IMMERSE

Implementing Mobile MEntal health Recording Strategy for Europe

H2020 - 945263

D3.3	**Implementation report for 3rd platform release**
Dissemination level[1]	**PU**
Contractual date of delivery	31.03.2024
Actual date of delivery	05.03.2024
Type[2]	R
Work package	3
Work package lead	Thomas Ganslandt

[1] **Please choose the appropriate reference:**
PU = Public, fully open, e.g. web;
CO = Confidential, restricted under conditions set out in Model Grant Agreement;
CI = Classified, information as referred to in Commission Decision 2001/844/EC.

[2] **Use one of the following codes:**
R: Document, report (excluding the periodic and final reports)
DEM: Demonstrator, pilot, prototype, plan designs
DEC: Websites, patents filing, press & media actions, videos, etc.
OTHER: Software, technical diagram, etc

Author list

Organisation	Name	Contact information
6 - UKER	Thomas Ganslandt	Thomas.ganslandt@medma.uni-heidelberg.de
6 - UKER	Geun Hyun Kim	GeunHyun.Kim@medma.uni-heidelberg.de
6 - UKER	Wolfgang Krebs	Wolfgang.krebs@uk-erlangen.de
2 - CIMH	Anita Schick	anita.schick@zi-mannheim.de
8 - movisens GmbH	Simon Krause	simon.krause@movisens.com

Document history

Version	Date	Description
1	**05.03.2024**	**Initial release**

List of abbreviations

- API: Application Programming Interface
- CSV: Comma-separated values
- DMMH: Digital Mobile Mental Health
- DMP: Data management plan
- eCRF: Electronic case report form
- FAIR: Findability, Accessibility, Interoperability, and Reuse
- GPS: Global Positioning System
- GUI: Graphical User Interface
- ESM: Experience Sampling Method
- FHIR: Fast Healthcare Interoperability Resources
- HL7: Health-Level 7
- ID: Identifier
- IDE: Integrated Development Environment
- IMMERSE: Implementing Mobile MEntal health Recording Strategy for Europe
- JSON: Java Script Object Notation
- LOINC: Logical Observation Identifiers Names and Codes
- SNOMED CT: Systematized Nomenclature of Medicine, Clinical Terms
- UKER: Universitätsklinikum Erlangen (University clinic Erlangen)
- WIFI: Wireless Fidelity
- WP: Work package

Table of contents

1. Summary

This report captures the current status of the implementation of the IT infrastructure, programming scripts and other tasks, as well as changes to the plans described in detail in the implementation guide (Immerse D3.2).

In the implemented architecture section, the current status of the implementation and modifications to the previous planning are described. The section research database and subsections depict the three different data sources in the Immerse project, how the data is exported and backed up, and how the data needs to be converted to allow the import into the new DMMH research database. The following section describes the planned and already established methods to increase the quality of the data and the capture of the finalization status of the study data. In section 2.4, the reasoning and the current status of the generation of synthetic data is explained. The next section includes all the details of the generated programming code and additional data protection and security concerns. The last section deals with the possibility of transferring data for all Immerse scientists at the different locations.

2. Deliverable Report: D3.3 Implementation Report

2.1. Implemented architecture

The following figure schematically shows the main components of the platform, study-specific components and interfaces as well as user interactions between them and is described in detail in the Implementation Guide document:

