Electronic health card and telematics infrastructure

Specification
TI-Messenger Client

Note: This document is non-binding.

Version:	1.1.1
Revision:	682477
Last updated:	31/07/2023
Status:	released
Classification:	public
Referencing:	gemSpec_TI-Messenger-Client

Document information

Changes to previous version

Adjustments to this document compared to the previous version can be found in the table below.

Document history

Version	Last updated	Section/Page	Reason for change, special notes	Editing
1.0.0	01/10/2021		Initial version of the document	gematik
1.1.0	29/07/2022		Revision of the following features: – Accessibility of individual organisational units by means of function accounts – Opening of TI-Messenger for third-party systems by client-side interfaces for integration into practice management system – Quick finding of contact data by accessing the FHIR-based address book	gematik
	16/08/2022		Possibility of some kind of access control for Org Admin	gematik
1.1.1	31/07/2023		Integration of TI-Messenger Maintenance 23.1 / VZD FHIR Maintenance_23.1	gematik

Contents

1 Classification of document

1.1 Objective

This document defines the specifications for the first expansion stage of TI-Messenger. This expansion stage is defined by ad-hoc communication between healthcare organisations. Particular attention will be paid to ad hoc communication between service providers and between service provider institutions. Specifications on the user group of insured persons and requirements for health insurance organisations will be taken into account in the second stage of the TI-Messenger expansion and therefore not further considered in this document.

This specification defines the requirements for production, testing and operation of the TI-Messenger client product type. The TI-Messenger client provides the user with the required functionality for secure ad-hoc communication with other participants. Interfaces to be used by the TI-Messenger client result from the communication relations with the TI-Messenger specialist service and the VZD-FHIR directory. This document describes the use of these interfaces for secure ad-hoc communication and the functionalities required for this purpose. Interfaces used by the TI-Messenger client are defined in the corresponding product type specifications.

1.2 Target group

The document is aimed at manufacturers of the TI-Messenger client product type as well as suppliers who operate this product type [gemKPT_Betr]. All manufacturers and suppliers of TI applications using interfaces of the component, or exchanging data with or processing the TI-Messenger client product type, must also consider this document.

1.3 Coverage

This document contains normative provisions on the telematics infrastructure of the German healthcare system. The validity period of the present version and its application in approval or acceptance procedures is defined and disclosed by gematik GmbH in separate documents (e.g. gemPTV_ATV_definitions, product type profile, supplier type profile, etc.) or web platforms (e.g. gitHub, etc.).

Intellectual property / Patent legal notice

The following specification has been created by gematik solely from a technical point of view. In individual cases, it cannot be excluded that the implementation of the specification interferes with the technical property rights of third parties. It is solely the responsibility of the supplier or manufacturer to take appropriate measures to ensure that the products and/or services offered by it on the basis of the specification do not violate the property rights of third parties and to obtain the necessary permissions/licences from the affected property right holders. Gematik GmbH therefore assumes no warranty whatsoever.

1.4 Demarcation

The document specifies the interfaces provided (offered) by the product type. Used interfaces, on the other hand, are described in the specification of the product type that provides this interface. Reference is made to the corresponding documents (see also annex, Section 6.5 – Referenced documents).

The complete requirements for the product type result from further concept and specification documents, which are recorded in the product type profile of the TI-Messenger product type.

1.5 Methodology

The specification is written in the style of an RFC specification. This means:

- **The entire text in the specification is to be considered binding for the manufacturer of the TI-Messenger client product as well as for the operating provider according to [gemKPT_Betr] and is to be considered as an approval criterion for both the product and the supplier.**
- The binding nature SHOULD be indicated by the keywords MUST, MUST NOT, SHOULD, SHOULD NOT, MAY, written in capital letters and corresponding to [RFC2119].
- As in the example sentence "An empty list MUST NOT contain an item." the phrase "MUST NOT" would be semantically misleading (if not one, maybe two?), "An empty list MUST NOT contain any items." is used in this document instead.
- The keywords MAY also be completed with pronouns in capital letters if this improves the language flow or clarifies semantics.

Use cases and acceptance criteria as expressions of normative requirements are examined and verified through tests as a basis for obtaining approval. They have a unique, permanent ID, which SHOULD be used as a reference. The tests are carried out against a reference implementation performed by gematik.

Use cases and acceptance criteria are presented in the document as follows:
<ID> – <Title of use case / acceptance criteria>
Text / Description
[<=]

The individual elements describe:

- **ID**: a unique identifier.
  - In a use case, the identifier consists of the string 'AF_' followed by a number,
  - The identifier of an acceptance criterion is assigned by the system, e.g., the string 'ML_' followed by a number
- **Title of use case / acceptance criteria:** A title that summarises the content
- **Text/description**: Detailed description of the content. Can contain tables, illustrations and models in addition to text

The use case or acceptance criteria include all contents listed between the ID and the text mark [<=].

The proof of fulfilment of the use case necessary for obtaining an approval is specified in the respective profiles, in which each use case is listed. Acceptance criteria are usually not listed in the profile.

Reference to open points

Open point: The section will be supplemented in a later version of the document.

2 System overview

The TI-Messenger client is installed as an application (or embedded in existing applications) on the end device of an actor and enables secure, message-based communication with other actors of the TI-Messenger service. The TI-Messenger client follows the open standards of the communication protocol Matrix and synchronises Matrix home server JSON objects defined by the Matrix Foundation that are provided as part of the messenger service of a TI-Messenger specialist service.

The communication between the actors of the TI-Messenger service takes place end-to-end encrypted in rooms. The messages are created on the respective TI-Messenger client and sent end-to-end encrypted. The sent messages are stored encrypted on the respective Matrix home server. The key required for decryption is only shared with verified end devices within the respective room. The participating Matrix home servers cannot decrypt the messages.

Communication between a TI-Messenger client and a TI-Messenger specialist service takes place via the Messenger Proxies. TLS termination of connections from the TI-Messenger clients takes place on the Messenger proxies. The TI-Messenger proxies only allow the registration of an actor with approved TI-Messenger clients. This is made possible by the fact that the `client_id` stored on the client is checked by the messenger proxy during login. In addition, during the login process, the TI-Messenger client checks whether it is an approved Matrix home server on the auth service of the VZD-FIR directory.

In the following figure, all involved components of the TI-Messenger architecture are shown in simplified form. The TI-Messenger client shown in green in the figure shows the component described in this specification.

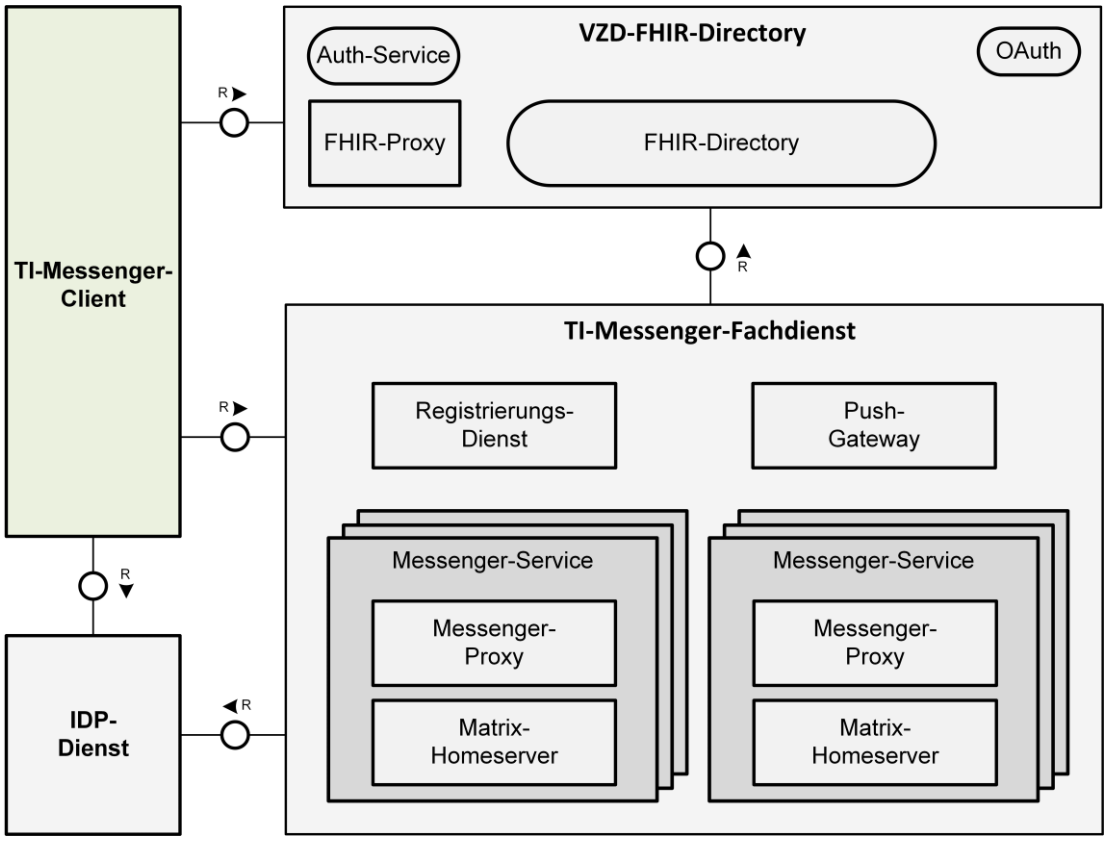

Figure 1: System overview (simplified presentation)

3 System context

The following section puts the TI-Messenger client into the system context of the TI-Messenger service.

3.1 Neighbouring systems

The TI-Messenger client enables stakeholders to interact with the TI-Messenger service. For interaction with the TI-Messenger service, other systems are required by the TI-Messenger client. The following figure shows the neighbouring components of the TI-Messenger client:

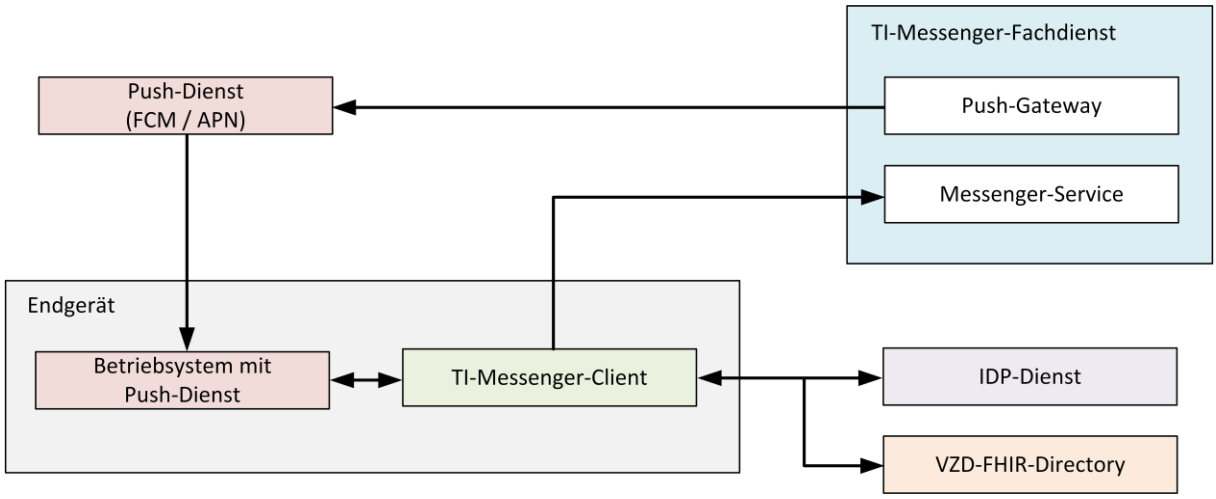

Figure 2: Neighbouring components of the TI-Messenger client

The neighbouring systems of the TI-Messenger client named in the figure are sufficiently described in the [gemSpec_TI-Messenger-Service] and [gemSpec_TI-Messenger-FD]. For the classification of components in the context of the TI-Messenger client, these are briefly explained below.

