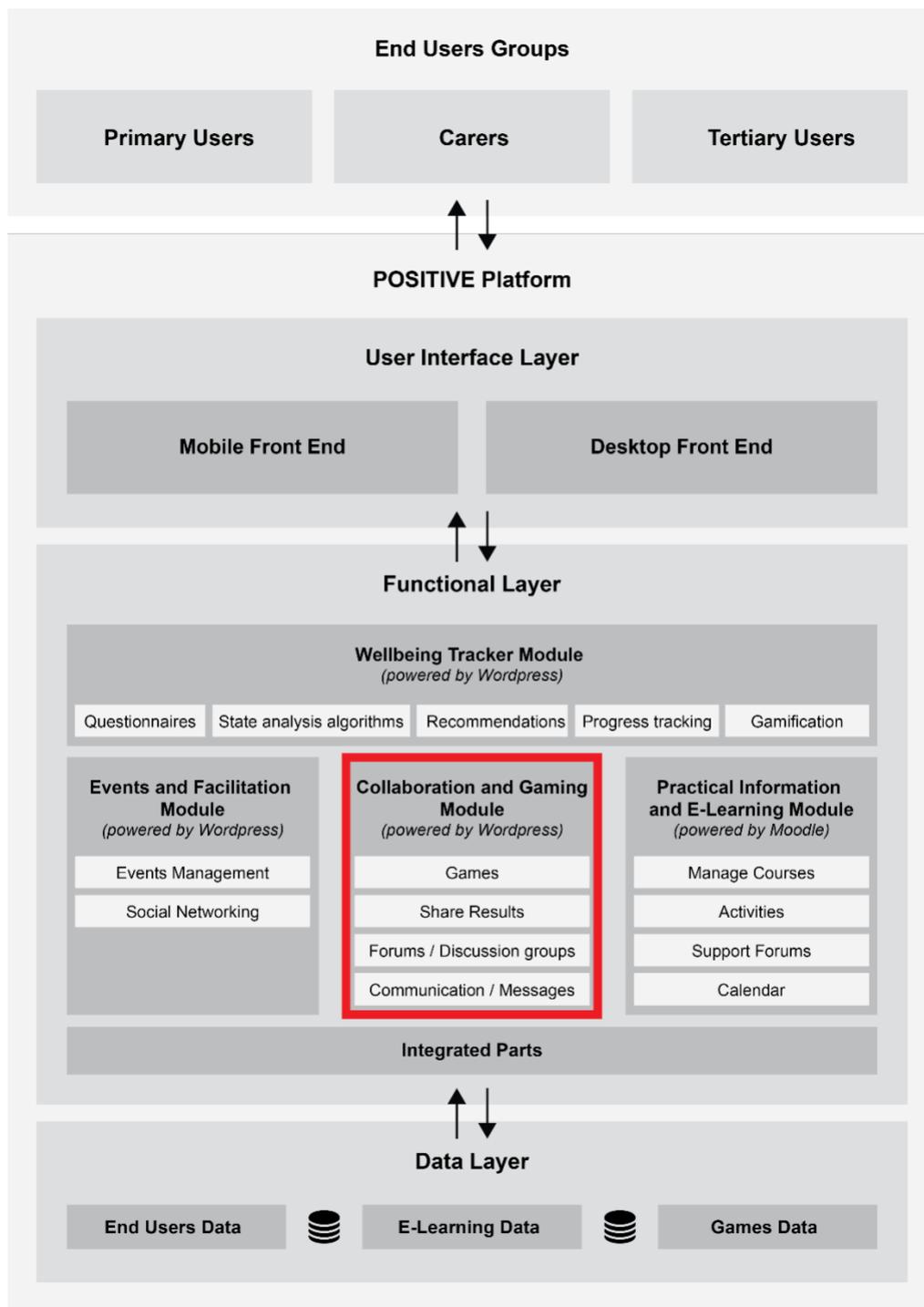


Figure 15: POSITIVE Collaboration and Gaming Module

**7.4.1 Supported Functionality**

The functionality will include:

**Games**

Actual games will be implemented covering various types of games. The game will include a welcome screen with instructions for the user to follow and as a next step the ability to select a specific difficulty level in order to play. The time he needed to complete the game will be monitored along with his final score.