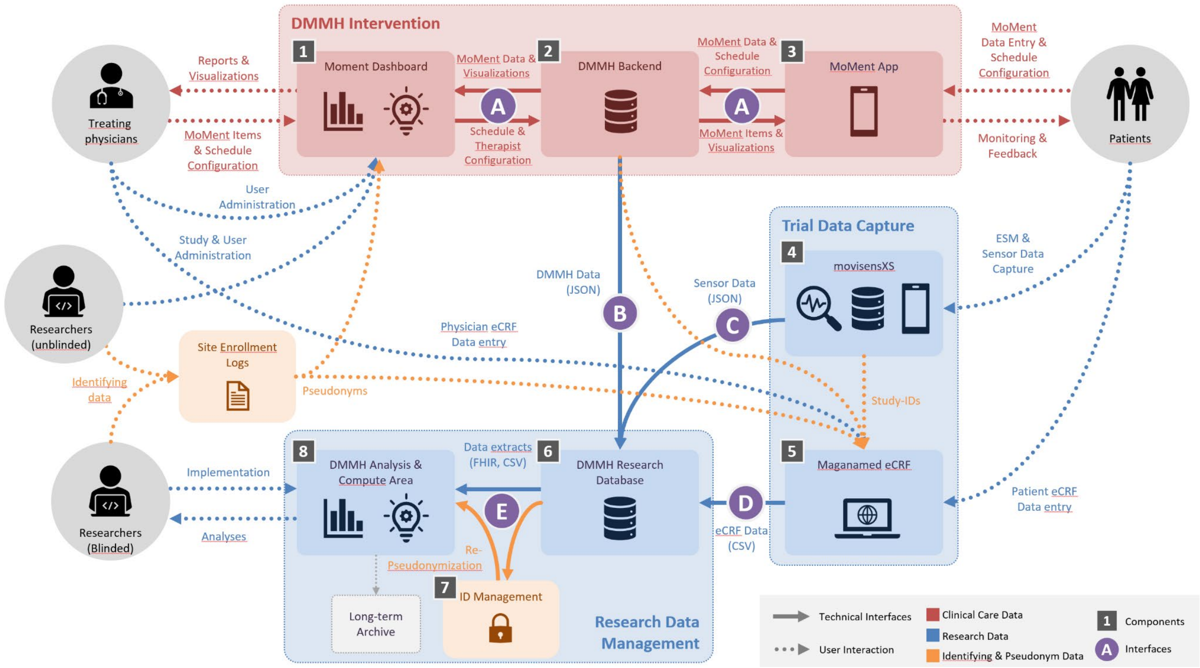

Figure 1: DMMH Platform components (boxes labeled clockwise with numbers 1 to 8), technical interfaces (arrows labeled clockwise with letters A to F) and manual user interactions (dotted arrows). DMMH Prototype components are shown in red, study-specific components for scientific use within the project are shown in blue, and the ID management component is shown in orange.

In addition to the described infrastructure in the implementation guide, a REDCap instance was set up at the UKER which contains data about study progress and pseudonymization (details in DMP version 3). Contrary to previous planning, site enrolment logs are recorded on paper at the respective IMMERSE study locations.

The currently implemented interfaces (B, C, D, E) are described in more detail in section 2.3 for the three different data sources to perform weekly backups or import the data into the DMMH research database. In the current release, data from the DMMH central research database are kept in their respective source formats, and approved data use-projects are provisioned with re-pseudonymized extracts in identical formats. The conversion of the native formats to HL7 FHIR resources as specified in the Implementation Guide will be implemented the next release with the primary goal of providing standardized representations of IMMERSE data to external users and enable sustainable re-use after the project funding phase.

2.2. Research database

The main function of the research database is the merging and secure storage of pseudonymized data collected through the different data sources (Maganamed, movisensXS and Moment App) and the export of specific extracts of the data for each approved data analysis request from researchers. The exported data is re-pseudonymized to prevent the unintended merging of partial datasets, with the help of the independent Trust Center at UKER.

An SQLite database (v3.35.5) is established via Python code and located on a storage path within the secure UKER network. The Python code for database generation and also the scripts for the import of the data are made available on the IMMERSE GitHub repository (https://github.com/immerse-eu/research_database).

Before the data can be sent to researchers, the used participant_IDs must be exchanged in a second pseudonymisation with new identifiers to prevent the merging of individual data exports. This is performed with a list of new pseudonyms from the Trust Center of UKER. An automated script replaces all IDs with the new pseudonyms (https://github.com/immerse-eu/re-pseudonymization).

2.2.1. Integration of Maganamed eCRF data

The eCRF data from Maganamed is exported via the export functionality of the GUI and stored in a secure location within the UKER network. The resulting data dictionary and CSV files from the export are used as input files to import the data into the research database. The programming code is located in the IMMERSE GitHub repository https://github.com/immerse-eu/research_database. For each of the eCRFs/CSV files, a new table is created in the database. To merge and filter the data for analysis requests from researchers, the different tables are merged as needed via SQL queries, with the pseudonym “participant_id” used as the primary key.

2.2.2. Integration of movisensXS ESM and mobile sensing data

A REST API is used to export the movisensXS ESM and mobile sensing data via a Python programming script (https://github.com/immerse-eu/movisensxs-extract). The API endpoints for the different datasets can only be accessed with the correct authorization (API key). For each of the time points (baseline – T3) and Immerse locations, a folder is created for the ESM data, and the data is stored in

JSON and Excel format. The sensing data is also saved in separate folders for each location, and subfolders for each participant include different sensing data files (e.g. activitylog, appusage or location) in CSV format. These files are also backed up weekly on a secure storage location in the UKER network.