Table 1: Overview of components and their functions

Component	Function
Push gateway	• Forwarding push notifications to push services on the internet
Push service	• Push services (e.g. FCM/APN) are services of push providers and are required for native support of push notifications on mobile devices.

Component	Function
Messenger service	• Provides for the TI-Messenger client interfaces according to [Client Server API]. • Schedules the TLS connection of TI-Messenger clients. • Checks requests from TI-Messenger clients. • Provides an interface for maintaining the personal release list. • Provides Matrix OpenID token for TI-Messenger clients.
IDP service	• Issues ID_TOKEN, for example, to authenticate itself on a Matrix home server using OpenID Connect.
VZD-FIR directory	• Issues access-tokens (search-accesstoken and owner-accesstoken) • Reading or writing FHIR resources

3.2 Characteristics of TI-Messenger clients

3.2.1 User groups

According to the architecture of the TI-Messenger service, a distinction is made between two types of TI-Messenger clients. The distinction is exclusively drawn from the stakeholder point of view. The two forms are described below.

3.2.1.1 TI-Messenger client with administration functions (Org-Admin client)

The TI-Messenger client with administration functions is a client for administrators of an organisation. This is also referred to as the Org Admin client in the TI-Messenger context. The Org Admin client is used for comfortable management of messenger services at a TI-Messenger specialist service. The Org Admin client allows you to provide or edit FHIR resources on behalf of the organisation. Administrators of an organisation can also use the Org Admin client to manage users and devices on the respective messenger service. In addition, it is possible to verify or validate registered device sessions on the messenger service via the Org Admin client. This means, for example, that an actor in the "Org-Admin" role can unsubscribe a TI-Messenger client of an actor if necessary. In addition, function accounts can be administered via the Org-Admin client according to [gemSpec_TI-Messenger-Dienst#Funktionsaccounts] for cross-communication within an organisational structure of the TI-Messenger specialist service.

3.2.1.2 TI-Messenger client for actors

The TI-Messenger client for actors supports most of the functions of a Matrix messenger defined by the Matrix specification. Actors can use this client to send and receive end-to-end encrypted chat messages. Within the chat rooms, you can access chat processes or

exchange media. It is also possible for actors to verify their own devices and devices of interlocutors and to search the VZD-FHIR directory for organisations in order to start a new chat conversation with an organisation. The manufacturer is free to choose how the interface is designed. For example, it is possible to organise chat rooms for different purposes. Actors in the "User-HBA" role also have the option of adding their own MXID as the contact address of the existing *Practitioner* entry on the VZD-FHIR directory. Entering the MXID allows you to search for other actors listed on the VZD-FHIR directory in the "User-HBA" role and allows other actors to find you.

Note: The two forms described above MAY also be integrated into a TI-Messenger client. The type of implementation is the responsibility of the respective TI-Messenger client manufacturer.

3.2.2 Platforms

TI-Messenger clients have different requirements for security, data protection and functionality depending on the platform (mobile/stationary). The platforms to be supported are described in more detail below.

3.2.2.1 TI-Messenger client for mobile scenarios

This is a TI-Messenger client application specifically designed for use on mobile devices (e.g. Android/IOS). The provision MAY take place as a native mobile application or as an integration into already existing applications. The mobile application MUST use the operating system related security features. The application MUST ensure that data is stored separately and encrypted from the file system. Unauthorised access by third parties MUST be actively prevented (e.g. by PIN query when opening the application).

3.2.2.2 TI-Messenger client for stationary scenarios

This is a TI-Messenger client application that has been specially developed for use on stationary devices (e.g. Windows/MacOS). The provision MAY take place as a standalone solution or as an integration into already existing solutions.

3.2.2.3 TI-Messenger client as web application

It is also possible to run the TI-Messenger client as a local web application in a web browser. The encryption and decryption MUST be performed locally in the browser on the end device. It MUST also be ensured that unauthorised access by third parties is actively prevented when using a local web application (e.g. by invalidating the session or by actively logging out).

3.2.3 Further specifications

Each provider of a TI-Messenger MUST offer both the TI-Messenger client for actors and the TI-Messenger client with administration functions (Org Admin client) to organisations that receive a messenger service from the provider.

4 General specifications

4.1 Data protection and security

To ensure data protection and security within the framework of the TI-Messenger service, the requirements to be observed for the TI-Messenger client are described below. Requirements that are ensured by other system components are not further listed here.

Note: For data protection requirements for the TI-Messenger service, reference is made to the opinion of the conference of the independent data protection supervisory authorities of the Federation and the states according to [DSK2021]. The contents of the statement are summarised in simplified terms in the requirements:

- *A_22715 – Manufacturer's requirements declaration from the conference of the independent data protection supervisory authorities,*
- *A_23613 – Forced deregistration and blocking of users and*
- *A_22955-0X – Protection of stored data received.*

A_22715 – Manufacturer's requirements declaration from the conference of the independent data protection supervisory authorities

- The TI-Messenger client MUST provide data protection information that is clearly identifiable to the actor.
- The TI-Messenger client MUST support a general and selective deletion function.
- The TI-Messenger client MAY implement a feature to obscure sections of image captures.
- The TI-Messenger client MUST ensure that all parts are sent when sending messages or documents in parts.
- The TI-Messenger client MUST inform the user about shipping errors.
- The TI-Messenger client MUST NOT collect location data permanently.

[<=]

A_22955-01 – Protection of stored data received

For received and sent data that has not been explicitly exported by the user for use in other contexts, the TI-Messenger client MUST ensure that, if stored, it can also only be read by the TI-Messenger client and after authentication of the respective user, provided that the TI-Messenger client is a desktop application. If, on the other hand, it is an application for mobile platforms – for example, for smartphones or tablets – which implement and use inherent security mechanisms to protect against unauthorised access, the required security performance MAY instead be provided by the runtime environment of the application, for example, by encrypting the memory on the operating system side and isolating the applications. Unauthorised access to stored received data by bypassing the TI-Messenger client, for example, by access via file system, MUST be excluded.

[<=]

A_24003 – Volatility of the collection of location data
When collecting location data, it MUST be ensured that this collection is triggered exclusively by a human user and that after termination of the use case collecting the location data, it is deleted again from the client context or not persisted with in the first place.
[<=]

A_23114 – App lock TI-Messenger client
TI-Messenger clients MUST use at least one 6-digit PIN when unlocking (the app or the device). Alternatively, biometry, strong passphrase or fido tokens are allowed. If biometry is selected, it MUST meet the specifications from [BSI-TR-03166] Sections 2.3.1.5 or 2.3.1.6. After each logout, user change, closing of the application or at the latest 12 hours after the last unlocking, the new unlocking MUST be performed by the actor.
The TI-Messenger client MUST check if a device lock is active. If a compliant device lock is activated, then no additional app lock must be provided. If a non-conforming device lock is activated, then a compliant app lock must be provided.
For a TI-Messenger client module integrated into a third-party system (KIS, PVS, AVS, etc.), an existing block of the higher-level system can be re-used.
App locks for TI-Messenger clients and integrated TI-Messenger client modules MUST be disabled by the actor.
Browser-based TI-Messenger clients do not require an app lock. The browser-based web client MUST have a lock that automatically logs out after prolonged inactivity. The required duration of inactivity MUST be configurable by the actor and set to one hour by default.
[<=**]**

A_22717 – Preventing the creation of screenshots
TI-Messenger clients for mobile scenarios MUST prevent screenshots and screen capturing if allowed by the operating system or make clear to actors after creating a screenshot that it cannot be protected by the TI-Messenger client. This function MUST be deactivated by actor opt-out. If the function is deactivated, operators MUST be alerted to the risks of sensitive content screenshots.
[<=]

A_22718-01 – Multitenancy of TI-Messenger clients
In the case of shared endpoints, TI-Messenger clients MUST prevent an actor of the TI-Messenger client from being able to access data or functions of another actor of the TI-Messenger client on this device. The TI-Messenger client MUST NOT rely on the operating system to segregate users to prevent access to other actors' data, as such functionality is not necessarily used. Instead, the TI-Messenger client itself MUST ensure the separation of data of the logging-in users.
[<=]

A_22720 – Information obligation regarding dangers of insecure end devices
Actors of a TI-Messenger client as a web application MUST be informed in a notice text about the dangers of using hardware that is not under the control of the actor. In addition to shared end devices without IT security monitoring, this particularly applies to publicly available end devices. The actor MUST be advised not to use the TI-Messenger client on such devices.
The TI-Messenger client MUST inform the actor in a notice text about the dangers of operating the TI-Messenger client on hardware that is not under the actor's control.
The test regulations according to [BSI Frontend] must be considered.
[<=**]**

A_22721 – Key sharing between an actor's devices
TI-Messenger clients MUST implement the Matrix default SHOULD "Key sharing only for verified devices" as a MUST.

Note: The requirement is essential to enable message content synchronisation between several devices of an actor via the key sharing functionality provided by Matrix.
[<=**]**

A_22722 – Key sharing between devices within a chat room
TI-Messenger clients MUST have a function to make key-sharing requests to other devices and to accept or reject key-sharing requests from other devices within a chat room.
[<=]

A_22723 – Sending files via Matrix
The following applies to sending files according to the Matrix specification via the TI-Messenger client:

- TI-Messenger clients MUST use encryption for transferred content.
- TI-Messenger clients MUST be able to send files of at least 100 MB in size.
- TI-Messenger clients MUST have a size limit on content to be sent.
- TI-Messenger clients for stationary scenarios MAY have an interface and functionality that allow received and decrypted files to be transmitted to an interface of known virus scanners for malware check purposes before they are processed. Files that do not successfully pass such a check SHOULD be rejected. If a file is rejected, the actor MUST be informed about it as well as the reason.
- TI-Messenger clients MUST notify actors of their inspection status and potential hazards in case a file check fails.

If TI-Messenger clients have a function to display documents directly via the TI-Messenger client without using third-party software, they MUST prevent the execution of active content. This function MUST also make it possible to view associated metadata without opening or downloading the file itself.
The TI-Messenger client MUST inform the actor that documents can contain malware and what measures the actor can take to protect themselves.
The TI-Messenger client MUST implement malware protection measures in the documents when directly displaying document content. **[**<=**]**

Note:
Proposed measures to protect against malware

- *Verify that the document format and its content matches the specified document type in the metadata.*
- *Before displaying a document in the TI-Messenger client, special and meta characters in the document must be replaced with the correct escape syntax for the respective display software.*
- *Operate the display software of the TI-Messenger client in a sandbox.*

A_23115 – Device integrity check
TI-Messenger clients for mobile scenarios MUST check if the device is rooted. If this is the case, a warning MUST be displayed to the user and the sending of attachments MUST be prevented.

