MASARYK UNIVERSITY

FACULTY OF INFORMATICS

# Anonymizing experimental data in the MAFIL laboratory

Bachelor's Thesis

FILIP KOVÁŘÍK

Advisor: Mgr. Ing. Tomáš Svoboda

Multimodal and Functional Imaging Laboratory – Central European Institute of Technology

Brno, Spring 2024

MUNI
FI

## Declaration

I hereby declare that this paper is solely my work. All sources, references, and literature supporting this work is properly cited and referenced.

Filip Kovářík

**Advisor:** Mgr. Ing. Tomáš Svoboda

## Acknowledgements

I would like to thank my advisor for his patience in answering my questions, and my family and friends for providing me with support during my work on this thesis.

## Abstract

Since its launch in 2015, CEITEC's Multimodal and Functional Imaging Laboratory (MAFIL) has been accumulating experimental data from volunteer participants. This data, processed under legitimate interest, is scheduled for disposal starting in 2025. MAFILDB, the laboratory's information system, currently lacks a process for removing personally identifiying information from this data.

Based on the laboratory's requirements, we design and partially implement an extension to MAFILDB capable of de-identifying user records. To track the new process, managed via new user interface views, we expand MAFILDB's data model, including a new *VisitDeidentification* entity. Lastly, we construct a pipeline to de-identify the records across MAFILDB and its PACS imaging subsystem.

Given a full implementation, MAFIL can fulfill legal obligations to volunteer participants mandated by privacy legislation. In applying the principle of storage limitation, de-identification reduces the impact of a potential data breach.

## Keywords

de-identification, anonymization, medical data, GDPR, PACS, CEITEC, MAFIL

# Contents


**1 Introduction** **1**
1.1 The MAFIL laboratory . . . . . 1
1.2 Anonymization and de-identification . . . . . 1
1.2.1 Anonymization in literature . . . . . 1
1.2.2 Anonymization as used herein . . . . . 2
1.3 The MAFILDB system . . . . . 2
1.4 MAFIL and user data . . . . . 3
1.4.1 Data collected by MAFIL . . . . . 3
1.4.2 Legal basis for data processing . . . . . 4
1.5 Aim of the thesis . . . . . 4
1.6 Limitations of the thesis . . . . . 5
1.6.1 Disclaimer of legal advice . . . . . 6

**2 Methods** **7**
2.1 Technologies . . . . . 7
2.1.1 Django . . . . . 7
2.1.2 DICOM and PACS . . . . . 7
2.2 Analysis . . . . . 9
2.2.1 Requirements elicitation . . . . . 9
2.2.2 MAFILDB data model . . . . . 10
2.2.3 De-identification pipeline . . . . . 13
2.2.4 Interfacing with PACS . . . . . 13
2.3 Design . . . . . 14
2.3.1 Interaction design . . . . . 14
2.3.2 Extending the data model . . . . . 17
2.3.3 De-identification state . . . . . 18
2.3.4 Modular decomposition . . . . . 22
2.4 Implementation . . . . . 22

**3 Results** **23**
3.1 Implementation results . . . . . 23
3.1.1 Other contributions . . . . . 23
3.2 Practical application . . . . . 23
3.3 Suggested amendments . . . . . 24

**Appendices** **26**

**A Implementation and other code contributions** **27**

**B Software specification for TomoCon® PACS script** **29**

# List of Tables

# List of Figures


2.1 Illustration of SCU and SCP Roles . . . . . 9
2.2 Entity-relation diagram of select MAFILDB entities . . . . 12
2.3 The *Anonymize visits* view . . . . . 16
2.4 The *Anonymization* view (prototype) . . . . . 16
2.5 Extension to the MAFILDB data model . . . . . 19
2.6 State diagram of the *VisitDeidentification* entity . . . . . 21

# 1 Introduction

## 1.1 The MAFIL laboratory

*MAFIL*, short for *Multimodal and Functional Imaging Laboratory*, is a Brno-based biomedical research laboratory.

MAFIL was established in 2014 as a project of *CEITEC*, the *Central European Institute for Technology*. Run in collaboration with Masaryk University, MAFIL is an open access laboratory: registered researchers from around the world may perform experiments on volunteer subjects and take measurements using the laboratory's equipment [1]. In 2022 alone, 48 research projects used MAFIL's facilities. In total, 1721 measurements were taken, amounting to nearly 3000 hours of experimental time [2].

True to its name, MAFIL's equipment is capable of measuring various modalities of data. Examples include magnetic resonance imaging (*MRI*), as well as electrography, such as *ECG* and *EEG*[1], among others [1].

## 1.2 Anonymization and de-identification

The terms *anonymization* and *de-identification* see inconsistent use across computer-science and biomedical literature [3]. However, both terms refer to processes with a similar aim: reducing the scope and impact of storing personally identifying information.

### 1.2.1 Anonymization in literature

According to Kushida et al., anonymization and de-identification differ in the following way: *Anonymization* is a process which makes it "virtually impossible" to link a data subject to her medical records. The goal of *de-identification*, on the other hand, is merely to increase the difficulty of making such a link [4].

Additional requirements for anonymization are set forth by the ISO/IEC 29100:2024 standard. For one, the ISO and IEC mandate that

1. *ECG* and *EEG* are short for electrocardiography and electroencephalography, measurements of the activity of the heart and the brain, respectively.

anonymization be an irreversible process. Moreover, the data subject must not be identifiable in any way, whether by the data controller "alone[,] or in collaboration with any other party" once the data in question is anonymized [5].

### 1.2.2 Anonymization as used herein

The laboratory's requirements do not include full anonymization of user data. In fact, full anonymization would also result in the devaluation of raw imaging data, and would pose an undue implementation burden. As such, the term *de-identification* is more fitting in describing the laboratory's requirements, and the thesis shall use this term to refer to the processes detailed below.

On the other hand, users of the system presented herein need not be concerned with the distinction between *anonymization* and *de-identification*. In prioritizing comprehension over precision, aspects of the system use the term *anonymization* and its variants to convey information to the user.

