|  |  |
| --- | --- |
|  | Moving Picture, Audio and Data Coding by Artificial Intelligence  www.mpai.community |

**Public document**

|  |  |
| --- | --- |
| **N875** | 2022/09/30 |
| **Source** | Requirements (AIH) |
| **Title** | MPAI-AIH Use-cases and functional requirements WD0.2 |
| **Target** | MPAI-24 |

# Introduction

Artificial Intelligence for Health data (MPAI-AIH) is an MPAI project addressing secure collection, AI-based processing, and secure access to Health data. The motivation for the development of this standardization comes from the AIM Health Portuguese project (https://istar.iscte-iul.pt/aimhealth-en/)[[1]](#footnote-1).

# AIM Health Secure Platform

The overall architecture for MPAI-AIH (Figure 1) comprises a set of different actors, specific distributed services and APIs that will be described in the following sections.

Diagram

Description automatically generated

Figure 1 - Reference Model of MPAI-AIH Secure Platform

The platform is divided into front-end and back-end components. The front-end architecture is detailed in the following image (Figure 2).



Figure 2 – Front-end architecture of the MPAI-AIH Secure Platform

## Actors

The AI for Health data system identifies and recognizes the following different actors/entities.

1. *End-user*: a user for which his/her health-related data is going to be collected by the system. This end-user will control and audit the access of any third party to his/her health-related data according to the terms of a smart contract issued at the time a “third party” entity requires access and gets user approval to do so.
2. *Third-party*: any third-party entity requiring access to the data on the system or to process that data and extract knowledge through the usage of some artificial intelligence mechanism (or through the orchestration of multiple intelligence mechanisms). This includes hospitals, clinics, research centers, caretakers, and others. Access is granted according to the terms of the smart contract between that third party and the end-user. TBD: how the terms are verified for legal compliance.
3. *External Data Sources*: represent platforms from which the system may collect subsidiary data for the integration of relevant information for health-related predictions.

## Services

The AI Health data system is composed of a set of distributed group components and services that are depicted in Figure 1. These are:

1. The *front-end system*, a smart device (smartphone or other similar) application (AI-Health app) that is capable of capturing, storing, and processing health-related data from the end-user. The user smart device represents the personal gateway to the user-data. This data can be collected directly using the device sensors and applications installed in the smartphone, such as Google Fit and Apple Health. The smartphone can also act as communication gateway for any external biometric sensors that capture health-related data events from the user. All the collected data will be securely stored in the user’s smartphone using a “Secure Data Vault” whose access is controlled by the end-user (Figure 2). The smartphone will also alert the end-user about any deviation that may be associated to a disease or symptomatic abnormality that can be inferred from a machine learning data analysis.
2. The back-end system, composed by a set of tools that implement the necessary services to securely store data, de-identify and anonymize data, control entity authentication and access control to data, and to license and audit the access to health-related user data on the backend system. This system gathers anonymized data from all the various sources (the users that allow it and other important sources) and acts as a broker gateway between the entities that would like to access data and those who will supply it.

There are two types of smart-contracts (SC): between the user and the backend, and between the backend and the entity.

Elements of the SC are:

1. Parties.
2. Data.
3. Rights granted
   1. Processing to be made on the data (need a taxonomy).
   2. Auditing of the processing done by the backend and by the third-party entity using the data.

The backend grants the rights without making reference to the end users identification who have provided the data. Of course, the backend may not grant the entity the right to make processing that the user did not grant to the backend.

The system takes advantage of Blockchain and Distributed Ledger technologies (B&DLT). The objective of this B&DLT system is to enable the transparency and auditability of the system. Every health-related data access request will require the emission of a license (smart contract) that will be stored on the B&DLT. This smart contract will contain information about the requesting entities (someone requesting access to data), the data requested, the access conditions (e.g., timeframe and the user’s permission). This information can be audited by the system and by the user to be sure that the data is being used according to what was established in the smart contract. TBD implications of the legal verification of the smart contract.

The system will also display a set of AI services that can be used to directly treat and process the data on the device for the extraction of specific knowledge of interest to the end-user or for third parties with contracted rights. These services may be selected from the ones available from the MPAI Store and may be orchestrated to produce specific analyses for the entities that request access to health-related data. AI services through data processing enable specific and customized training of machine learning models to identify and assist in the identification of medical diagnosis and prognosis.

## APIs

The system will use REST API interfaces that will provide data access. A special API acts as the interface between the overall system’s backend data and the AI modules for using and processing data. Data may also be collected from other sources such as public services and third-party entities, using other specialized APIs.

### API: Mobile App <-> Back-end System

This describes the API that is exposed by the back-end system to the smart device app. This API will provide the necessary services to register, authenticate and control access of the user in the system. Moreover, this API will also provide the mechanisms for handling user health related data on the system – data storage, permissions for data usage, and data usage auditing.

This API is composed of the following API entry points. Further details of this API is provided in the Annex A: API Description.

TBD: the description of the API.

### API: Back-end System <-> Third Parties

This API is composed of the following API entry points. Further details of this API is provided in the Annex A: API Description.