When using TI-Messenger clients based on the Android operating system, Safetynet MUST be used for the integrity check.
[<=]

A_22724 – Content compartmentalisation in TI-Messenger client
TI-Messenger clients for mobile scenarios MUST ensure that locally stored data is stored in a protected storage area on the end device.
To this end, clients SHOULD compartmentalise the memory that the TI-Messenger client occupies for user data. For this purpose, the resources provided by the operating system are usually sufficient.
Web clients MUST ensure that sensitive data in the browser (e.g. OLM keys, ACCESS_TOKEN) cannot be read by other applications.
[<=]

A_23130 – Use of data by third-party systems
To enable seamless integration of TI-Messenger clients into, e.g., primary (PVS, ZPVS, HIS, AVS etc.) or archive systems, TI-Messenger clients MAY provide an interface to access their data by means of third party systems.
The TI-Messenger client MUST ensure that when using such function, actors are appropriately informed that they are moving data from the protected area of the TI-Messenger client. In this context, appropriately means that information is provided about which data is forwarded to which third-party system.
[<=]

A_22725 – Safety critical updates
TI-Messenger client manufacturers MUST ensure that stakeholders are informed about the release of updates for their TI-Messenger clients. In the case of security critical updates, they MUST ensure that after a suitable period of time, further use of the TI-Messenger client is not possible without a prior security update. For this purpose, a client-side lock is sufficient instead of proof towards the Matrix home server. The option of continuing to import updates MUST continue in this case. Actors MUST be properly informed that they need to install security-critical updates to continue using the TI-Messenger client.
The manufacturer of the TI-Messenger client MUST inform gematik when a new product version is released and provide a statement on security suitability.
[<=]

A_22792 – Device verification, cross-signing and SSSS for TI-Messenger clients
TI-Messenger clients MUST support the Cross-Signing and Secure Secret Storage and Sharing (SSSS) functions for device verification. The specification according to [Client-Server API#Sharing keys between devices] MUST be followed.
[<=**]**

A_22793-01 –End-to-end encryption
TI-Messenger clients MUST support end-to-end encryption based on OLM/MEGOLM. For this, the specification according to [Client-Server API#End-to-End Encryption] MUST be followed.
TI-Messenger clients MUST use this encryption to send messages. Communication MAY be unencrypted in public rooms if the user is explicitly informed of this. To this end, the TI-Messenger client MUST identify public rooms as such through appropriate UI elements so that the user is aware of them – that is, the unencrypted communication and the public nature of the room. Beyond the identification of the room, the TI-Messenger client MAY provide information about the public nature and non-encryption of the room as soon as it is entered.
[<=]

A_22794 – Explicit prohibition of profiling for TI-Messenger clients
TI-Messenger client manufacturers and vendors MUST NOT collect data for profiling purposes. This particularly concerns monitoring which actors communicate with which other actors.

Note:
Pursuant to Section 331(2) of the German Social Code (SGB V), gematik may specify data that providers of components and services must disclose or transmit to gematik, insofar as this data is required to fulfil gematik's legal mandate to monitor operations to ensure the security, availability and usability of the telematics infrastructure. Only personal data required for this purpose may be collected by providers and manufacturers as an exception to the profiling ban and used exclusively for that purpose.
[<=**]**

A_22795 – Inserting and storing keys and tokens
TI-Messenger client manufacturers MUST ensure that keys and tokens are safely placed into the TI-Messenger client.
TI-Messenger client manufacturers MUST technically ensure that keys and tokens cannot be transferred to memories other than those provided by the TI-Messenger clients or the SSSS [Matrix SSSS] of the involved home server.
[<=**]**

A_22796 – Use of TLS to communicate with specialist service and VZD-FHIR directory
TI-Messenger clients MUST be able to connect to other components of the TI-Messenger service via TLS. For this purpose, the specifications of [gemSpec_Krypt] apply.
[<=**]**

A_22797-01 – Automatic deletion
TI-Messenger clients MUST have an automatic deletion function. All locally stored rooms and room contents are affected by this automatic deletion function if (1) the user is no longer a member of the room, or (2) the room has been deleted on the server side.
[<=]

A_23112-02 – Function to track deletions and modification of TI-Messenger content
TI-Messenger clients MUST have a message-based deletion function that allows actors to not only delete their own messages from their own TI-Messenger client, but also in the room state. If a deletion has been performed by another client, the deletion of the message MUST also be performed and marked on all other clients involved in the communication. The tag MUST contain the deleting actor, the date and time of the deletion.
[<=**]**

A_22798 – Privacy by Default
TI-Messenger clients MUST always use the most data protection friendly preset as their default.
[<=]

A_22799-01 – Protection against OWASP Top 10 Mobile Risks
Manufacturers of TI-Messenger clients for mobile scenarios MUST ensure for the TI-Messenger mobile clients they offer that the client is resistant to the risks identified in the current and the two previous OWASP Mobile Top 10 Report(s). Additionally, the specifications according to [BSI Frontend] are implemented analogously for the TI-

Messenger client, with the exception of the following points:

Point	Reason
O.Arch_6	The actual security gain is disproportionate to the effort.
O_Auth_4	This measure will be added to later TI-Messenger specification versions as the Zero Trust model is introduced.
O.Sess_1 to _5	The session handling of Matrix deviates too far from the assumed status to implement these measures as intended.
O.Data_7	This measure is diametrically opposed to the security goals of the TI-Messenger.
O.Ntwk_9	This measure is not appropriate in terms of data protection law.
O.Resi_4 to _5	This measure creates user problems that do not outweigh the small security gains and the rather low risk of non-compliance.
O.Resi_7 to _8	This measure creates user problems that do not outweigh the small security gains and the rather low risk of non-compliance.

Note: The test depth of the requirements is always CHECK for the security assessment. This test depth also applies to the requirements for which the test depth EXAMINE is prescribed in the document.
[<=**]**

A_22800 – Minimise security risks of software libraries
The TI-Messenger client MUST implement measures to minimise the impact of undetected vulnerabilities in used software libraries.

Note: Example measures can be found in [OWASP Proactive Control#C2]. The selected method must have the same effectiveness as the encapsulation according to [OWASP Proactive Control#C2 Point 4].
[<=]

A_22801 – Secure procurement of foreign program components
The manufacturer MUST source the software components of the TI-Messenger client that are not developed by the manufacturer or commissioned for development (e.g. TLS libraries or Matrix implementations) from known and trusted sources.
[<=]

A_22802-01 – Secure software distribution
The manufacturer of a TI-Messenger client MUST inform actors about the trusted sources from which actors can obtain the TI-Messenger client and how they can recognise the trustworthiness of the source. The manufacturer MUST ensure that the actor can verify the authenticity of the trusted source of supply when first purchasing a TI-Messenger client. The TI-Messenger client MUST ensure that updates are only obtained from known and trusted sources after technically successful verification of the authenticity of the source.

Note: There are configurations in which updates are not loaded and applied by the client, and the client therefore does not have the necessary degree of control to ensure that they are obtained from trustworthy sources itself. Examples of this are software distribution via the Apple App Store and Google Play Store.
[<=]

A_22804 – Data protection compliant tracking
The TI-Messenger client MUST NOT use advertising tracking.
In the following, tracking is also understood as usability tracking as well as crash reporting.
The TI-Messenger client MUST ensure, if it implements tracking functions, that no security features such as a device ID or security related data is included in the transmitted tracking information.
The data protection officer for the TI-Messenger client MUST process and evaluate any collected tracking data of the TI-Messenger client itself and not have it performed by a third party provider.
The TI-Messenger client MUST ensure, if it uses tracking functions, that tracking data does not contain data that directly identifies individuals.
The TI-Messenger client, if using tracking features without the actor's consent, MUST ensure that the tracking data

- only relates to one client usage (from the user's first interaction with the client until the client is closed or until the inactivity timeout) and is not linked to other client usages of the actor,
- contains neither personal nor pseudonymised personal data,
- does not contain user-specific IDs or device-specific IDs of the user devices,
- does not allow conclusions to be drawn about insured persons, their representatives, service providers or payers, in particular conclusions based on user behaviour over time or across client uses,
- cannot be de-anonymised by linking personal data from other sources.

The TI-Messenger client, if it uses tracking functions without the consent of the actor, MUST inform the actor about tracking in the TI-Messenger client in an understandable and easily accessible form and in a clear and simple language before the tracking data is collected.
The TI-Messenger client MUST randomly generate new usage identifiers for each client use if it uses tracking functions without the actor's consent. The actor MUST be able to force the regeneration of these identifiers at any time.
The TI-Messenger client MUST technically ensure that if it implements tracking functions with tracking data linking of multiple client uses, these tracking functions are disabled by default when installing the TI-Messenger client and only enabled after explicit consent by the actor (opt-in). Refusal to use such functions must not restrict the standard functions of the TI-Messenger client.

If such functions are implemented, the following consent information MUST be displayed to the actors prior to consent to the activation of these tracking functions in an understandable and easily accessible form and in a clear and simple language:

- which data is collected by the tracking functions,
- the purposes for which the data is collected;
- what information is obtained from the analysis of the collected data and whether it would be possible to draw conclusions about the state of health of the actor,

- who the recipients of the data are,
- how long the data will be stored.

These functions MUST NOT be activated until explicit consent has been given by the actors and it MUST be possible for them to deactivate it at any time.
A reference to the terms and conditions of TI-Messenger client use is NOT sufficient. An understandable and easily accessible form is an explicit short explanation in simple and non-legal language, which is displayed directly in the TI-Messenger client.
The client MUST NOT repeatedly ask the actor in order to force consent through harassment. After a one-time rejection by the actor, any display of the dialogue MUST be explicitly initiated by the actor.
[<=]

A_22806 – No write access for TI-Messenger clients to room states
TI-Messenger clients MUST prevent actors from being able to enter additional information in room states.
[<=]

A_22937 – Sole use of audited encryption
TI-Messenger clients MUST use an audited and sufficiently secure implementation of OLM/MEGOLM to encrypt messages. If a different implementation is used than the one intended by gematik, the manufacturer MUST provide proof of security, e.g., in the form of a commissioned audit.**[<=]**

Note: gematik has commissioned an audit for the OLM/MEGOLM rust implementation Vodozemac in cooperation with the Matrix Foundation. Based on this audit, Vodozemac is designated as the implementation envisaged by gematik.