## 1.3 The MAFILDB system

*MAFILDB* is a biomedical research information system developed for MAFIL by Tomáš Svoboda, Jakub Čillík and Adrián Rošinec of CERIT-SC. At MAFIL, *MAFILDB* is the heart of operations; the system tracks all of MAFIL's measurements, sessions, visits, volunteer participants and research projects. MAFILDB consists of two main systems: the *PACS* medical imaging system[2], and the MAFILDB full-stack web application [6]. The instance of PACS in use by MAFIL is provided by TatraMed Software, s.r.o. under the trademark name TomoCon® PACS [7].

When a volunteer participant arrives at MAFIL, MAFILDB is responsible for establishing a new *Visit* record based on the volunteer participant's registration form data. Experimental data pertraining to the visit arrives in MAFILDB primarily via the *PACS* imaging system.

2. PACS stands for *Picture Archiving and Communication System*; see Section 2.1.2 'DICOM and PACS' on page 7.

From here, the data is manually imported into the MAFILDB application and linked to the visit using a common Visit identifier [6][8].

## 1.4 MAFIL and user data

In the course of its operation, MAFIL collects the personal data of its volunteer participants. Naturally, collection of personal data entails legal requirements on its safe handling and disposal. To date, however, no procedure[3] for disposing of this personal data has been established at MAFIL.

What's more, since commencing data collection in 2015, MAFIL will soon be obliged to dispose of its oldest data, as the end of its ten-year consent window is fast approaching. It is therefore imperative to establish a procedure for data disposal at MAFIL consistent with legal requirements [7].

This section therefore provides an overview of the data collected by MAFIL and the legal basis for collecting this data.

### 1.4.1 Data collected by MAFIL

Data collected by MAFIL as part of the registration procedure notably include the volunteer's

1. full name,
2. sex,
3. national identification number,
4. date of birth,
5. native language,
6. height and weight, and
7. answers to safety questions incl. pregnancy, medical history and body augmentations.

Apart from the registration procedure, MAFIL also collects data from experiments performed using its measuring equipment.

3. As noted later, this thesis only concerns itself with the disposal of metadata associated with experiments, as opposed to the de-identification of raw imaging data obtained from measuring equipment.

### 1.4.2 Legal basis for data processing

The cited legal basis for the processing of personal data by MAFIL is *legitimate interest* (Czech: *oprávněný zájem*) [9].

Article 6 of the European Union's General Data Protection Regulation defines legitimate interest as one of six legal bases for the lawful processing of personal data [10]. In practice, legitimate interest justifies the processing of personal data to "carry out tasks related to [your] business activities" [11].

In contrast, the legal basis of *informed consent* (Czech: *informovaný souhlas*) is used by MAFIL to process the contact details of consenting volunteers. These contact details are provided by volunteers who wish to receive a copy of the experimental data obtained during their visit. Volunteers are not required to submit this information to participate in research experiments [12].

The designation of informed consent allows the data subject to *withdraw consent* to processing at any time [10, article 7(3)]. On the other hand, legitimate interest allows the data subject to *object* to the processing of their personal data.

If the data subject *withdraws* their consent, the data controller no longer has any ground to process the subject's personal data. On the other hand, if the data subject *objects* to processing based on legitimate interest, the data controller is barred from this processing unless they demonstrate "compelling legitimate grounds for the processing which override the interests, rights and freedoms of the data subject or for the establishment, exercise or defence of legal claims" [10, article 21(1)].

One legitimate interest cited by the laboratory is the limitation of MAFIL's liability. In the event of injury resulting from use of MAFIL's measurement equipment, the laboratory has a legitimate interest to prove that they were not negligent in their safety measures [8].

## 1.5 Aim of the thesis

This thesis aims

1. to evaluate the current state of the MAFILDB system and highlight the technologies in use by MAFILDB,

2. to analyze the laboratory's requirements for metadata de-identification, and
3. to propose and design an extension to MAFILDB implementing the above requirements.

The author shall then work to implement the extension for MAFILDB according to the above design. Requirements given by the laboratory include:

1. de-identification of personal data stored both in the MAFILDB and PACS systems,
2. integrity checks for personal data subject to de-identification,
3. availability of de-identification functionality via the MAFILDB web interface, in particular
   - 3.1. to safely manage the de-identification process from within MAFILDB,
   - 3.2. to review upcoming obligations to de-identify personal data, and
   - 3.3. to perform a simple audit of past de-identification processes [7].

## 1.6 Limitations of the thesis

This thesis does not aim to:

1. De-identify raw imaging data.

   The laboratory has chosen to indefinitely preserve raw imaging data for research purposes. Publicly shared data, however, such as contributions to biomedical datasets, undergo domain-specific de-identification processes, such as *de-facing* [8]. These topics are covered in Zemanovičová's 2022 thesis; see [13].

2. Respond to the withdrawal of informed consent.

   As mentioned in Section 1.4.2 'Legal basis for data processing' on page 4, volunteers may individually request the erasure of their contact information. The modality of data represents part

of a separate use case for MAFIL. This use case falls outside the scope of this thesis.

3. De-identify personal data in system backups.

   The MAFILDB system is subject to regular system backups. To aid in system recovery, backups of the MAFILDB system are preserved for a fixed time period following their creation. In particular, the laboratory retains daily backups of the system for up to one month, while monthly backups of MAFILDB remain in storage for up to a year.

   According to the laboratory, these backups do not fall within the legal requirements for de-identification [7]. This is justified by, in the author's opinion, the legal classification of system recovery as a legitimate interest. The author posits that system recovery holds substantial weight relative to the subject's freedoms, rights and interests.

### 1.6.1 Disclaimer of legal advice

This work includes statements by the author which may be interpreted as legal advice. However, the author neither intends nor is qualified to give such advice. Thefore, the author recommends that any interested reader instead seek professional advice on the topic.

Some system requirements presented by the author may have legal ramifications on the nature of the resulting system. Such requirements, however, draw directly from the laboratory's requirements, who are, in turn, advised by their legal team. In this context, any original research by the author merely represents an effort in due diligence; the author does not rely on original research for crucial design decisions.

# 2 Methods

## 2.1 Technologies

The following section describes a selection of the technologies in use by MAFILDB.

### 2.1.1 Django

*Django* is a Python-based web framework for developing full-stack web applications. The Django framework allows the developer to:

1. interact with a relational database using Django *models*, an object-oriented abstraction of an underlying relational database;
2. to define *views*, procedures from which users request information from the application, and
3. to design a hierarchy of *templates* used by the views to embed retrieved information into web-compatible markup [14].