TBD: the description of the API.

### API: Back-end System <-> B&DLT

This API is composed of the following API entry points. Further details of this API is provided in the Annex A: API Description.

TBD: the description of the API.

TBC: is there any other API that is relevant to specify in the document?

# Healthcare use case

This section presents a simple use case[[2]](#footnote-2) that describes the system and its usage (Figure 1).

## User health data collection

* The user signs into the system, which is equipped with the necessary security features and initializes the end-user secure data vault.
* The user configures the smartphone app to connect to the different data sources (either external sensors or internal installed apps).
* The MPAI-AIH app starts collecting data from the user and securely stores it locally on the smartphone.
* The user is requested for permission to contribute with health data to the system. The user analyses the request and gives permission for this contribution (it may be all or just some specific data). If the user gives access to his/her data, the backend will create a smart contract to be accepted by the user and data will be collected via an API.
* The data from all the users participating in the system that have given their permission is collected into the global secure data storage where it will be further de-identified and anonymized.

## Background processes

Background processes include:

1. Housekeeping services.
2. Inference services: a set of services processing data and using machine learning models enabling results of general use.

## Access to healthcare data

This section deals with on-demand processing of health data. A demand may involve a huge amount of data transfer and processing. However, the project does not consider the associated cost, latency etc.

* Any authorized and authenticated third-party entity may request data access. This entity needs to be properly registered and authenticated on the system to be able to access the proper APIs to request access to data.
* The entity requests access to the data catalogue existing on the system. The catalogue provides the following data with the following level of detail. The entity optionally selects the intelligent mechanisms that exist to process the data and extract some type of results and intelligence from the data. TBD: shall there be a need of a SC for processing the data inside the smart phone? What is written in the SC: I give you the right to process my data the way you like of I allow you to process my data with proc1, proc2,…
* This involves the selection of the specific services to process the data (one service, or multiple services properly orchestrated) that can access the system’s data and perform some treatment. The existing services to process data will be selected and instantiated for this processing. Access to the service is based on a choice made from a service taxonomy, likely to be compatible with the taxonomy used in SCs.
* The entity accepts a smart contract created by the B&DLT users. Access to data is permitted as long as the smart contract is valid.

## Intelligent processing of healthcare data

Intelligent processing of healthcare data follows best practices and of state-of-the-art machine learning techniques. Thus, it must be technically and socially robust, that is, accurate and reproducible, and able to deal with and inform about possible failures, inaccuracies and errors, aware of the potential repercussions of false positive (resp. negative) responses, and adopting privacy and security-preserving techniques, and allow for adequate knowledge sharing.

* State of the art machine learning: This includes the dimensions of efficient computational processing with the proper trade-offs between computational cost and accuracy; the adoption and identification of techniques to identify bias-free and representative datasets.
* Efficient Implementation Architecture: This addresses the search for computational organization that is best adapted to the task at hand in terms of resource utilization, namely execution hardware resources, and energy and computational requirements. The alternatives include the adoption of centralized server organizations that concentrate processing and distribute global knowledge, as well as distributed and continuously-learning models.
* Explainable Artificial Intelligence: The communication of the results of intelligent processing of healthcare data must strive to adopt techniques that provide a rationale for computer-generated decisions, along with reasonable estimates of the accuracy of a particular response, as well as a relative ranking of plausible alternatives.
* Security and Privacy preservation: The intelligent of large health-related datasets is done in such a way as to preserve the privacy of individuals, namely via the adoption of anonymization techniques and the use of security-preserving communication and storage methods.
* Knowledge Sharing: This point focuses on how learning from one user can be transferred and aggregated with learning from another user, while maintaining user’s privacy.

In practical terms, the processing of healthcare data includes:

1. If the data has not been already processed, an AIW (AI Workflow) is selected to process the data and the AIMs (AI Modules) load the data as needed and may store the data in the “AIM Storage”.
2. The AIW orchestrates the execution of the AIMs, which operate over the data. All of them can be downloaded or updated from the “MPAI Store”, if necessary.
3. The data is processed based on the AIW and stored in the secure data vault.

## Auditing healthcare data access

This is a mechanism in the system that allows users to audit the usage of their healthcare data. To accomplish this, the users require auditing through the backend API.

# Intelligent Computational Service Organization

This section discusses the potential organization of intelligent computational services, which are classified as Centralized server-slave architectures, or Distributed organizations that collaborate

## Centralized Services

This section discusses the centralized learning processes and services offered by MPAI-AIH. Chief among those are centralized master-slave architectures driven by a high-resource master in charge or controlling and distributing intelligent models to slave processing.

## Distributed Services

This section discusses the Distributed intelligent services organization. Chief among those are Federated Learning processes and services offered by MPAI-AIH.

## Federated Learning Mobile applications

Federated Learning (also known as collaborative learning) is a technique that enables machine learning algorithms deployed across multiple decentralized edge devices or servers holding local data samples to collaboratively train a global model. This is done without exchanging user data, only (incremental) changes from the local machine learning model are uploaded to the global machine learning model and, eventually, a new (updated) model is downloaded into the edge devices, in an exchange between the decentralized devices and a server.