A_22938 – Sole connection to valid messenger services
TI-Messenger clients MUST only offer valid messenger services belonging to the selected provider when configuring the messenger service to be used.
[<=]

A_22964-01 – Access protection for administration functions
TI-Messenger clients that are used both as a client for communication and as an Org Admin client MUST use separate user interfaces to provide the functionalities for the respective role, which display the information and provide functions relevant for the respective purpose. In this sense, buttons needed to perform the role of Org Admin MUST NOT be available in the context of communication as an actor in the role of "User/User-HBA". Conversely, the actor in the role of "Org Admin" MUST NOT be hindered by UI elements for communication with other users. To allow an actor to access administration functionalities in the “Org Admin” role, the TI-Messenger MUST force a new authentication of the actor towards the TI-Messenger client.
[<=]

A_23774 – Deadlines for reminding about data cleansing
TI-Messenger clients MUST have a configurable room and room content deletion reminder time period that offers to delete all data older than the configured time period. The time period MUST be preset to six months and refers to the timestamp of the creation of the last message in a room. If this period has elapsed for a room, the client MUST notify the user and recommend deletion of the room and its contents. If the user agrees, they are removed from the room, which also results in a notification from the server.**[<=]**

A_23612 – Password-based key backup
If the TI-Messenger client offers the use of passwords for the protection of cryptographic material (as part of the key backup) in order to encrypt the key for the key backup, it MUST explicitly inform the user at the time of password assignment that the password to be assigned MUST be different from the password for the user account at the Matrix home server, because otherwise the end-to-end encryption will no longer be effective, at least vis-à-vis the home server and those actors who control it.
In addition, the quality of the password is crucial for the protection of the key backup, which is why the TI-Messenger client SHOULD suggest a password to the user as part of the password assignment process that has a random combination of at least 14 characters in length, made up of numbers, special characters and upper and lower case letters. The source for coincidence that the TI-Messenger uses for this MUST be at least as good as those sources it uses in the negotiation of key material for communication.
For password-based key derivation (PBKDF2), >= 210,000 iterations are to be selected according to [OWASP PBKDF2] and the hash function is specified as SHA-512 by the Matrix specification.
The TI-Messenger client MAY provide the user with functions to facilitate secure storage of the password or key, for example, by offering export to an installed password manager.
If the TI-Messenger client suggests passphrases rather than passwords, the length MUST be increased accordingly, as combining existing words to form a phrase with a length of 14 characters does not span the same space of possible combinations as a random string of characters as required for the password. In the case of passphrases, strings are not to be chosen at random, instead the choice and sequence of words used MUST be random.**[<=]**

4.2 Access to the VZD-FHIR directory

Access to the VZD-FHIR directory FHIR proxy requires an access token issued by the Auth service. For this, the REST interfaces provided in the Auth service must be called up by the TI-Messenger client.

For write access to the FHIR directory, the TI-Messenger client MUST check if a valid owner-accesstoken exists locally. If no valid owner-accesstoken is available, the TI-Messenger client is forwarded to the central IDP service to negotiate an authorisation code. The TI-Messenger client receives a valid owner-accesstoken by presenting the authorisation code negotiated with the central IDP service at the `/signin-gematik-idp-dienst` interface. The Org Admin client is a special feature. If there is no valid owner-accesstoken in local storage, the Org Admin client MUST request a RegService OpenID token from the appropriate registration service, which will be exchanged for an owner-accesstoken at the `/owner-authenticate` endpoint.

For read access to the VZD-FHIR directory, the TI-Messenger client MUST check if a valid search-accesstoken is locally available. If no valid search-accesstoken is available, the TI-Messenger client MUST request this from the Auth service of the VZD-FHIR directory by calling `GET /tim-authenticate` while presenting a Matrix-Open ID-Token.

4.3 User guidance

By means of suitable user guidance, a high acceptance of the user is achieved. This includes simple and self-explanatory operation of the interface, which is based on common app design recommendations that can be found on the market. All relevant target groups MUST also be considered. The following interoperable functions MUST be provided by the manufacturer in order to achieve a minimum level of user acceptance. These are described below.

4.3.1 Presence display for other users

For a real-time user experience, TI-Messenger clients MUST, according to [Client-Server API#Presence], provide a presence display for other conversation partners. The presence display MUST be able to be switched on and off and MUST be deactivated by default in accordance with Privacy-by-default (Article 25(2) of the GDPR and downstream according to [A_22798]).

4.3.2 Mentioning users in the chat room

TI-Messenger clients MUST make it possible for other users to be mentioned in the respective chat room according to [Client-Server API#User, room and group mentions] via the input field. For this purpose, the TI-Messenger client MUST display a corresponding user list as soon as the user starts a new word with "`@`" or offer a corresponding "`@`" button in the chat room. TI-Messenger clients MUST display user mentions accordingly as a *pile* in the chat room. If it is a TI-Messenger client for mobile scenarios, the TI-Messenger client MUST display a corresponding push notification when the user has set the corresponding push rules.

4.3.3 Read confirmations

Read confirmations are used to provide information about when, if and by whom a message has been read within a chat room. For this reason, TI-Messenger clients MUST implement the Matrix specification according to [Client-Server API#Receipts]. TI-Messenger clients MUST implement the functions of viewing and sending read confirmations. The TI-Messenger client MUST support *Fully-Readmarkers*. Read confirmations MUST be able to be switched on and off and MUST be deactivated by default in accordance with Privacy-by-default (Article 25(2) of the GDPR and downstream according to [A_22798]).

4.3.4 Input notifications

TI-Messenger clients for mobile scenarios MUST implement the Matrix specification according to [Client-Server API#Typing Notifications]. TI-Messenger clients SHOULD indicate when the other party is writing a message in a chat room. The input notifications MUST be able to be switched on and off and MUST be deactivated by default in accordance with Privacy-by-default (Article 25(2) of the GDPR and downstream according to [A_22798].

4.3.5 Accessibility

ML-123582 – Accessibility standards
Manufacturers of a TI-Messenger client SHOULD follow the quality guidelines listed in [ISO 9241] for ensuring the ergonomics of interactive systems and requirements from the Ordinance on the Creation of Barrier-Free Information Technology in accordance with the Equal Opportunities for Persons with Disabilities Act (Barrier-Free Information Technology Ordinance – [BITV 2.0]).
[<=]

4.4 Configuration

The following section describes all functions to be configured that MUST be configured in the TI-Messenger client by the actor.

4.4.1 Setting push notifications

TI-Messenger clients MUST have a function to configure push notifications on an end device. In addition to push rules, device-side settings must also be made available to users according to [Client-Server API#Push Rules].

4.4.2 Ignore user

TI-Messenger clients MUST have a function to ignore messages from other users. Therefore, TI-Messenger clients MUST implement the Matrix specification according to [Client-Server API#Ignoring Users]. TI-Messenger clients MUST display a list of all ignored users and provide the option to undo the ignoring of users.

4.4.3 Room history

TI-Messenger clients MUST implement the Matrix specification according to [Client-Server API#Room History Visibility]. TI-Messenger clients MUST provide settings to determine the visibility of the room history. By default, the room history SHOULD be visible from the time of joining a chat room.

4.4.4 Visibility

TI-Messenger clients MUST have a function that can turn the visibility of an actor in the "User-HBA" role for the TI-Messenger service on or off in the person index of the VZD-FHIR directory. For this, the `status` attribute of the endpoint of a Practitioner resource MUST be set to the value `status == active` for switching on or `status == off` for switching off via the REST interface`/owner` at the FHIR proxy of the VZD FHIR directory. If the actor changes the `status` from `active` to `off`, the TI-Messenger client MUST check whether this MXID is also entered in the organisation directory via the REST interface `/search` on the FHIR proxy of the VZD-FHIR directory. If the MXID is also found in the organisation directory and if the stored `status` in this directory is `active`, then the TI-Messenger client MUST indicate to the actor that there is inconsistency in the stored visibility. The note MUST indicate that it is necessary to contact the administrator of their organisation in order to store the desired visibility in the organisation directory as well.

4.5 Test

Product testing to ensure compliance with the specification is entirely the responsibility of the suppliers/manufacturers of the TI-Messenger client. During approval, gematik focuses on the interaction of the products through E2E and IOP tests.

The independent product tests at the industrial partners include:

- Develop test environment,
- Create test case catalogue (for own product tests), and
- Perform and document product test.

The manufacturers of the TI-Messenger specialist services MUST assure that gematik can check the product tests of the industrial partners in the form of reviews of test concepts, test specifications, test cases and with the review of the test protocols (log and trace data).

Gematik promotes close cooperation and helps industrial partners to improve the quality of products. This is done by organising timely IOP tests, synchronising milestones and regular cross-industry test sessions. The test sessions include peer IOP and E2E testing.

gematik provides a TI-Messenger service reference implementation. To ensure interoperability between different TI-Messenger specialist services within the TI-Messenger service, the TI-Messenger specialist service of a TI-Messenger provider MUST be tested against the reference implementation (TI-Messenger client and TI-Messenger specialist service).

ML-124204 – Test of TI-Messenger client against reference implementation
The TI-Messenger client MUST be successfully tested against reference implementation. The test results are to be submitted to gematik.
[<=]

For vendor approval, TI-Messenger specialist services and TI-Messenger clients MUST be provided by the TI-Messenger provider. To enable an automated test for the TI-Messenger service, the test app of the TI-Messenger client MUST also provide a test driver module internally or externally. The following illustrations show the internal and external test driver module.

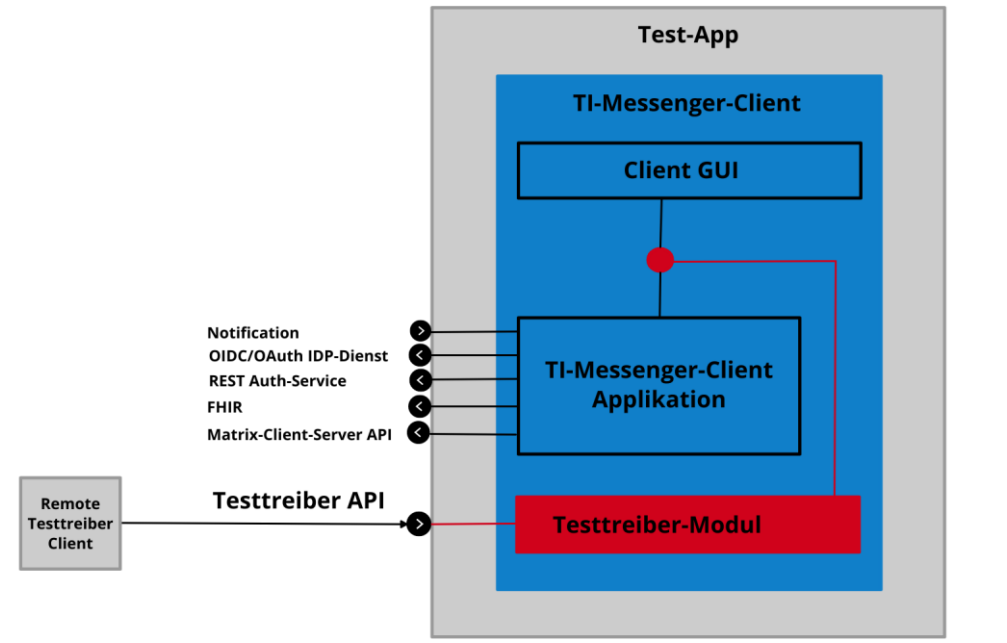

Figure 3: Internal test driver module

The external test driver module allows access to the test environment of the manufacturer and thus controls the test app.