### 2.1.2 DICOM and PACS

*DICOM*, short for *Digital Imaging and Communications in Medicine*, is a NEMA standard for the storage and transmission of medical imaging data [15]. *PACS*, short for *Picture Archiving and Communication System*, is a service capable of storing and transmitting such data [16].

#### DICOM data model

The DICOM standard defines several *Information Object Definitions* to represent "Real-World Objects", including *Patient*, *Visit*, *Study*, *Equipment*, *Series*, and others [15, PS3.3].

Each Information Object Definition carries properties, termed *Attributes* [15, PS3.1]. Attributes are each identified by an *Attribute Tag*, a pair of hexadecimal integers uniquely identifying the Attribute. [15, PS3.3]. The data type and format of each Attribute is specified by its *Value Representation* (VR) [15, PS3.5].

For example, the *Person Name* Attribute has a tag of `(0040,A123)` and a Value Representation of *PN*. Here, the *PN* Value Representation is a person name format shared with similar Attributes such as *Referring Physician's Name* and *Operators' Name* [15, PS3.6].

#### DIMSE Services

In DICOM, a client may make use of *DIMSE Services* provided by a server. To access these services, the client establishes an *Association* with the server. After an Association is established, both peers provide a list of supported services using a *Presentation Context*.

Within a given DIMSE Service, the designations *Service Class User* (SCU) and *Service Class Provider* (SCP) replace the traditional roles of a "client" and a "server". As such, a client peer may assume the role of a Service Class User for one service, and immediately become the Service Class Provider for another service. The peers no longer rely on the client-server architecture. To see an example illustrating this distinction, see: SCU & SCP Role example.

#### Service-Object Pairs

A DIMSE Service allows the client to request an operation on a given Information Object. However, each operation may behave differently based on the Information Object Definition which it is applied to. For this purpose, DICOM defines *Service-Object Pairs*.

Service-Object Pairs represent the application of a DIMSE Service on an Information Object Definition.[1] For example, by invoking the *C-MOVE* DIMSE Service, a client may ask a server to send all information about a patient to another server. This combination of operation and object would then be indicated by the following *SOP Class* [15, PS3.4]:

> *Patient Root Query/Retrieve Information Model - MOVE*
>
> SOP UID: *1.2.840.10008.5.1.4.1.2.1.2*.

1. More precisely, a given Service-Object Pair represents the application of a DIMSE Service on the *SOP Instance*, an abstraction of the data contained by Information Object Definitions.

#### SCU & SCP Role example

Suppose Client A and Server B participate in the following exchange: Client A requests the retrieval of an item from Server B using a *C-GET* request. Upon locating the item, Server B requests to store the item with Client A using *C-STORE*.

In the first request, Client A is the SCU and Server B is the SCP, as expected. However, the reverse is true for the second request; Client A is providing the C-STORE service to Server B. See Figure 2.1 'Illustration of SCU and SCP Roles' on page 9.

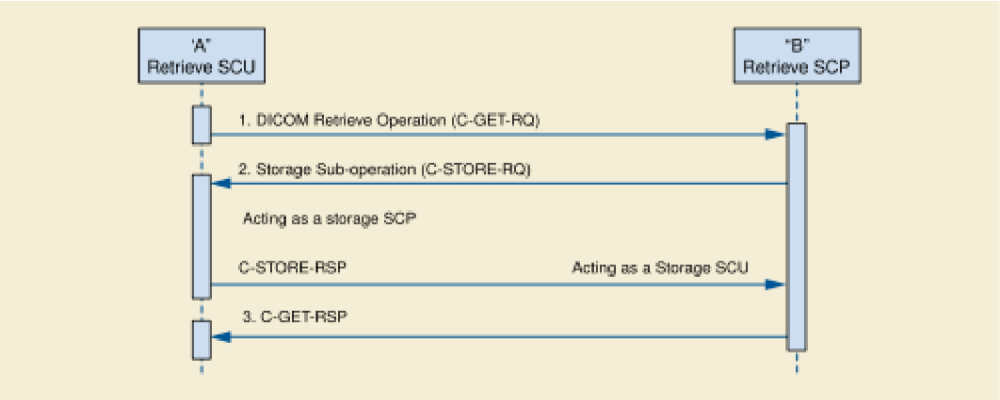


**Figure 2.1:** Illustration of SCU and SCP Roles

Image taken from Appendix C of PS3.4 of DICOM standard [15].

## 2.2 Analysis

### 2.2.1 Requirements elicitation

This section elaborates the requirements for de-identification in the form of question-and-answer pairs. These questions and their answers represent the author's requirements elicitation process with the author's consultants.

Question: How should de-identification be organized?

Answer: Data will typically be disposed once a project has concluded. De-identification should therefore be applied to a selection of the visits under a certain *Project*.

Question: The laboratory leaves some details out; in particular, which data entries are to be de-identified and how. Is a configuration file a suitable option to manage these rules?

Answer: These details depend on consensus between the laboratory and its legal team; however, preliminary rules are available. It is both acceptable to specifying these rules using code ("hard-coding") and to use a configuration file.

Question: Suppose that, due to a change in requirements, some data has to be de-identified more than once. Should de-identifications remain traceable?

Answer: In this case, it is acceptable to rely on system logs to trace the de-identification of data. Other functionality has higher priority.

### 2.2.2 MAFILDB data model

The following section describes the data structure used by the MAFILDB system.

#### MAFILDB application data model

The MAFILDB application data model is composed of a total of 36 types of entities. This section highlights a selection of entity types most relevant to this thesis. Each entity type is accompanied by a brief description. Altogether, the model captures a high-level view of ongoing research at MAFIL.

Relevant entity types include:

1. The *Project* entity.

   An entity of type *Project* represents a research project conducted at MAFIL.

2. The *Subject* entity.

Entities of type *Subject* refer to an individual partaking in a research study at MAFIL.

Notable attributes of the *Subject* entity include data collected as part of the registration procedure; see items 1 to 6 in Section 1.4.1 'Data collected by MAFIL' on page 3.