**Share results**

The user will have the possibility to share his progress along with his family members through social network (e.g. Facebook).

**Forum groups / Discussion Rooms**

The necessary mechanisms will be available for the users in order to enable them participate in forum groups of related to specific interest topics or discussion rooms.

**Communication**

The communication mechanisms will support the communication through messages/chat.

**7.4.2 Technical functionality**

The implementation of the “Collaboration and Gaming” module will also be using WordPress, a free and open-source blogging tool, and a content management system (CMS). Analysis about its features are presented above on section **7.1.2** .

Games will be developed in a way that will support cross platform compatibility and they will be accessible from any device (Desktop/Mobile/Tablet) and operating system (Windows/Mac/Linux).

**7.4.3 Data exchange and Interoperability**

Collaboration & Gaming Module	
<b>Input Data</b>	<ul style="list-style-type: none"> <li>● Configure Games</li> <li>● Configure Communication</li> </ul>
<b>Output Data</b>	<ul style="list-style-type: none"> <li>● List of available games</li> <li>● List of games suggested for a user</li> <li>● Information on game activity of a user (time,score)</li> </ul>

## 6. Integration plan

The integration planning and delivery principles are the set of actions through which all integration activities should be focused to deliver a complete solution for POSITIVE.

### 8.1 Overview

Integration of POSITIVE modules includes the integration of the four (4) software modules that comprise the platform. Apart from the integration steps that should be followed, validation and verification of the each module integration along with whole system validation and verification must be performed, before providing the system to the end users for the purposes of the pilot studies. In that sense, at least one person from each participating (technical) partners must be involved in this process, to ensure that the guidelines have been followed and the integrated solution fulfils the user requirements.

### 8.2 Purpose

The purpose of the integration plan is targeting technical partners and provides the steps to be followed for the development and delivery of each component, to be fully compatible with the rest of POSITIVE platform.

### 8.3 Integration Strategy

The first principle for an unobtrusive integration process is focused on the non-modification of the agreed data exchange contracts. Any modifications should be properly communicated between the involved partners and should not deviate from some specific prerequisites, such as the backward compatibility with the communication contracts and the application of the security requirements. Software developers use the Software Development Life Cycle (SDLC) to implement ICT projects successfully. The software industry follows the SDLC to design, develop, and test projects before deployment. When carried out effectively, the SDLC produces high-quality software that meets customer expectations and reaches completion within time and cost estimates. Because it defines which tasks must be performed at each step in the software development process, the SDLC is a method of quality control and a way to ensure that tech development teams stay on the same page. For POSITIVE platform, the hardware components used are off-the-self, so the main focus is on the integration of the software components of the system.

The SDLC consists of a detailed plan describing the development, maintenance, and enhancements involved with specific software. The engineers and developers use the SDLC to plan for, design, build, test and deliver ICT systems and they must carry it out in a structured and methodical way to be successful, because each stage relies on the successful results of the previous one. A continuous integration practice will be adopted for the integration of POSITIVE modules to minimize the risk of incompatibilities between the modules at an early stage of the integration process. For the components for which the code will be shared among the partners, a remote repository to hold the source code will be defined. For each module, detailed build and configuration guidelines must be provided. For components such as the Cloud backend infrastructure, a live documentation is also recommended.

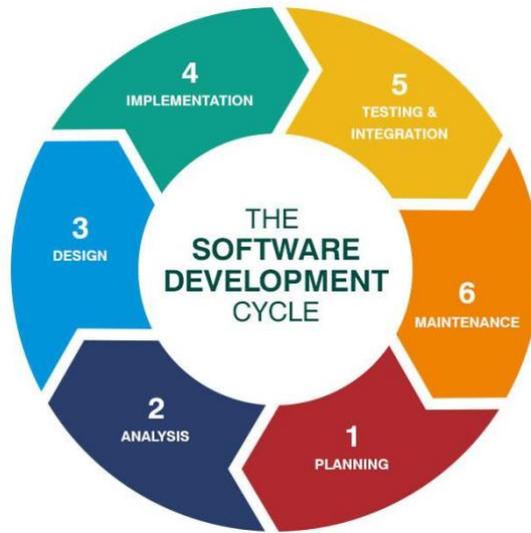


Figure 16: Software Development Life Cycle

## 7. Deployment

For the deployment of the platform layers the following software will be installed.