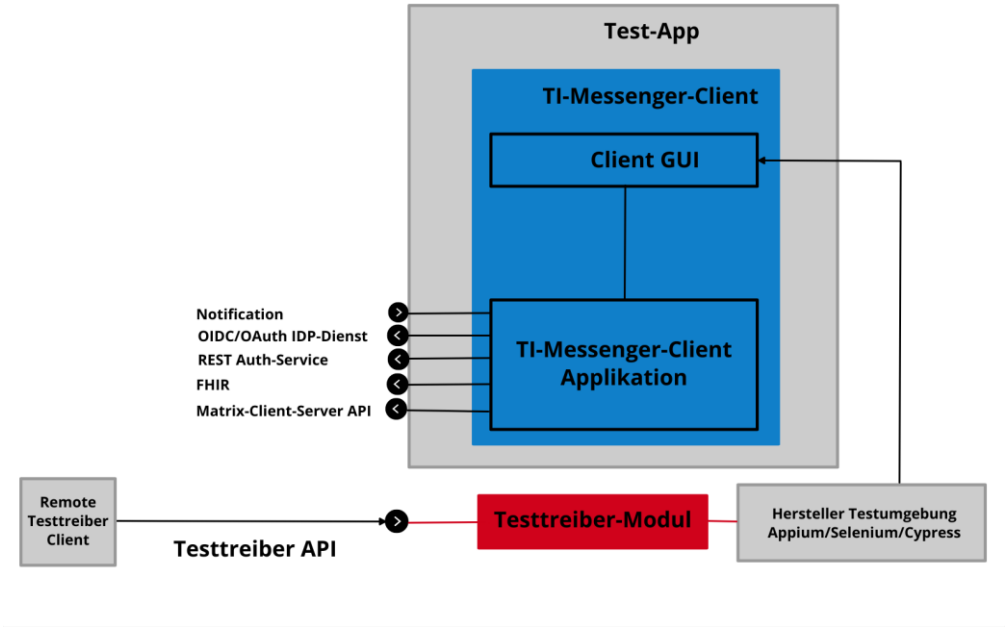

Figure 4: External test driver module

The test driver module MUST make the functionality of the product-specific interfaces of the TI-Messenger client accessible from the outside via a standardised interface and allow remote access. This test driver module MUST be part of the test APP (internal test

driver module) or ensure access to the manufacturer's test environment (external test driver module). The interface is specified and provided according to [Test driver API]. The test driver module MUST use the functionality offered by the TI-Messenger client via a product-specific interface to implement the operations of the TI-Messenger client. For an internal test driver module, the REST interface is integrated into the test app (access is directly via the end device). Testing of web clients (TI-Messenger client as web application) takes place exclusively via external driver modules. Testing requires organisations and messenger services. These organisations and messenger services MUST be set up by the manufacturers before the start of the test phase and the data (organisation names, etc.) MUST be transmitted to the system.

ML-124877 – Test app of the TI-Messenger client and test driver module
The test app of the TI-Messenger client MUST include a test driver module or ensure access to the manufacturer's test environment. The test driver module MUST use the functionality offered by the TI-Messenger client (the approval object) via a product-specific interface to implement the operations of the interfaces. The test driver module MAY process the output of the TI-Messenger client according to the technical interface, but MUST NOT corrupt the content.

Note: The interface according to [Test driver API] is specified and provided by the system.
[<=**]**

ML-124878 – Restriction of test driver module use
The productive TI-Messenger client MUST NOT contain a test driver module. The use of the test driver module is limited to the approval process in test apps and MUST NOT be used in operational apps.
[<=]

ML-124879 – No subject logic in test driver module
The test driver module MUST NOT implement the technical logic of the TI-Messenger client.
[<=]

Gematik tests on the basis of use cases as part of the approval process. Reference is made to the use cases from the [gemSpec_TI-Messenger service]. An attempt is made to include as many functional areas of the components of the TI-Messenger service as possible. The tests are first carried out against the reference implementation of gematik. In this step, the functionality of the "TI-Messenger service" approval object is checked. The IOP and E2E tests then demonstrate interoperability between the different TI messenger providers. For this purpose, all already available TI-Messenger services (the test instances of individual manufacturers) are then merged and tested against each other. All providers MUST perform this IOP and E2E testing independently and on their own responsibility in advance. In case of approval problems, suppliers MUST assist in the analysis. The following figure shows a system environment for manufacturer testing.

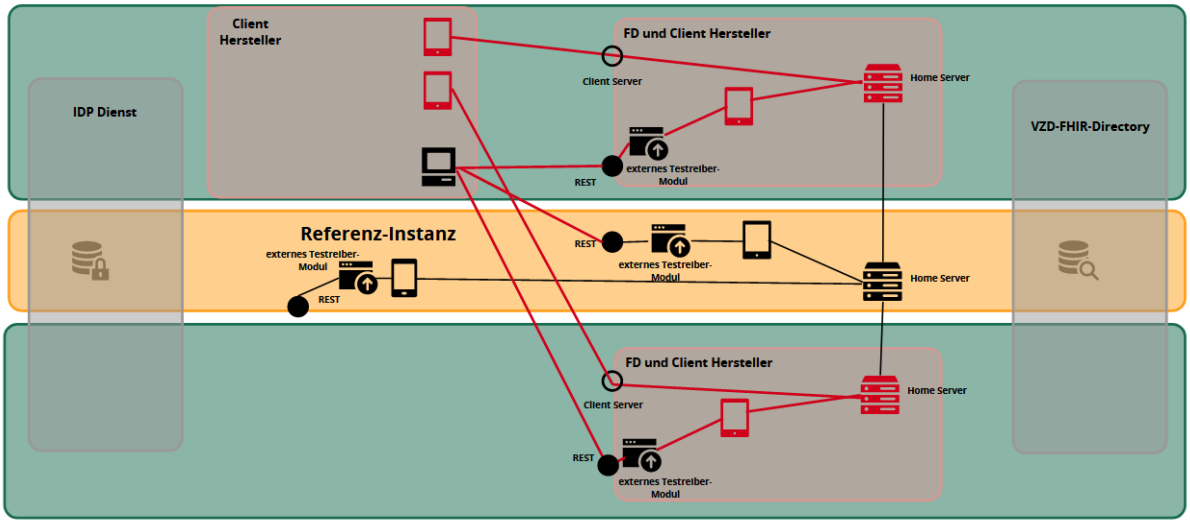

Figure 5: Test environment for manufacturer tests

In addition to the IOP and E2E tests already carried out, further interoperability tests of various TI-Messenger solutions are carried out before and after approval by gematik. The following figure shows the use of the existing test environment by gematik during the approval and interoperability tests.

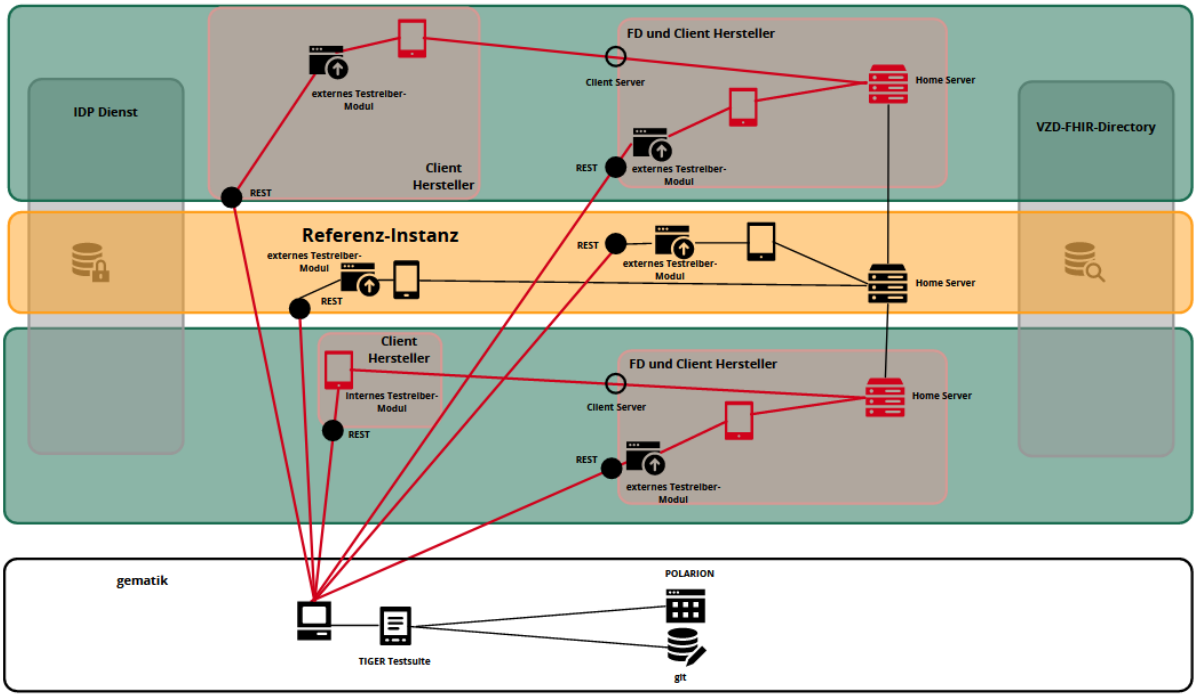

Figure 6: gematik test environment

4.6 Operational aspects

The operational readiness of the TI-Messenger provider client(s) in this section refers to systems on the server side that are necessary for the client to be operated safely by the user. Safe operation in terms of use on their TI-Messenger client end devices is ultimately the responsibility of the users or actors of TI-Messenger.

The TI-Messenger provider MUST support its users or actors in enabling the safe and functional operation of the TI-Messenger clients.

The TI-Messenger client MUST be fully functional with 98% availability.

The TI-Messenger provider MUST offer the TI-Messenger client product(s) to its users with a fully functional availability of 98%.

5 Functional features

The functionality of the TI-Messenger client results from the Matrix specification and MUST be supported by the respective TI-Messenger client. Functionalities that are not part of this specification but are included in the Matrix specification v1.3 MUST be implemented. Additional functionality included in a Matrix specification greater than v1.3 MAY ONLY be implemented if it has fallbacks.

5.1 Authentication procedures

TI-Messenger clients MUST at least support the following authentication procedures:

- **SSO login** according to [Client-Server API#SSO client login/authentication], and
- **OpenID-Connect** according to [Client-Server API#OpenID]

If an authentication procedure already used in the organisation is employed, the TI-Messenger client MUST support the entry of the required client credentials.

In addition, the manufacturer of a TI-Messenger client MUST ensure that the creation of guest accounts is prevented.

5.2 Matrix client server API

The core components of the TI-Messenger client are based on the Matrix client server API. In addition to the actual functionality for an ad-hoc message service, this also includes the management of sessions, notifications, etc., which is not discussed further in this specification. TI-Messenger clients MUST implement the Matrix client server API according to [Client-Server API] version v1.3. When implementing the Matrix client server API, the following should be observed:

5.2.1 Dealing with the createRoom event

The TI-Messenger client MUST allow a maximum of one additional person to be invited in the `createRoom` event according to the Matrix specification [Client-Server API#Creation].

5.2.2 Room upgrades

TI-Messenger clients MUST implement the Matrix specification according to [Client-Server API#Room Upgrades]. TI-Messenger clients MUST be able to handle room upgrades. The user SHOULD NOT notice that there is a new room version.

5.2.3 Send-to-device messaging

TI-Messenger clients MUST implement the Matrix specification according to [Client-Server API#Send-to-Device messaging].

5.2.4 Device management

TI-Messenger clients MUST support device management for a user's own devices. TI-Messenger clients MUST only implement the matrix specification for their own device management according to [Client-Server API#Device Management]. During implementation, device management MUST NOT be supported for the devices of other users in a chat room nor for the devices of all users of a messenger service.

5.2.5 Reporting content

TI-Messenger clients MUST implement the Matrix specification according to [Client-Server API#Reporting Content] and allow users to report unwanted content to users in the "Org Admin" role.