3. The *Visit* entity.

   Entities of type *Visit* represent a subject's participation in a research study on a given day. This typically includes several entities of type *Session*, each composed of a number of entities of type *Measurement*.

   Notable attributes include the *Visit* identifier and the date of visit.

#### PACS data model

While the MAFILDB application is responsible for tracking research more generally, it is the associated PACS system which stores raw data obtained from measurement devices[2].

The PACS data model used by MAFILDB includes the following entities:

1. The *Study* entity. The *Study* corresponds to MAFILDB's *Session* entity. A *Study* typically consists of several entities of type *Series*.
2. The *Series* entity. The *Series* maps to MAFILDB's *Measurement* entity. A *Series* typically consists of several entities of type *Instance*.
3. The *Instance* entity. An *Instance* is the smallest unit of raw data.

For a visual overview of entities from the MAFILDB and PACS data models, see Figure 2.2 'Entity-relation diagram of select MAFILDB entities' on page 12. Fundamentals of the PACS data model are better described in Section 2.1.2 'DICOM and PACS' on page 7.

2. In reality, the MAFILDB application does maintain a cache of raw imaging data from its associated PACS system, as mentioned in Section 2.2.3 'De-identification pipeline' on page 13. This is an implementation detail and is omitted here to aid legibility.

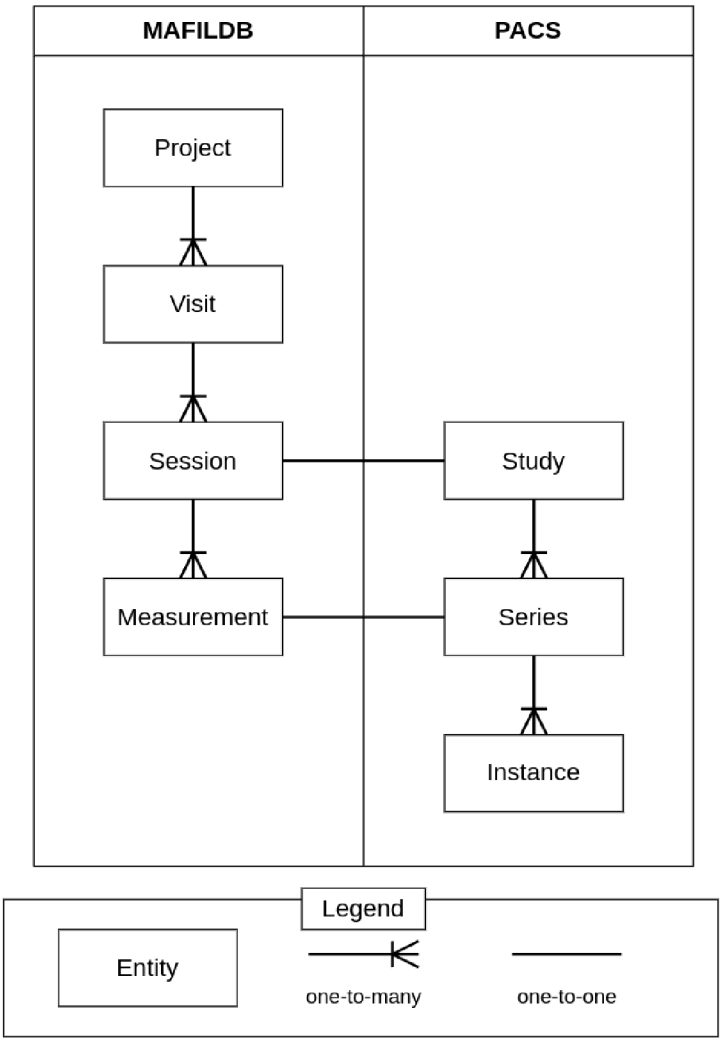


**Figure 2.2:** Entity-relation diagram of select MAFILDB entities

### 2.2.3 De-identification pipeline

To de-identify records stored across multiple systems, the de-identification procedure must take care to perform operations in the correct order.

In particular, personal data is stored in three databases: the MAFILDB application database, the PACS system, and a third system known as the PACS connector. The PACS connector, a bridge between MAFILDB and PACS, serves as a relational-database interface to the PACS. To improve performance, the PACS also connector caches metadata from PACS objects [17]. Therefore, it is also relevant for the successful de-identification of MAFIL's personal data.

The proposed pipeline to accomplish this goal consists of the following steps:

1. De-identify PACS *Study* objects associated with the given *Visits*.
2. Refresh cache in the PACS connector pertraining to the above *Study* objects.
3. Verify the successful de-identification of the *Study* objects by accessing the cache of the PACS connector.
4. Finally, de-identify the *Visits* in the MAFILDB application database.

### 2.2.4 Interfacing with PACS

To edit records in the PACS component of MAFILDB, the author has considered the following two approaches:

1. using the DICOM protocol to mutate records in place, and
2. executing database queries in the PACS via a vendor-specific interface.

#### De-identification using DICOM

The DICOM protocol offers methods to retrieve and store information [15]. However, this approach is not preferable for performance reasons: a de-identification pipeline may involve a large number of DICOM requests. Due to the overhead of the DICOM protocol, and, in the author's experience, the lack of documentation and/or support for

effective manipulation in Python-based DICOM libraries, the author has decided against this method.

#### De-identification using vendor-specific interface

As the name implies, a vendor-specific interfaces for a service is only available at the discretion of the software vendor. In the case of MAFIL's instance of PACS, this vendor is TatraMed Software s.r.o.

When contacted by MAFIL regarding the de-identifying of PACS records, TatraMed Software s.r.o. offered to produce an executable script to perform this de-identification. [7]. A software specification was prepared by the author for this script; see Appendix B.

## 2.3 Design

The following section describes decisions taken in designing the de-identifying system extension:

### 2.3.1 Interaction design

This section gives an overview of the MAFILDB user interface, and presents an interaction flow taken by users to de-identify volunteer records.

#### User interface overview

The main interactive elements in MAFILDB's web interface are the sidebar on the left-hand side and the view content itself. The sidebar is used for navigation between views, while the view content consists of a title and interactive elements belonging to the view.

An example view is the *Visits by project* view. This view allows the user to display a table of visits belonging to a certain project. The project visits are rendered once the user selects a project from the view's drop-down input.