To ensure the data integration of the movisensXS data with the Maganamed data in the research database, the data includes not only the movisensXS specific participant_ID but also the Maganamed pseudonym. These IDs can be used for a merge to export data for researchers out of the research database. To allow the import of the data files into the research database, additional steps are required and are described in the following section.

To enrich GPS sensing data with additional metadata, local Openstreetmap servers (https://nominatim.org) are established for each Immerse location (country) to get additional metadata for each set of latitude and longitude coordinates for all participants and sites. Each server is started in a local Docker container to ensure the protection of sensitive location data, and the country-specific data (in pbf format) is downloaded from the mentioned website. The available API can then be used to get metadata attributes for osm_type, osm_key, osm_value and geocoding_type (https://nominatim.org/release-docs/latest/api/Reverse/). In addition, GPS coordinates are subsequently anonymized by subtracting the median values for each participant. With this approach, the GPS positions are centered around the GPS coordinates 0,0, but the relative position of each set of coordinates to each other for a single participant can be retained for data analysis. In a final step, the custom IDs from sites Lothian and Lothian_CAMSH are replaced by the corresponding participant_IDs to allow the data integration of TherapyDesigner data with MovisensXS sensing data. All described procedures are carried out with an automated Python script stored in an Immerse GitHub repository (https://github.com/immerse-eu/sensing_location_metadata_anonymize).

2.2.3. Integration of DMMH data (Moment App)

At the time of publication of this document, the version of the Moment App API provided by movisens can only be used to export test data from a staging environment for the duration of a few days due to performance limitations. The functionality to export the complete data set has yet to be finalized by movisens. Until then, the data of the entire database can be transferred after encryption and a manual export by the developers of movisens GmbH via a secure transfer solution (e.g., UKER cloud) to the WP3 team. The data is saved in separate JSON format files, one file for each location, on a secure UKER network drive. The data is converted to tabular format prior to being imported into the research database. The computer code for the flattening was uploaded to the IMMERSE GitHub repository (https://github.com/immerse-eu/flatten_json_therapy_designer). Each dataset includes the participant_ID, which can be used to integrate the data with the data from the other two sources (Maganamed eCRFs and movisensXS).

2.3. Data quality and data capture finalization status

Detailed data quality checks were implemented to allow plausibility checks, e.g., if gender is identical in all questionnaires, or to cross-check questionnaire data at the item level to find discrepancies.

Custom programming scripts were created for dashboard visualizations to allow the assessment of the data quality and finalization status. Metadata timestamps from Maganamed exports are used to summarize the “completion status” of eCRF entries separated by location or eCRF as graphs or in tabular form. In collaboration with the project management team (WP1) specific eCRFs are chosen to estimate the finalization status in more detail to maximize the completion of data entries and allow a fast intervention if necessary. In addition, MovisensXS data entries are also monitored on the level of completion status for the different visits (T0-T3) and on the data quality level. This is, for example, necessary to ensure the presence of IDs that allow the integration of Maganamed data and MovisensXS data into one research database when the data collection phase is finished. These dashboard visualizations are updated on demand in close collaboration with the WP1 team. The programming code is stored in the IMMERSE GitHub repository (https://github.com/immerse-eu/immerse-dqa).

2.4. Synthetic data generation (dummy data)

To allow the establishment of data analysis pipelines and scripts for researchers even before data capture has been completed and original study data can be released, the WP3 team established the generation of simulated synthetic (dummy) data for the three different data sources, similar to real study data. Python scripts were generated to build eCRF and movisensXS ESM data in JSON or Excel format. This could already be finished for data from Maganamed and movisensXS and is currently in progress for Moment App data. The programming code is stored in the IMMERSE GitHub repository (https://github.com/immerse-eu/project-dummydata).