5.2.6 Instant messages

TI-Messenger clients MUST provide a function to exchange instant messages in a chat room according to [Client-Server API#Instant Messaging]. A TI-Messenger client MUST ensure that all incoming and outgoing events are displayed to the user in the correct chronological order. A TI-Messenger client MUST support retry logic for sending messages. TI-Messenger clients MUST notify users if an event has not been sent or has been sent incorrectly. TI-Messenger clients MUST display the display name of an actor; the associated MXID of an actor MAY be displayed. If a display name occurs more than once within a room because actors with identical display names are in it, the TI-Messenger client MUST display the associated MXID of the respective actors to enable unique identification. The detailed implementation procedure for this is described in [Client-Server API#Calculating the display name for a user].

The following `events` and `Msgtypes` MUST be supported by the TI-Messenger client:

Table 2: Events and Msgtypes

Events	Msgtypes
`m.room.message`	`m.text`
`m.room.name`	`m.emote`
`m.room.topic`	`m.notice`
`m.room.avatar`	`m.image`
	`m.file`
	`m.audio`
	`m.location`
	`m.video`

Messages in Matrix can be sent both in plain text and in HTML format. In the case that a TI-Messenger client does not support formatted messages a fallback for, e.g., replies as plain text, MUST be possible according to [Client-Server API#Fallbacks for rich replies].

The TI-Messenger client MUST support the following fallback events:

- Fallback for reply/quote, and
- Fallback for `m.text`, `m.notice`

Note: A fallback means that the TI-Messenger client sends an unformatted body in addition to the formatted body, which can be used by TI-Messenger clients without the respective formatting.

5.2.7 Direct messages

TI-Messenger clients MUST provide a function to exchange direct messages with other users of the TI-Messenger service according to [Client-Server API#Direct Messaging]. Direct messages means that a chat room is only created between two actors. This chat room cannot be extended to other players unless it is a technical system for archiving purposes. If a chat room is to be created for more than two actors, Group Messaging MUST be used.

The following options MUST be offered by the TI-Messenger client:

Table 3: Process – direct messages

Direct messaging between actors within an organisation	
User story: Searching for an actor via the user directory of the Matrix home server	1. Actor wants to start a new conversation 2. TI-Messenger client displays all actors of its organisation in the user directory of the Matrix home server 3. Actor selects an interlocutor and starts the chat The TI-Messenger client indicates that it is a direct chat. It is not possible to convert it into a group chat.
Direct messages between actors outside an organisation	
User story: Search for an actor via the person directory of the VZD-FHIR directory	1. Actor A in the role "User-HBA" wants to start a new conversation with actor B in the role "User-HBA" 2. Actor A searches the person directory of the VZD-FHIR directory for actor B 3. TI-Messenger client displays profile (e.g. name, organisation affiliation, occupation, etc.) of actor B 4. Actor A launches chat with actor B The TI-Messenger client indicates that it is a direct chat. It is not possible to convert it into a group chat.

Direct messaging between actors within an organisation	
User story: Exchange of contact data using QR-Scan	1. Actor A and actor B meet in person 2. Actor A and actor B each select "Start new conversation" in the TI-Messenger client 3. Actor A selects "Share QR code" 4. Actor B selects “Scan QR code” and scans “QR code” of actor A and receives the MXID from actor A 5. Actor A's MXID is entered in Actor B's release list 6. Actor A and actor B click "next" 7. A QR code is displayed to actor B, the QR code scanner is displayed to actor A 8. Actor A scans actor B's QR code and receives actor B's MXID 9. Actor B's MXID is entered in actor A's release list 10. At a later time, actor A or actor B MAY start a joint chat room. The TI-Messenger client indicates that it is a direct chat. It is not possible to convert it into a group chat.

5.2.8 Group chats

TI-Messenger clients MUST provide a function to start group chats and exchange messages within a chat group with users of the TI-Messenger service. TI-Messenger clients MUST be able to view all participants in a chat group. In addition, TI-Messenger clients MUST notify all participants in a group when another participant is added to the chat group. Participants may only be added to a chat group by invitation. Chat rooms to be managed with an organisation MUST always use Group Messaging.

The following options MUST be offered by the TI-Messenger client:

Table 4: Process – group chats

Group chats between actors within an organisation	
User story: Searching for an actor via the user directory of the Matrix home server	1. Actor wants to start a new group chat. 2. TI-Messenger client displays all actors of its organisation in the user directory of the Matrix home server 3. Actor selects interlocutors. 4. Participants are invited to the group chat. 5. Actor can add other interlocutors.

Group chats between actors within an organisation	
Group chat between actors outside an organisation	
User story: Search for an actor via the organisation directory of the VZD-FHIR directory	1. Actor wants to send a message to another organisation and start a group chat 2. Actor searches the organisation directory of the VZD-FHIR directory for the organisation 3. The TI-Messenger client displays the profile of the organisation (e.g. name, type, contact options, etc.) 4. Actor selects the MXID of an actor of the organisation and starts a chat with this actor
User story: Search for an actor via the organisation directory of the VZD-FHIR directory in order to invite other actors to the group chat	1. Actor wants to invite other organisations to the existing chat group 2. Actor searches the organisation directory of the VZD-FHIR directory for the organisation 3. TI-Messenger client displays profile of organisation (e.g. name, type, contact options) 4. Actor invites the actor of the organisation to existing group chat
User story: Search for an actor via the user directory of the Matrix home server or the person directory of the VZD-FHIR directory	1. Actor wants to invite other actors to the existing chat group 2. Actor searches either the user directory of their organisation or the person directory of the VZD-FHIR directory for the invitation of an actor outside their organisation 3. Actor selects a found actor 4. Actor is invited to existing chat group

5.2.9 Push notifications

TI-Messenger clients for mobile scenarios MUST implement the Matrix specification according to [Client-Server API#Push Notifications]. The following illustration shows the flow of push notifications sent to a mobile phone where push notifications are delivered via the mobile phone provider.

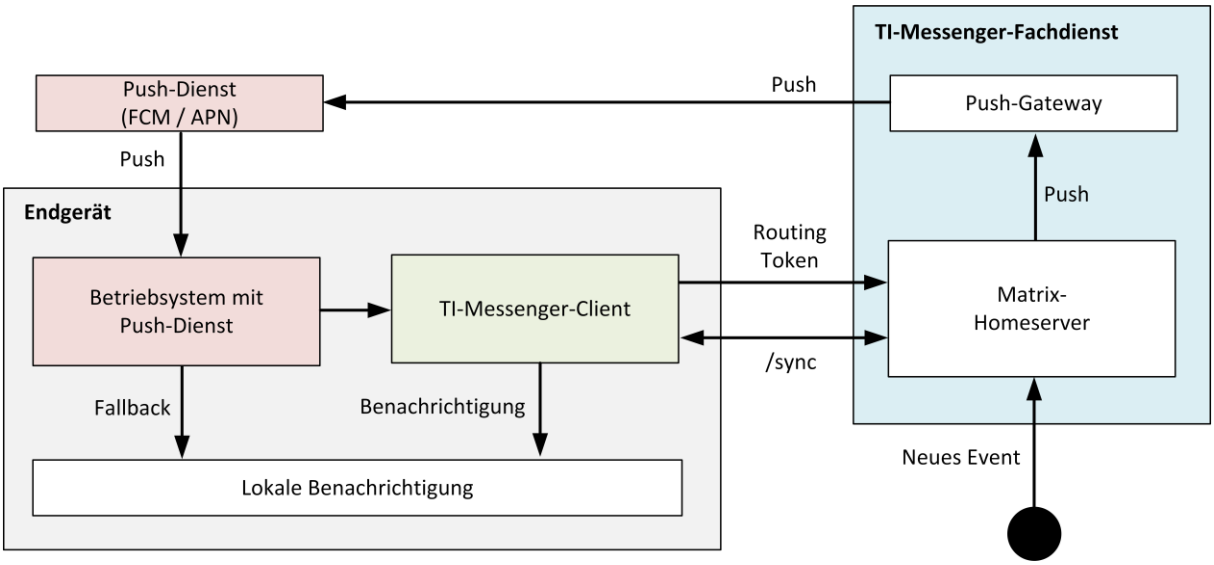

Figure 7: Push notification for end devices

Note: In the illustration, the messenger proxy was not shown for clarity reasons.

Flow:

1. The TI-Messenger client logs in to a Matrix home server.
2. The TI-Messenger client logs in to the push provider and receives a routing token.
3. The TI-Messenger client uses the Matrix Client / Server API to add a pusher by specifying the URL of the push gateway configured for the TI-Messenger client and passes the routing token.
4. The Matrix home server routes push notifications to the push gateway specified on the URL. The push gateway forwards this notification to the push provider, passing the routing token along with any necessary private credentials that the provider needs to send push notifications.
5. The push provider sends the notification to the end device.
6. The end device's operating system passes the notification to the TI-Messenger client.
7. The TI-Messenger client decrypts the notification.
8. The TI-Messenger client synchronises with the Matrix home server and displays the notification locally.

5.2.9.1 Push providers

A push provider is a service managed by the device manufacturer that can send notifications directly to the end device. A mobile TI-Messenger client MUST support the respective push provider of the system.

5.2.9.2 Push gateway

A push gateway is provided by the TI-Messenger provider and is a server that receives event notifications from Matrix home servers and forwards them to other services. The TI-Messenger clients will organisationally receive a routing token from the TI-Messenger

provider and notify the Matrix home server to which push gateway the notifications will be sent. A TI-Messenger client for mobile scenarios MUST be organisationally linked to the push gateway of the TI-Messenger provider. The TI-Messenger client MUST ensure that the routing token is securely stored on the end device and cannot be misused.

5.2.9.3 Push rule

A push rule is a single rule that determines the conditions under which an event is forwarded to a push gateway and how the notification is presented. These rules are stored on the user's Matrix home server. The TI-Messenger client MUST allow users to create and display push rules for each room.

5.2.9.4 Push rule set

A push rule set covers a set of rules according to certain criteria. For example, some rules can only be applied to messages from a specific sender, a specific room or by default. The push rule set contains the entire set of scopes and rules. The TI-Messenger client for mobile scenarios MUST offer users possibilities to manage push rule sets.

5.2.9.5 Opt-in

The manufacturer of a TI-Messenger client MUST provide an opt-in procedure for push notifications by users. The opt-in procedure MUST be provided for each end device.

5.3 Administration functions

The TI-Messenger client with administration functions is a client for actors of an organisation in the "Org Admin" role. This is also referred to as the Org Admin client in the context of the TI-Messenger service. The Org Admin client is used for convenient management of messenger services at a TI-Messenger specialist service. The Org Admin client MAY be provided as a standalone client or as an integration into a TI-Messenger client for actors. If regular user functions and administration functions are offered in the same client, a clear distinction between user and administration functions MUST be made. TI-Messenger clients with administration functions MUST implement the Matrix specification in accordance with [Client-Server API#Server Administration]. In the following, the administration functions to be provided by the Org Admin client are described in more detail.

The Org Admin client MUST allow the administration of actors and devices on the messenger services assigned to its organisation. The Org Admin client MUST also display active and inactive sessions of the devices from which a user has logged in. The Org Admin client MUST provide the Org Admin with the ability to invalidate the access tokens of the individual devices or all devices in order to log off the user. In addition, the Org Admin client MUST allow sending information / system messages to the TI-Messenger clients logged in to a messenger service.

With the Org Admin client, it is possible to manage FHIR resources on behalf of the organisation in the VZD-FHIR directory. To this end, the Org Admin client MUST be able to administer the FHIR resource *HealthcareService* via the `/owner` interface in the VZD-FHIR directory. The Org Admin client must also be able to read entries in the VZD-FHIR directory using the `/search` interface. To administer records on the VZD-FHIR directory, the Org Admin client MUST first show the relevant entries to the actor in the "Org Admin" role before it changes the data by calling the `/owner` interface in the VZD-FHIR directory.