#### Anonymization flow

The first user interface view added is the *Anonymize visits* view[3]. Using the *Anonymize visits* view, the user may create a new *Anonymization* record involving a selection visits from a given project. The *Anonymize visits* view reuses elements of the *Visits by project* view, including the project dropdown, and a modified table of visits; see Figure 2.3 'The *Anonymize visits* view' on page 16.

Upon creating an *Anonymization* record using the previous view, the user is redirected to the *Anonymization* view. This view then displays details of the *Anonymization* record. These details include the associated project, the user responsible for initiation, the state of the de-identification procedure, and relevant comments and timestamps; see Figure 2.4 'The *Anonymization* view (prototype)' on page 16.

To manage the de-identification procedure, the user may then choose from a list of available *user actions* (not pictured). This list is determined by the state of the *Anonymization* record. For a list of states and user actions, see Section 2.3.3.

#### Anonymization log and obligations

Two functional requirements are missing from the above flow: a log of *Anonymization* records, and a notification mechanism upcoming obligations to de-identify data. Both of these requirements are met by the *Anonymization dashboard* view, accessible from the navigation sidebar.[4] The *Anonymization dashboard* view contains two main elements: the *Priority* panel and the *Log* view.

The *Priority* panel displays a list of projects with upcoming obligations, i.e., projects with visits soon to be de-identified. Interacting with any project entry redirects the user to the aforementioned *Anonymizatioe visits* view for the given project. Next to each project in the *Priority* panel, a numeric badge highlights the number of such visits. A similar badge, counting the number of projects in the *Priority*

3. As implied in Section 1.2.2 'Anonymization as used herein' on page 2, the vocabulary of user interface elements derives from the more familliar term *anonymization* as opposed to *de-identification*.
4. Note that no prototype or mock-up is available for the *Anonymization dashboard* view, as it is currently only a design proposal.

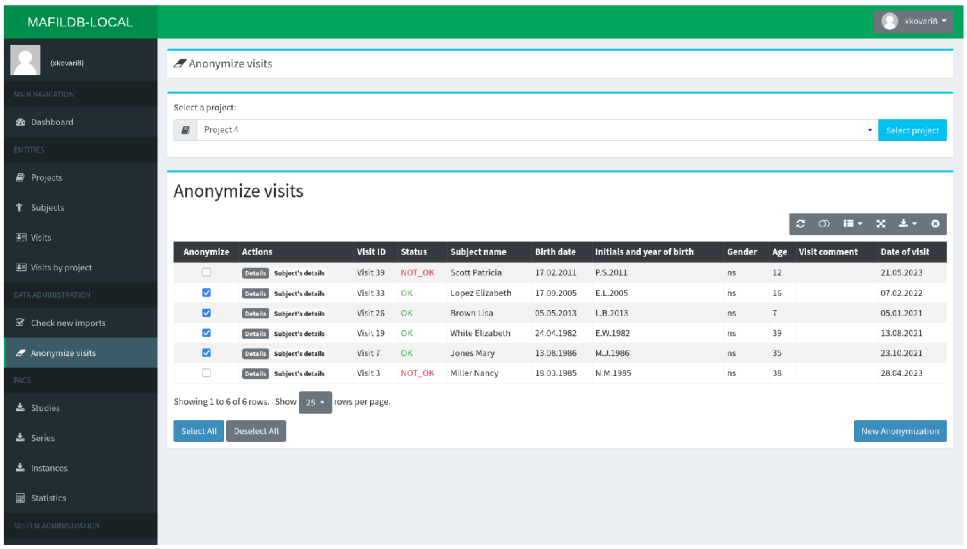


**Figure 2.3:** The *Anonymize visits* view

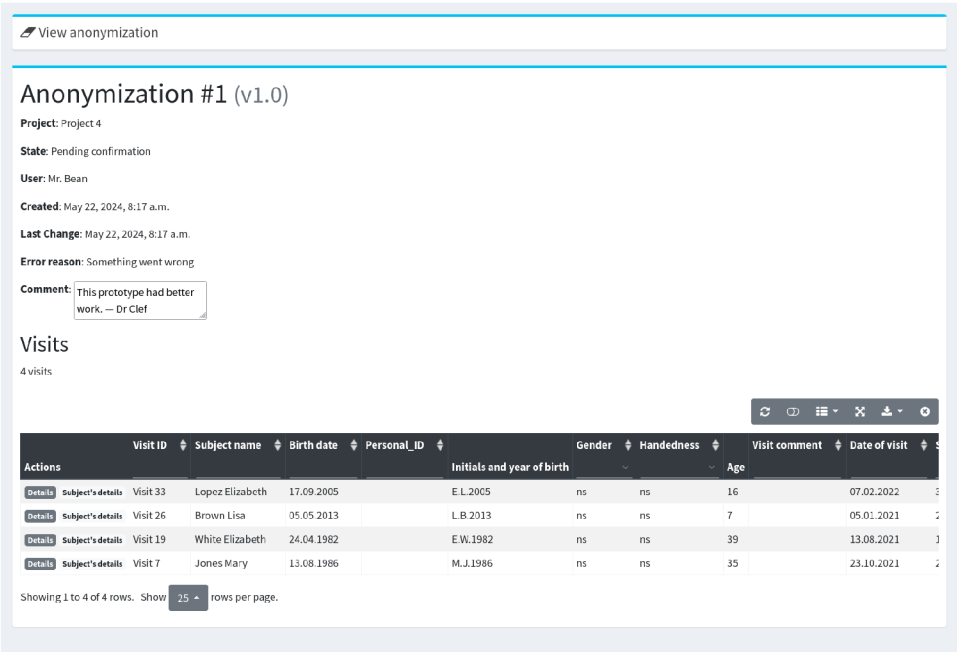


**Figure 2.4:** The *Anonymization* view (prototype)

panel, is also displayed next to the *Anonymization dashboard* view entry in the navigation sidebar.

The *Log* view is equally simple: it lists *Anonymization* records, with the most recent records listed first. Each record entry then serves to redirect the user to the *Anonymization* view, showing details for the given record.

### 2.3.2 Extending the data model

To accommodate the above interaction flow, the following extension to MAFILDB's data model was devised:

1. A new attribute, `data_processing_period_years`, is added to the *Project* entity.

   The `data_processing_period_years` attribute indicates the duration of the data processing period for a given project in years. In particular, this is the period in which MAFIL is permitted to process the personal data of a volunteer in the given project.