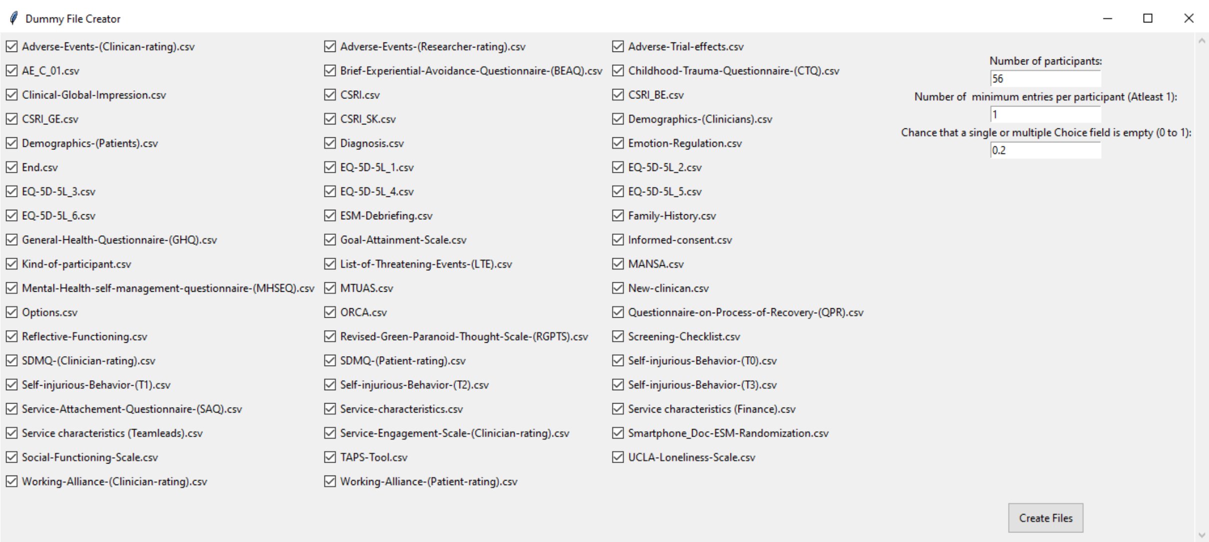

Figure 2: Dummy file creator: GUI to create Maganamed eCRF synthetic data. Specific eCRFs can be selected and additional parameters can be set at the upper right corner of the window.

2.5. Software development considerations

2.5.1. Code quality

A number of best practices are followed to ensure code quality. Code standards, including readability, maintainability and security, are established. The usage of version control via GitHub can track changes over time with multiple software developers, facilitates collaboration and transparency, and allows easy backups. Code reviews and the four-eyes principle improve code quality, enhance knowledge sharing and reduce the risk of bugs. A sophisticated IDE reduces errors, enhances productivity and improves code quality. Finally, an independent validation of computer code and its results by the usage of computer code by another programmer with a different programming language ensures high code quality and a reduced risk of errors.

All IMMERSE GitHub repositories and included code are published under the MIT license (https://opensource.org/license/mit) and will be made available to everyone as soon as the code is finished.

2.5.2. Independent validation

The WP3 team exports study data from the different sources used in the IMMERSE project on a regular basis (SOP DM04 "Primary Data Archiving"). This data is used to generate overview visualizations to determine the completion status of the data. This is performed with Python code that was created by data scientists of the data management team. These visualizations are shared with the study coordinator and PIs of the project. To validate the correct function of the Python scripts, a separate validated software suite (SPSS) was used, and an experienced software developer / data scientist from the study team at UKER developed code that executes a similar procedure as the Python code to transform and aggregate the raw data. With the same raw data, both scripts are executed by different data scientists. Finally, the results (aggregated numbers) of the SPSS code are compared with the results of the Python script. This comparison is done for at least two randomly chosen eCRFs/files to validate the results if the scripts are modified.

2.5.3. Data protection and security

The exported pseudonymized raw data and the generated code is only accessible by the WP3 team, and multiple procedures are established by the IT-department of UKER to protect the data and secure the access to the network system as described in the Data Processing Agreement. The developed computer code is saved in a specific Immerse GitHub repository. In addition, the Python code was designed to save sensitive configuration data like API tokens or usernames in a specific configuration.yaml file. This file is not available in the GitHub repository, and a template file needs to be filled out before the APIs can be used to export/download sensitive study data.

2.6. Data transfer and sharing

Folders on the Nextcloud-based UKER-cloud service were set up to allow secure data transfers between different Immerse sites for files of medium and even large sizes (Gigabyte range). These password-protected folders can be used independently of the site location and country. Data transfer and storage can be performed in a separate way for all folders, and only authorized personnel have access to them. The data will be backed up on a secure UKER network drive, where only the WP3 data management team has access to.