Using the Org Admin client, it must be possible to enter function accounts into the VZD-FHIR directory as the end point of a *HealthcareService* resource of an organisation. When the endpoint is configured by the actor in the "Org Admin" role, the display name MUST contain the `Chatbot` marker if the function account is implemented via a chatbot. The following formation rule must be used for the display name: [Name of the function account] (Chatbot).

Summary

- User management (list of all actors, creation, editing, deletion)
- Device management (display, unsubscribe, delete all devices of a messenger service of its organisation)
- Manage entries in the VZD-FHIR directory
- Send system messages to operators of a messenger service (e.g. make maintenance window known)
- Set up function accounts

5.4 Other functions

The following section describes other functionalities that the TI-Messenger client MUST implement.

5.4.1 Logging in to a messenger service

The TI-Messenger client MAY display a list of all messenger services supported by the TI-Messenger provider to the actor during the login process. If this is not supported by the provider, the actor MUST be offered an opportunity to configure the desired messenger service.

Note: The provision of parameters to be used by the actor (e.g. Matrix domain of the messenger service) is left to the respective provider.

5.4.2 Authentication mask

The TI-Messenger client MUST display an authentication mask with authentication methods supported by the messenger service to the player during the login process.

5.4.3 Creation of the local part

The TI-Messenger client MAY ensure that no personal data is identifiable when creating the local part of an actor's MXID. For this purpose, the TI-Messenger client MAY compute the local part of the actor's used MXID as Base32 SHA256 Hash. If this variant for creating the local part of the MXID is not desired, this can deactivate an actor.

5.4.4 Display name

The TI-Messenger client MUST apply the following formation rule when initially assigning the display name: `[Surname], [First name]`. The TI-Messenger client MUST also

ensure that an actor cannot subsequently change their own display name.

ML-132303 – Editability of display names
It is not possible for an actor to edit their own display name in the "User/User-HBA" role.
[<=]

5.4.5 Identification features

To ensure that only approved TI-Messenger clients are used, a user agent identifier MUST be implemented by the TI-Messenger client manufacturer in the TI-Messenger client. The components related to this approval MUST be provided by the TI-Messenger client manufacturer to the TI-Messenger provider after each change so that they can be used in testing the messenger proxy of a messenger service. The user agent identifier MUST be transferred on each call in the HTTP header.

A_23104-01 – TI-M client user agent
The TI-Messenger client for actors and the TI-Messenger client with administration functions (Org Admin client) MUST transmit the following user agent identifier for each connection establishment to the TI-Messenger specialist service:

```
User agent: $Product type version,$Product
version,$Characteristic,$Platform,$OS,$OS version,$client_id
```

For a description of the respective data fields, see [gemSpec_Perf#A_22940-x].
[<=]

5.4.6 Overview of devices used

The TI-Messenger client MUST be able to display an overview of registered devices to the actor. The display MUST include a subdivision into verified and unverified devices. For each displayed device, the last activity status MUST be displayed and the actor MUST unsubscribe from individual devices and thus be able to invalidate their Matrix ACCESS_TOKEN.

5.4.7 Connection only to messenger services available in the Federation

The TI-Messenger client MUST ensure that use is only possible with Matrix home servers that are part of the Federation. If the TI-Messenger client connects to a Matrix home server that is not part of the Federation, the actor MUST be logged out directly.

5.4.8 Third party networks / bridging

Bridging to other messaging protocols MUST NOT take place. Only the Matrix Client Server and the Matrix Server server API MUST be used as the messaging protocol. One-client bidirectional exchange with third-party systems MAY be possible, for example, to allow the archiving of chat messages or chatbots. To this end, the TI-Messenger client can be integrated as a module into an existing system.

5.4.9 Dealing with the createRoom event

The TI-Messenger client MUST inform the user in the event of an error that the communication could not be started. To do this, the TI-Messenger client MUST inform the user about the error in an understandable way.

5.4.10 User directory of a messenger service

The TI-Messenger client MUST provide a function that enables actors to access and search a directory of other actors within their organisation on the respective Matrix home server of a messenger service.

5.4.11 VZD-FHIR directory search queries

The TI-Messenger client MUST provide a function that enables actors to search the VZD-FHIR directory for resources. The TI-Messenger client MUST provide a function to view detailed information about the resources stored on the VZD FIR directory. Further specifications can be found in [gemSpec_VZD_FHIR_Directory].

5.4.12 Creating and displaying 2D barcodes

The TI-Messenger client for mobile scenarios MUST provide a function to create 2D barcodes and display them on the end device display. Here, the 2D code MUST be encoded into a QR code representation according to ISO/IEC 18004:2006. At least the fields of the following vCard object MUST be used as content for generating the 2D code:

```
BEGIN:VCARD
VERSION:4.0

N:<surname>;<first name>;<additional
first names>;<title>;<name affixes>
FN:<first name><surname>
IMPP:matrix://<MXID>
END:VCARD
```

Note: The structure of the Matrix URI MUST be formed according to [Matrix-Appendices#uris].

5.4.13 Scanning and processing the 2D barcode

The TI-Messenger client for mobile scenarios MUST provide a function that allows the actor to scan a 2D barcode (in a QR code representation) via the camera of the end device. The TI-Messenger MUST decode the scanned 2D code in accordance with ISO/IEC 18004:2006 and display at least the full name and Matrix user ID from the `N` and `IMPP` parameters to the actor so that the latter can confirm or reject inclusion in the release list. In addition, the TI-Messenger client MUST transfer the MXID to its local TI-Messenger contact list.

5.4.14 Administration of the release list

The TI-Messenger client MUST provide a function that allows an actor to release invitations to a chat room for other actors. For this purpose, the TI-Messenger client MUST call the operations of the RESTful Webservice `/tim-contact-mgmt/v1.0` according to [api-messenger#TiMessengerContactManagement.yaml] in version 1.0 on the messenger proxy of its messenger service. The TI-Messenger client MUST be able to display to the actor a list in which all actors who have received a release are shown. Likewise, the TI-Messenger client MUST be able to create and process releases.

Note: The release list is required if the actors were contacted in person using a QR code scan, for example.

5.4.15 Archiving of conversation content

In order to comply with the documentation requirements of physicians, it is necessary that case-related chat processes can also be stored beyond deletion of the conversation data. Therefore, the TI-Messenger client MUST ensure that chats can be extracted from the TI-Messenger client so that they can be transferred to archive systems, for example. gematik does not set any specifications such as archiving, as both the type of archiving and the systems to be connected vary greatly.

5.4.16 Case-related communication

Case-related communication allows users to exchange structured data on a medical case and further process it in their primary system. For this, the TI-Messenger client MUST also initialise the room type during room creation and fill it with the designated *custom state events* at initialisation time. To do this, the TI-Messenger client MUST create and use the *custom room type* `de.gematik.tim.roomtype.casereference.v1` for case-related communication using a parametrised call of the `/createRoom` endpoint (`m.room.create` *state event* under call of the `/createRoom` endpoint).

Type of room generation: calling the `/createRoom` endpoint

Mandatory parameters of this call as a sorted hierarchical list:

- `m.room.create` (State Event)
  - `creator: <user_id of the creator>`
  - `type: de.gematik.tim.roomtype.casereference.v1` (Custom Room Type)
- `initial_state` (state event list)
  - `de.gematik.tim.room.casereference.v1` (Custom State Event)
    - `Event type: de.gematik.tim.room.casereference.v1`
      `Event room_id: <room_id of the existing chat room>`
      `Event state_key: <defined by the sender>`
      `Event content:<defined in [simplifier]>`
  - `de.gematik.tim.room.name` (Custom State Event)
    - `<See "Other TI-Messenger-specific custom state events" section for description>`
  - `de.gematik.tim.room.topic` (Custom State Event)

    - <See "Other TI-Messenger-specific custom state events" section for description>
- m.room.name (State Event)
    - <empty> (0-length string)
- m.room.topic (State Event)
    - <empty> (0-length string)

In the *custom state event* de.gematik.tim.roomtype.casereference.v1, the de.gematik.tim.roomtype substring corresponds to the namespace assigned by gematik. The casereference and v1 substrings correspond to the unique *customroom type of* the room to be initialised or its room type version number *(note: does not mean room version)*.

In the *custom state event* de.gematik.tim.room.casereference.v1, the de.gematik.tim.room substring in the *event type* corresponds to the namespace assigned by gematik. The casereference and v1 substrings correspond to the unique ID of the *event type* or its version number. The FHIR resources are entered in the *event content* as JSON data and summarised as an FHIR bundle. The profiles of FHIR resources are located in the Simplifier project [simplifier]. It should be noted that the canonical URLs of the resources always contain http://gematik.de/fhir/TIM/CaseReference.

The TI Messenger client MUST be able to generate a *custom state event* with the *event type* de.gematik.tim.room.casereference.v1. In doing so, the TI-Messenger client MUST ensure that the *state events* m.room.topic and m.room.name are empty (0-length string). Instead, the *custom state events* specific to gematik with the *event types* de.gematik.tim.room.topic and de.gematik.tim.room.name MUST be used. The TI-Messenger client MUST also allow room-related endpoint calls, e.g., /_matrix/client/v3/rooms/{roomId}/invite to continue after using events of this type and rooms with this *custom room type*. Apart from these specifications, the call sequence for the /createRoom endpoint call applies according to [Client-Server API].

Example of mandatory parameters in the *state event* m.room.create according to [Client-Server API] for the /createRoom endpoint call:

```
{
    "content": {
      "creator": "@example:example.org",
      "type": "de.gematik.tim.roomtype.casereference.v1",
      "room_version": "10"
    },
    "event_id": "$143273582443PhrSn:example.org",
    "origin_server_ts": 1432735824653,
    "room_id": "!jEsUZKDJdhlrceRyVU:example.org",
    "sender": "@example:example.org",
    "state_key": "",
    "type": "m.room.create",
    "unsigned": {
      "age": 1234
    }
}
```

Example of further mandatory parameters according to [Client-Server API] for the `/createRoom` endpoint call:

```
{
    "initial_state": [
        {
            "content": {<defined in [simplifier]> },
            "event_id": "$143273582443PhrSn:example.org",
            "origin_server_ts": 1432735824653,
            "sender": "@example:example.org",
            "state_key": "<defined by the sender>",
            "type": "de.gematik.tim.room.casereference.v1",
            "unsigned": {
                "age": 1234
            }
        },
        {
            "content": {
                "name": "A TI-Messenger-specific room name"
            },
            "event_id": "$143273582443PhrSn:example.org",
            "origin_server_ts": 1432735824653,
            "sender": "@example:example.org",
            "state_key": "",
            "type": "de.gematik.tim.room.name",
            "unsigned": {
                "age": 1234
            }
        },
        {
            "content": {
                "topic": "A TI-Messenger-specific room topic"
            },
            "event_id": "$143273582443PhrSn:example.org",
            "origin_server_ts": 1432735824653,
            "sender": "@example:example.org",
            "state_key": "",
            "type": "de.gematik.tim.room.topic",
            "unsigned": {
                "age": 1234
            }
        }
    ],
    "invite": [
        null
    ],
    "name": "",
    "topic": ""
}
```

Note: The prerequisite for productive use is the implementation of MSC3414 Encrypted state events (https://github.com/matrix-org/matrix-spec-proposals/pull/3414). Necessary adaptations to rulesets (client-side, server-side) will be defined with the advent of the productive usability of these functionalities, but at the earliest in the next version of this specification.