   The duration of the data processing period is given in MAFIL's volunteer registration form. Its value defaults to 10 years for old MAFIL projects, and 5 years for newer projects [7][9].

2. A new entity, *VisitDeidentification*, is added to the data model.

   The *VisitDeidentification* entity represents the *Anonymization* record mentioned in Section 2.3.1. The entity is associated with a number of *Visit* entities scheduled for de-identification, a *Project* entity, limiting the scope of the associated visits, as well as the *User* responsible for its creation.

   The attributes carried by *VisitDeidentification* hold information needed in the *Anonymization* view; namely, the creation date and date of last update, its procedure state, reason for error and user comment.

   The *VisitDeidentification* entity also includes an attribute named

   `deidentification_procedure_version`,

   indicating the version of de-identification procedure used at time of creation. This attribute is useful in cases where the procedure

of de-identification is updated in the course of development or as a result of changes to legal obligations: the attribute allows users to select for *Anonymization* records made prior to or following a certain change in procedure.

An overview of this data model extension can also be seen on Figure 2.5 'Extension to the MAFILDB data model' on page 19.

### 2.3.3 De-identification state

The *VisitDeidentification* entity carries state representing the progress of the *Anonymization* record, as mentioned in Section 2.3.2. This section defines this state in greater detail.

For a visual summary of the states and transitions of the *VisitDeidentification* entity, see Figure 2.6 'State diagram of the *VisitDeidentification* entity' on page 21.

#### States

The *VisitDeidentification* entity occupies exactly one of the following states:

1. The *Pending* state.

   The *Pending* state is the initial state for an *VisitDeidentification* entity. The state indicates that the *Anonymization* record has been prepared, but is still awaiting execution.

2. The *Executing* state.

   The *Executing* state signifies that the de-identification pipeline has started processing the associated visits.

3. The *Complete* state.

   The *Complete* state indicates that the de-identification pipeline has finished successfully. This state is a final state.

4. The *Error* state.

   The *Error* state signals that an error has occured during de-identification.

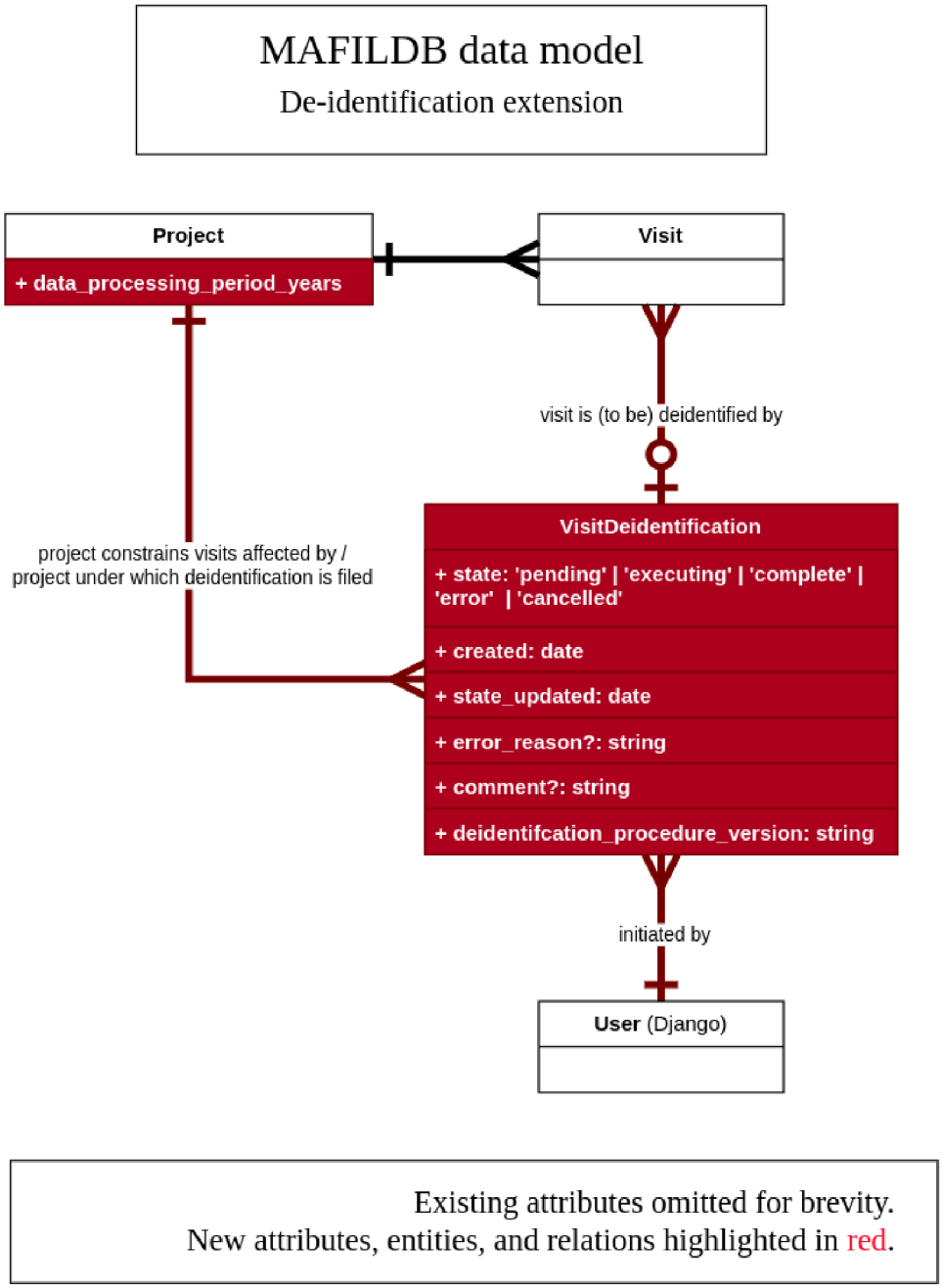


**Figure 2.5:** Extension to the MAFILDB data model

5. The *Cancelled* state.

   The *Cancelled* state represents a de-identification pipeline aborted by the user before execution, after an error, or by the system due to prolonged inactivity. This state is a final state.