## Federated Learning: Healthcare Application Use Case

Healthcare and the health insurance industry may leverage federated learning systems since it allows for the protection of sensitive user data in the original edge device. Because data is kept locally, Federated Learning may be used to build AI models on user’s health information from a data pool of smartphones without leaking personal data. Federated learning models can provide for improved data diversity by gathering data from various locations and use cases (e.g., hospitals, electronic health record databases), for instance, to diagnose rare diseases.

The article in Reference [2] by Rieke et. Al. illustrates how federated learning can help solve challenges about data privacy and data governance by enabling machine learning models from distributed data sources.

## Federated Learning Processes

Federated learning is composed of two processes: training and inference.

### Training process

In the most common approach, presented by McMahan *et al.* [1], the training is done using a client-server architecture, as illustrated in Figure 3. A shared global model is defined by a central controller, also referred to as server (the backend system). Each client who participates in collaborative learning has a copy of the shared global model (their local machine learning model) and their private data set. The shared global model training is performed in rounds. At each round, the following steps are performed:

1. Groups of clients are selected in sequence by the server and sent a copy of the global model parameters (W).
2. The selected clients load the received parameters into their respective models and train them with their respective private datasets for a defined number of iterations/epochs. At the end, each client sends its parameters to the server (Δw).
3. Using an aggregation algorithm, the server combines the parameters received from the various clients and updates the global model.
4. The executions are then repeated until the model reaches convergence. In this way, sensitive data is not sent directly to the server, guaranteeing a certain level of privacy for clients

It is worth noticing that, even when the global shared model reaches convergence, it can be incrementally trained as the client’s data sets grow, in order to be more robust.

For example, as end-users interact with a machine learning application in their smartphones, such as keyboard’s word suggestions, they naturally spot and correct discrepancies in the predictions of the machine learning application (*e.g.,* discarding wrong predicted words in a sentence). These corrections create adjustments to local training datasets in each users’ device, and the local model is updated. Similarly, a user’s specific health data may be collected locally on the device and used to update a local model.

Diagram

Description automatically generated

Figure 3 - Federated Learning training process.

### Inference Process

For inference, each client simply uses the weights received from the global model and runs it on the desired data. Depending on the problem, sometimes the client may wish to perform a fine-tuning in their local model, in order to improve the accuracy and specialization of the model for herself/himself.

## MPAI-AIH Federated Components

End-user devices and third-party entities communicate with the MPAI-AIH Secure Platform using secure APIs, to provide and request data.

The AIWs describes the process used to handle healthcare data. These AIWs orchestrate the usage of the AIMs, which are responsible for performing the computational operations, including transformations, training and inferences. For example, an AIW used to detect the presence of COVID-19 from a given user may have two AIMs, one for select, load and pre-process (e.g., apply normalization and cleaning) and another to load the deep learning model and perform the inference.

The MPAI Store is responsible for providing these AIWs and their AIMs. The AIMs are stored in the end-user devices, at “AIM Storage”. This process is orchestrated by the controller which may use the different components (e.g., “communication”, “global store” and “access”) to communicate with the backend and perform desired operations.

It is worth mentioning that multiple instances of AIWs can exist at the end-user device, and they can have the same or different objectives and may work with the same or different data, data sets or versions.

# To be considered

Study workflows for potential standardization. Data ingested, workflows, etc.

Identify data types acquired;

Identify processing on the data, either as single entity or workflow;

Identify new processing that can leverage the architecture.

New research are: “System aspects of federated learning implemented in the system and AIF in particular”

Potential requirement to have smart contracts executed in multiple BC.

# References

[1] McMahan, Brendan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. "Communication-efficient learning of deep networks from decentralized data." In *Artificial intelligence and statistics*, pp. 1273-1282. PMLR, 2017.

[2]Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H.R., Albarqouni, S., Bakas, S., Galtier, M.N., Landman, B.A., Maier-Hein, K. and Ourselin, S.; The future of digital health with federated learning; NPJ digital medicine, 3(1), pp.1-7; 2020.

# Annex A: API Description

## API: Mobile App <-> Back-end System

|  |  |  |  |
| --- | --- | --- | --- |
| **API entry point** | **Input Data** | **Output Data** | **Description** |
|  |  |  |  |

## API: Back-end System <-> Third Parties

|  |  |  |  |
| --- | --- | --- | --- |
| **API entry point** | **Input Data** | **Output Data** | **Description** |
|  |  |  |  |

## API: Back-end System <-> B&DLT

|  |  |  |  |
| --- | --- | --- | --- |
| **API entry point** | **Input Data** | **Output Data** | **Description** |
|  |  |  |  |

1. The AIM Health project is funded by the Portuguese government and aims an early identification of symptoms that might be related to a COVID-19 infection, especially for patients suffering from heart and cardiac problems. [↑](#footnote-ref-1)
2. This should be considered as just a simple possible use case. Multiple usage scenarios may coexist in the system, but for simplicity purposes, we have just described a possible one. [↑](#footnote-ref-2)