5.4.17 Federated and intersectoral communication

The federated and intersectoral communication allows actors within the TI-Messenger service to communicate with other actors in an inter-organisational and federated way. For this, the TI-Messenger client MUST also initialise the room type during room creation and fill it with the designated *custom state events* at initialisation time. To do this, the TI-Messenger client MUST create and use the *custom room type* `de.gematik.tim.roomtype.default.v1` for federated and intersectoral communication using a parametrised call of the `/createRoom` endpoint (`m.room.create` *state event* under call of the `/createRoom` endpoint). This is the standard *custom room type*. Any chat room to which participants from other organisations are invited and thus use the federation MUST use this *custom room type* and the associated mandatory parameter list, unless another *custom room type* is explicitly selected by the user.

Type of room generation: calling the`/createRoom` endpoint

Mandatory parameters of this call as a sorted hierarchical list:

- `m.room.create` (State Event)
  - `creator: <user_id of the creator>`
  - `type: de.gematik.tim.roomtype.default.v1` (Custom Room Type)
- `initial_state` (state event list)
  - `de.gematik.tim.room.default.v1` (Custom State Event)
    - `Event type: de.gematik.tim.room.default.v1`
      `Event room_id: <room_id of the existing chat room>`
      `Event state_key: <defined by the sender>`
      `Event content:<defined in [simplifier]>`
  - `de.gematik.tim.room.name` (Custom State Event)
    - `<See "Other TI-Messenger-specific custom state events" section for description>`
  - `de.gematik.tim.room.topic` (Custom State Event)
    - `<See "Other TI-Messenger-specific custom state events" section for description>`
- `m.room.name` (State Event)
  - `<empty> (0-length string)`
- `m.room.topic` (State Event)
  - `<empty> (0-length string)`

In the *custom event* `de.gematik.tim.roomtype.default.v1`, the `de.gematik.tim.roomtype` substring corresponds to the namespace assigned by gematik. The `default` and `v1` substrings correspond to the unique *custom room type of* the room to be initialised or its room type version number *(note: does not mean room version)*.

In the *custom state event* `de.gematik.tim.room.default.v1`, the `de.gematik.tim.room` substring in the *event type* corresponds to the namespace assigned by gematik. The `default` and `v1` substrings correspond to the unique ID of the

event type or its version number. The FHIR resources are entered in the *event content* as JSON data and summarised as an FHIR bundle. The profiles of FHIR resources are located in the Simplifier project [simplifier].

The TI Messenger client MUST be able to generate a *custom state event* with the *event type* `de.gematik.tim.room.default.v1`. In doing so, the TI-Messenger client MUST ensure that the *state events* `m.room.topic` and `m.room.name` are empty (0-length string) as direct successor events. Instead, the *custom state events* specific to gematik with the *event types* `de.gematik.tim.room.topic` and `de.gematik.tim.room.name` MUST be used. The TI-Messenger client MUST also allow room-related endpoint calls, e.g., `/_matrix/client/v3/rooms/{roomId}/invite` to continue after using events of this type and rooms with this *custom room type*.

Example of mandatory parameters in the state event `m.room.create` according to [Client-Server API] for the `/createRoom` endpoint call:

```
{
    "content": {
      "creator": "@example:example.org",
      "type": "de.gematik.tim.roomtype.default.v1",
      "room_version": "10"
    },
    "event_id": "$143273582443PhrSn:example.org",
    "origin_server_ts": 1432735824653,
    "room_id": "!jEsUZKDJdhlrceRyVU:example.org",
    "sender": "@example:example.org",
    "state_key": "",
    "type": "m.room.create",
    "unsigned": {
      "age": 1234
    }
}
```

Example of further mandatory parameters according to [Client-Server API] for the `/createRoom` endpoint call:

```
{
    "initial_state": [
        {
            "content": {<defined in [simplifier]> },
            "event_id": "$143273582443PhrSn:example.org",
            "origin_server_ts": 1432735824653,
            "sender": "@example:example.org",
            "state_key": "<defined by the sender>",
            "type": "de.gematik.tim.room.default.v1",
            "unsigned": {
                "age": 1234
            }
        },
        {
            "content": {
                "name": "A TI-Messenger specific room name"
            },
            "event_id": "$143273582443PhrSn:example.org",
            "origin_server_ts": 1432735824653,
            "sender": "@example:example.org",
            "state_key": "",
            "type": "de.gematik.tim.room.name",
            "unsigned": {
                "age": 1234
            }
        },
        {
            "content": {
                "topic": "A TI-Messenger specific room topic"
            },
            "event_id": "$143273582443PhrSn:example.org",
            "origin_server_ts": 1432735824653,
            "sender": "@example:example.org",
            "state_key": "",
            "type": "de.gematik.tim.room.topic",
            "unsigned": {
                "age": 1234
            }
        }
    ],
    "invite": [
        null
    ],
    "name": "",
    "topic": ""
}
```

5.4.18 Other TI-Messenger-specific custom state events

In order to be able to describe the case-related as well as the federated and intersectoral communication in a targeted manner, the following *custom state events* are introduced, which have the properties of the *state events* `m.room.name` and `m.room.topic`:

Custom state event:
`Event type: en.gematik.tim.room.name`
`Event room_id: <room_id of the existing chat room>`

```
Event state_key: <empty> (0-length string)
Event content: <name: defined room name>
```

Custom state event:

```
Event type: en.gematik.tim.room.topic
Event room_id: <room_id of the existing chat room>
Event state_key: <empty> (0-length string)
Event content: <topic: defined room topic>
```

The `de.gematik.tim.room` substring in the *event type* corresponds to the namespace assigned by gematik. The `name` and `topic` substrings correspond to the unique ID of the respective specific *event type*. The TI-Messenger client MUST be able to generate the *custom state events* with the event type `de.gematik.tim.room.name` and `de.gematik.tim.room.topic`. In doing so, the TI-Messenger client MUST ensure that

- these two *custom state events* are used to name the room name or topic, provided that *custom state events* of case-related or internal/intersectoral communication were also used in the same room,
- the *state* events `m.room.topic` and `m.room.name` are empty (0-length string) as immediate successor events,
- the event definitions (including *event content*) and the event format of these two *custom state events*, apart from the unique ID *(event type*), match those of the *state events* `m.room.topic` and `m.room.name` according to [Client-Server API].

The TI-Messenger client MUST also allow room-related API calls, e.g., `/_matrix/client/v3/rooms/{roomId}/invite` to continue after using events of this type.

Example according to [Client-Server API]:

and

```
{
 "content": { "topic": "A TI-Messenger-specific room topic" },
 "event_id": "$143273582443PhrSn:example.org",
 "origin_server_ts": 1432735824653,
 "room_id": "<room_id of the existing chat room>",
 "sender": "@example:example.org",
 "state_key": "",
 "type": "de.gematik.tim.room.topic",
 "unsigned": {
   "age": 1234
 }
}
```

6 Annex A – Directories

6.1 Abbreviations

Abbreviation	Explanation
APN	Apple Push Notification Service
CC	Common Criteria
FCM	Firebase Cloud Messaging
FHIR	Fast Healthcare Interoperable Resources
IDP	Identity Provider
JSON	JavaScript Object Notation
MXID	Matrix ID
OLM/MEGOLM	Message content encryption protocol specified by the Matrix Foundation
OWASP	Open Web Application Security Project
PVS	Practice Management System
SMC-B	Institution card (Security Module Card Type B)
SSO	Single Sign-On
SSSS	Secure Secret Storage and Sharing
TI	Telematics infrastructure
TLS	Transport Layer Security
VZD	Directory service

6.2 Glossary

Term	Explanation
MXID	Unique identification of a TI-Messenger user

6.3 List of figures

6.4 List of tables

6.5 Referenced documents

6.5.1 gematik documents

The following table contains the names of the gematik documents on telematics infrastructure referenced in this document. The version-related state of development of these concepts and specifications is defined per release in a document map; the version and status of the referenced documents are therefore not listed in the table below. Their respective valid version numbers for this document are included in the current document map published by gematik, in which the present version is listed.

[Source]	Published by: Title
[api-messenger]	gematik: api-ti-messenger https://github.com/gematik/api-ti-messenger/
[gemGlossary]	gematik: Introduction of health card – glossary
[gemKPT_Betr]	gematik: Operating concept online productive operation
[gemSpec_Krypt]	gematik: General specification – use of cryptographic algorithms in the telematics infrastructure
[gemSpec_TI-Messenger-Service]	gematik: Specification TI-Messenger service
[gemSpec_TI-Messenger-FD]	gematik: Specification TI-Messenger specialist service
[gemSpec_VZD_FHIR_Directory]	gematik: FHIR Directory specification directory service
[simplifier]	gematik: TI-Messenger https://simplifier.net/tim

6.5.2 Other documents

[Source]	Publisher (publication date): Title
[BITV 2.0]	Verordnung zur Schaffung barrierefreier Informationstechnik nach dem Behindertengleichstellungsgesetz (Barrierefreie-Informationstechnik-Verordnung - BITV 2.0) https://www.gesetze-im-internet.de/bitv_2_0/BJNR184300011.html
[BSI-TR-03166]	BSI TR-03166 - Technical Guideline for Biometric Authentication Components in Devices for Authentication https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/TechGuidelines/TR03166/BSI-TR-03166.pdf
[BSI 2-Faktor]	BSI Bewertungstabellen IT-Sicherheit https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Studien/2FA/it-sicherheit.pdf?__blob=publicationFile&v=3

[Source]	Publisher (publication date): Title
[BSI Frontend]	BSI Prüfvorschrift für den Produktgutachter https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/DigitaleGesellschaft/Pruefvorschrift_Produktgutachter_ePA-Frontend.pdf?__blob=publicationFile&v=3
[Client-Server API]	Matrix Foundation: Matrix Specification - Client-Server API https://spec.matrix.org/v1.3/client-server-api/
[DSK2021]	Datenschutzkonferenz (DSK): Stellungnahme der Konferenz der unabhängigen Datenschutzaufsichtsbehörden des Bundes und der Länder vom 29. April 2021 https://www.datenschutzkonferenz-online.de/media/st/20210429_DSK_Stellungnahme_Messengerdienste_Krankenhausbereich.pdf
[ISO 9241]	Ergonomics of human-system interaction https://www.iso.org
[Matrix-SSSS]	Matrix Foundation: Secure Server Storage and Sharing https://matrix.org/docs/guides/implementing-more-advanced-e-2-ee-features-such-as-cross-signing
[Matrix-Appendices]	Matrix Foundation: Matrix Specification - Appendices https://spec.matrix.org/v1.3/appendices/
[OWASP MobileTop10]	OWASP Mobile Top 10 https://owasp.org/www-project-mobile-top-10/
[OWASP Proactive Control]	OWASP Proactive Controls https://owasp.org/www-project-proactive-controls/
[OWASP PBKDF2]	OWASP Password Storage Cheat Sheet https://cheatsheetseries.owasp.org/cheatsheets/Password_Storage_Cheat_Sheet.html#pbkdf2
[Testtreiber API]	Testtreiber API https://github.com/gematik/api-ti-messenger/tree/master/src/openapi