#### Transitions

The state of the *VisitDeidentification* entity changes according to two classes of transitions: *user actions* and *automatic transitions*.

The following is a list of user actions:

1. The *Run* action.

   The *Run* action starts the execution of the de-identification pipeline for a *VisitDeidentification* in the *Pending* state, transitioning into the *Executing* state.

2. The *Cancel* action.

   The *Cancel* action cancels the planned execution of the de-identification pipeline for a *VisitDeidentification* in the *Pending* state, moving it into the *Cancelled* state.

3. The *Retry* action.

   The *Retry* action restarts the execution of a failed de-identification pipeline. This action applies to a *VisitDeidentification* in the *Error* state, transitioning into the *Executing* state.

4. The *Abort* action.

   The *Abort* action aborts the de-identification following a failed execution of the pipeline. This action applies to a *VisitDeidentification* in the *Error* state, moving it into the *Cancelled* state.

The following is a list of automatic transitions:

1. The *Success* transition.

   The *Success* transition is triggered by the successful completion of the de-identification pipeline. This action applies to a *VisitDeidentification* in the *Executing* state, moving it into the *Complete* state.

2. The *Failure* transition.

   The *Failure* transition is triggered by one of two events: a failure of the de-identification pipeline, or a timeout while waiting for its completion. This action applies to a *VisitDeidentification* in the *Executing* state, moving it into the *Error* state and setting an appropriate error message.

3. The *Pending Timeout* transition.

   The *Pending Timeout* transition is triggered by a timeout while waiting for a user to start a de-identification pipeline. This action applies to a *VisitDeidentification* in the *Pending* state, moving it into the *Cancelled* state and setting an appropriate error message.

4. The *Error Timeout* transition.

   The *Error Timeout* transition is triggered by a timeout while waiting for the a user to retry a failed de-identification pipeline. This action applies to a *VisitDeidentification* in the *Error* state, moving it into the *Cancelled* state and setting an appropriate error message.

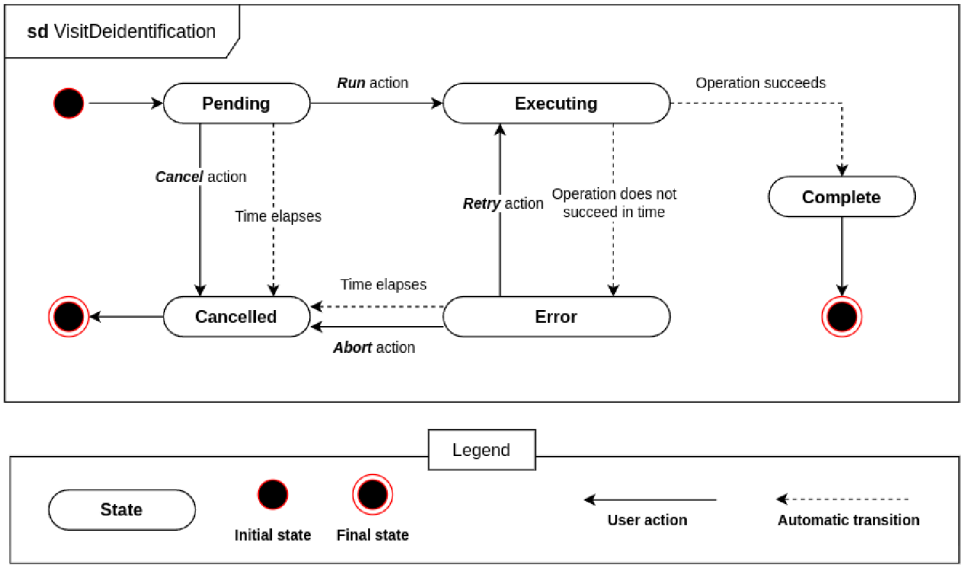


**Figure 2.6:** State diagram of the *VisitDeidentification* entity

### 2.3.4 Modular decomposition

Decomposition is crucial for an orderly implementation. This section briefly describes the decomposition of the above design into new or existing files and modules.

**User interface views** New user interface views were added to the `anonymization` subdirectory of the `mafil_db/mafil_db_app/templates` directory.

**Data model** Extensions to the data model were applied to MAFILDB's Django models file, `mafil_db/mafil_db_app/models.py`.

**Configuration** Configuration for the de-identification procedure was added to MAFILDB's YAML configuration file, `local/config.yaml`.

**Pipeline and automatic transitions** Another Django module named `deidentification` was allocated for the de-identification pipeline, as well as the automatic transitions of the *VisitDeidentification* entity. Automatic transitions triggered by timeout were modeled as periodic Django jobs.

## 2.4 Implementation

This section contains a brief summary of the tools and resources used to implement the proposed changes. To read about implementation results, see Section 3.1 'Implementation results' on page 23. To view the author's contributions, see Appendix A on page 27.

To track the implementation of the author's work, a *fork*[5] of the MAFILDB repository was created. The author's development environment was then deployed on a virtual machine provided by Stratus.FI, an instance of the OpenNebula service provided by the Faculty of Informatics.[6]

5. In Git, a fork of a repository is a copy of the repository: it is used for tracking changes to the base project and submitting the changes to the base repository for review.
6. See https://stratus.fi.muni.cz/.

# 3 Results

## 3.1 Implementation results

MAFILDB's de-identifying functionality is in the prototypal stage as of time of writing. Using feedback from MAFIL, the author further aims to develop their prototype into a working addition to the MAFILDB system. This work therefore also contributes to the elaboration of the laboratory's requirements.

### 3.1.1 Other contributions

The author has made the following contributions to MAFILDB's development tools:

1. Improvements to the performance and quality of mock data generation.
2. Shell script to re-initialize a developer instance of the project, allowing the application of new changes to Django model.
3. New local login flow on the main page of MAFILDB, used for developer logins where the Single sign-on (SSO) provided by CEITEC is unavailable.

The above contributions have been merged into the base MAFILDB repository [6]. To view all of the author's contributions, see Appendix A on page 27.

## 3.2 Practical application

The primary application of the author's extension of MAFILDB is the fulfilment of MAFIL's legal obligation, especially with respect to the EU's General Data Protection Regulation.