### MySQL

MySQL is an Oracle-backed open source relational database management system (RDBMS) based on Structured Query Language (SQL). MySQL runs on virtually all platforms, including Linux, UNIX and Windows. Although it can be used in a wide range of applications, MySQL is most often associated with web applications and online publishing. Described in more details at section 6.

### Apache Web Server

The Apache HTTP Server is a web server software. Apache supports a variety of features, many implemented as compiled modules which extend the core functionality. These can range from server-side programming language support to authentication schemes. Some common language interfaces support Perl, Python, Tcl, and PHP. Popular authentication modules include `mod_access`, `mod_auth`, `mod_digest`, and `mod_auth_digest`, the successor to `mod_digest`. A sample of other features include Secure Sockets Layer and Transport Layer Security support (`mod_ssl`), a proxy module (`mod_proxy`), a URL rewriter (`mod_rewrite`), custom log files (`mod_log_config`), and filtering support (`mod_include` and `mod_ext_filter`).

### PHP

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP code can be simply mixed with HTML code, or it can be used in combination with various templating engines and web frameworks. PHP code is usually processed by a PHP interpreter, which is usually implemented as a web server's native module or a Common Gateway Interface (CGI) executable. After the PHP code is interpreted and executed, the web server sends resulting output to its client, usually in form of a part of the generated web page; for example, PHP code can generate a web page's HTML code, an image, or some other data.

### WordPress

WordPress is a free and open source content management system (CMS) based on PHP and MySQL. It is the most widely used CMS software in the world, and as of May 2019, it powers more than 30% of the top 10 million websites and has an estimated 60% market share of all websites built using a CMS [2]. WordPress started as a simple blogging system in 2003, but it has evolved into a full CMS with thousands of plugins, widgets, and themes. It is licensed under the General Public License (GPLv2 or later). Described in more details at section 7 .

### Moodle

Moodle is a popular open source platform providing Learning Management Services. It's an open source LMS which provides a personalized learning environment to all its users. Described in more details at section 7.

---

## 8. Conclusions

The scope of this document is to provide the outcome of task T2.4. The technical architecture of POSITIVE platform is outlined and the respective layers and modules are presented and analyzed. At the bottom layer we have the data layer including the information persistence part of the platform. Its implementation will be based on MySQL and it will be deployed to an Ubuntu server. The communication will be done with all the other modules of the above layer (Functional Layer). Functional layer comprises of four different modules: Well-being tracker, Events and Facilitation Module, Collaboration & Gaming Module and Practical Information and e-Learning module. Well-being tracker module will be one of the most important modules of the system as it will analyze the state of the user and provide suggestions for the user including events to attend, webinars to subscribe and gaming activities to participate. Its implementation will be based on WordPress. Events and Facilitation Module will provide the necessary mechanisms for the different user groups to create, manage, join and/or their friends to join various events. It will also include the respective social networking functionalities. Its implementation will also be based on WordPress. Collaboration and Gaming Module will implement a set of services that will enable the primary end users of POSITIVE platform to participate in gaming and communication activities, as well as enable them to participate in forums and discussions groups. Its implementation will also be based on WordPress. Last module of functional layer of POSITIVE platform, Practical Information and e-Learning module will provide the necessary mechanisms for the different user groups to create, manage, plan and participate in practical information sharing and e-learning activities. Its implementation will be based on the well know, free and open source Learning Management System Moodle. Please note that as the development and integration of each component is progressing, minor deviation may arise, due to the dynamic nature of software engineering.

---

## 9. References

- [1] MySQL reference documentation: <http://dev.mysql.com/doc/>
- [2] WordPress documentation: <https://codex.wordpress.org/>
- [3] Moodle documentation: <https://docs.moodle.org/>