In particular, de-identification enables MAFIL to apply the *storage limitation* principle for the processing of personal data. In other words, data is "kept in a form which permits identification of data subjects for no longer than is necessary for the purposes for which the personal data are processed" [10, article 3].

The fulfilment of legal duties is not the only benefit of de-identification, however. In doing so, MAFIL also reduces the negative impact of any potential unwanted exposures of data.

Data breaches in the healthcare sector are a rising phenomenon in recent years, with increasing numbers of patient records compromised each year [18][19]. Although MAFIL is not part of a healthcare institution, the health data collected by the laboratory could still be of interest to attackers. For instance, measurements made using MAFIL's equipment can indicate the presence of health conditions; researchers may even mark measurements for incidental findings of such conditions. As such, de-identification also helps prevent the exposure of health data.

## 3.3 Suggested amendments

The following section lists possible amendments to the work presented in the thesis.

#### Support for subject-based de-identification

The legal basis of legitimate interest allows MAFIL to contest objections made to the processing of certain personal data, as mentioned in Section 1.4.2 'Legal basis for data processing' on page 4. Nevertheless, MAFIL may still choose to submit to these objections.

However, the author's extension of MAFILDB cannot easily accommodate this use case. To remedy this, an alternate de-identification flow rooted in a given *Subject* may be added to MAFILDB's user interface.

Currently, visits included in any de-identification record must be part of the record's assigned project. As such, the current data model is incompatible with new flow. The new subject-based de-identification flow would necessarily be accompanied by an update to the data model.

#### De-identifying contact information

As mentioned in Section 1.6 'Limitations of the thesis' on page 5, one modality of information omitted by de-identification: the contact in-

formation of study participants. This voluntary information is used to transmit experimental data to participants for personal use. Extending the current pipeline to involve the disposal of contact information would therefore prove useful to MAFIL.

# Appendices

# A Implementation and other code contributions

See the attached `implementation.zip` archive.

The archive contains files added by the author, included as-is, as well as changes made by the author to existing files, available as Git `.patch` files.

## B Software specification for TomoCon® PACS script

See the attached `tatramed-anonymization-script-spec.zip` archive.

The archive contains a `README.md` file containing the software specification, as well an example JSON request.

# Bibliography


[1] CEITEC. *Multimodal and Functional Imaging Laboratory*. cs.
URL: https://mafil.ceitec.cz/.

[2] K. Ruszová, M. Kojan, P. Kudlička, et al. *Bulletin sdílené laboratoře MAFIL vol. 8*. Jan. 2024.
URL: https://mafil.ceitec.cz/mafil-bulletin/.

[3] R. Chevrier, V. Foufi, C. Gaudet-Blavignac, et al. "Use and Understanding of Anonymization and De-Identification in the Biomedical Literature: Scoping Review". In: *J Med Internet Res* 21.5 (May 2019), e13484. ISSN: 1438-8871.
DOI: 10.2196/13484.
URL: http://www.ncbi.nlm.nih.gov/pubmed/31152528.

[4] C. A. Kushida, D. A. Nichols, R. Jadrnicek, et al. "Strategies for De-identification and Anonymization of Electronic Health Record Data for Use in Multicenter Research Studies". In: *Medical Care* 50 (July 2012), S82–S101. ISSN: 0025-7079.
DOI: 10.1097/mlr.0b013e3182585355.
URL: http://dx.doi.org/10.1097/MLR.0b013e3182585355.

[5] *Information technology — Security techniques — Privacy framework*. Standard. Geneva, CH: International Organization for Standardization, Feb. 2024.
URL: https://www.iso.org/standard/85938.html.

[6] T. Svoboda, A. Rošinec, J. Čillík, et al. *MAFILDB*. Software.
URL: https://gitlab.ics.muni.cz/mafil/db_mafil.

[7] T. Svoboda. Private communication. 2023–2024.

[8] T. Slavíček. Private communication. 2023–2024.

[9] MAFIL CEITEC. *Bezpečnostní formulář pro probandy* (*Safety form for MAFIL volunteers*).

[10] *Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC. (GDPR)*. 2016.
URL: https://eur-lex.europa.eu/eli/reg/2016/679/oj/eng
.

[11] European Comission. *What does 'grounds of legitimate interest' mean?*
URL: https://commission.europa.eu/law/law-topic/data-protection/reform/rules-business-and-organisations/legal-grounds-processing-data/grounds-processing/what-does-grounds-legitimate-interest-mean_en.

[12] MAFIL CEITEC. *Žádost o předání naměřených dat* (*Request for measurement data*).

[13] D. Zemanovičová. *Anonymization of Neuroimaging Data*. Bachelor's thesis. Brno, 2022.
URL: https://is.muni.cz/th/q7bf3/.

[14] *Django at a glance | Django documentation*.
URL: https://docs.djangoproject.com/en/4.2/intro/overview/.

[15] *NEMA PS3 / ISO 12052. Digital Imaging and Communications in Medicine (DICOM) Standard*. Version 2024b. Rosslyn, VA, USA: National Electrical Manufacturers Association.
URL: http://www.dicomstandard.org/.

[16] R. H. Choplin, J. M. Boehme, and C. D. Maynard. "Picture archiving and communication systems: an overview." In: *RadioGraphics* 12.1 (Jan. 1992), pp. 127–129. ISSN: 1527-1323.
DOI: 10.1148/radiographics.12.1.1734458.
URL: http://dx.doi.org/10.1148/radiographics.12.1.1734458.

[17] T. Svoboda. "MAFILDB technická realizace (MAFILDB technical realization)". In: FAIR data for CF MAFIL users (Workshop). Feb. 14, 2024.
URL: https://mafil.ceitec.cz/download/439.

[18] S. Liss and J. Y. Han. "Hacking healthcare: With 385M patient records exposed, cybersecurity experts sound alarm on breach surge". In: *Healthcare Dive* (Mar. 9, 2023).
URL: https://www.healthcaredive.com/news/cybersecurity-hacking-healthcare-breaches/643821/.

[19] "Healthcare Data Breach Statistics". In: *The HIPAA Journal* (Apr. 23, 2024).
URL: https://www.hipaajournal.com/healthcare-data-breach-statistics/ (visited on [visited on 05/22/2024]).