

Schnittstellen-Dokumentation

BefundApp

Projektname: **Befund**
Projektart: **Backendservices (Inside & Outside)**
Projektziel: **Anbindung der App-Services mit dem LIS & App**
Zielgruppe: **Administratoren**
Status: **In Arbeit**

Projektstatus und Versionsstand

Status:

- in Arbeit
- intern
- in Prüfung
- freigegeben
- Archiv

Version	Datum	Bearbeitung	Bearbeiter	Kommentar
1.0.3	15.07.2022	Ersterstellung	I. Abetschkhrischwili	Die verwendeten Anmeldeinformationen in der Dokumentation dienen nur zur Demo und dürfen nicht in der Produktion direkt übernommen werden

Dokumentenablage

dev.azure.com/labor-stein	SourceControl (Azure Github)
---------------------------	------------------------------

Inhalt

Ziel und Inhalt des Projektes	3
Glossar	3
Systemvoraussetzungen	3
Wichtige Hinweise für die Einrichtung der Backendservices	4
Konfiguration des Service-Inside	4
appsettings.json (Service-Inside)	4
Log-Einstellungen zum Protokollieren der verschiedenen Ereignisse im Service	5
LOG-Typen:	5
CleanUpDays.....	5
PrivateDirectory.....	5
PublicDirectory	6
MaxWorkerThreads.....	6
ServiceOutsideIP.....	6
ServiceOutsideURL	6
ConnectionStrings.....	6
Netzwerkinfrastruktur	6
Konfiguration des Service-Outside	7
appsettings.json (Service-Outside)	7
Log-Einstellungen zum Protokollieren der verschiedenen Ereignisse im Service	8
LOG-Typen:	8
IpRateLimiting.....	8
ServiceInsideIP.....	8
ServiceInsideURL	9
ConnectionStrings.....	9
PushNotifications.....	9
Freigabe vom ServiceInside (Datenverzeichnis)	10
HL7 Dateien	10
.ack Log Dateien.....	11
MySQL Datentabelle	14
Beispiel von MOLIS Anbindung.....	16

Ziel und Inhalt des Projektes

Dieses Dokument dient der Beschreibung der Einrichtung für die App-Services (Inside & Outside). Des Weiteren werden in der Dokumentation die Voraussetzungen zur Anbindung des In-Services mit einem LIS beschrieben. Die App-Services ermöglichen der mobilen Applikation die Anforderungen an LIS auf einem sicheren Weg zu übertragen bzw. die Rückantworten des LIS an die App zurückzuleiten. Sowohl der Service-Inside als auch der Service-Outside sollen in einer DMZ-Zone aufgebaut werden.

Glossar

Bezeichnung	Beschreibung
App/Applikation	Eine Anwendung, die auf einem mobilen Gerät ausgeführt wird.
Service	Eine Anwendung, die auf einem Server dauerhaft ausgeführt wird. Diese Anwendung stellt die Daten und Business-Logik den berechtigten Kommunikationspartnern zur Verfügung.
LIS	Labor Informationssystem
DMZ-Zone	Eine demilitarisierte Zone, die eine neutrale Zone zwischen dem privaten Netzwerk eines Unternehmens und einem externen öffentlichen Netzwerk bildet

Systemvoraussetzungen

Die beiden Services (Inside & Outside) benötigen ein Windows-Betriebssystem mit .NET Core > 6.0 Unterstützung. Zum Speichern von Daten wird eine MySQL Datenbank benötigt. Zu dieser Datenbank, sollen die beiden Services Zugang erhalten. Die Connection Strings zur MySQL-Datenbank können in den Konfigurationsdateien (S. appsettings.json) von jeweiligem Service verschlüsselt hinterlegt werden. Darüber hinaus müssen die beiden Services mit den gültigen SSL-Zertifikaten betrieben werden. **(Ein self signed certificate ist an der Stelle nicht geeignet)**. Da der Service Outside die Anforderungen der mobilen Applikation entgegennehmen soll, soll dieser unter einer Domain im Internet erreichbar sein. (S. Liste der verfügbaren Domains pro Labor)

Wichtige Hinweise für die Einrichtung der Backendservices

1. Die Zugänge der externen Systeme wie von MySQL Datenbank, SMTP-Server, etc... sind in der Konfigurationsdatei (appsettings.json) als verschlüsselte Werte zu hinterlegen. Zum Verschlüsseln der sensitiven Daten für die Konfigurationsdatei verwenden Sie bitte das mitgelieferten Tool PasswordEncryptor.exe. Diese verschlüsselt die sensitiven Daten und gibt sie als verschlüsselten Wert aus.
2. Bevor die beiden Services hochgefahren werden, sollen die erforderlichen Parameter in den Konfigurationsdateien(appsettings.json) konfiguriert werden. (S. Konfigurationsdatei)
3. Der Service-Inside darf nicht im Internet erreichbar sein.
4. Der Service-Outside muss im Internet unter einer Domain erreichbar sein. (S. Liste der verfügbaren Domains)
5. Der Service-Outside und Service Inside dürfen nicht mit demselben Server betrieben werden
6. Der Service-Inside gibt für Intranet(LIS/MIRTH) das Verzeichnis private für Intranet frei
7. Ein LIS soll mit dem Service-Inside die Daten(Befunde/AckLogs) über die Freigabe(private) zur Verfügung stellen
8. Die MySQL Datenbank soll für die beiden Services erreichbar sein
9. Der Service-Outside legt das Datenbankschema bei der ersten Verwendung selbständig an (Falls die Datenbank noch nicht existiert)
10. Die beiden Services müssen mit den gültigen SSL-Zertifikate betrieben werden(https)

Konfiguration des Service-Inside

Der Service-Inside soll in einer DMZ-Zone betrieben werden. Es ist empfohlen der Service-Inside in einem IIS zu hosten. Dabei ist es wichtig, dass der Webserver das Framework .NET > 6.0 unterstützt. In der Konfigurationsdatei (appsettings.json) vom Service-Inside sollen/können folgende Einstellungen konfiguriert werden. Die Konfigurationsdatei des Service liegt im Hauptverzeichnis vom Service.

appsettings.json (Service-Inside)

```
{
  "Log": {
    "Type": "DEBUG",
    "Path": "c:\\tmp\\inside",
    "SmtpServer": "127.0.0.1",
    "SmtpPort": 25,
    "SmtpSSL": false,
    "SmtpSenderEmail": "ServiceInside@labor-stein.de",
    "SmtpSenderName": "ServiceInside",
    "SmtpRecipients": "info@labor-stein.de"
  },
  "CleanupDays": 30,
  "PrivateDirectory": "c:\\befund_app\\private",
  "PublicDirectory": "c:\\befund_app\\public",
  "MaxWorkerThreads": 10,
  "ServiceOutsideIP": ":::1",
  "ServiceOutsideURL": "https://localhost:443",
  "ConnectionStrings": {
    "default": "dev",
    "prod": "",
    "dev":
      "g0HCqY/I8uctbqjdg+bF0FIzIYQBBCj3wcTeuRn5VfU30u3NHV1E1XyoD0kKPKsLenNGZG45tChvnEsBKKn0iymL+tI7/eeq5rwmS4gsA47j30YUSMdtjhVpumJt8IuN6Yq6s9M="
  }
}
```

Log-Einstellungen zum Protokollieren der verschiedenen Ereignisse im Service

LOG-Typen:

CRITICAL

Markiert das geschehene Ereignis als **kritisch** und löst eine E-Mail-Benachrichtigung aus. (Falls konfiguriert)

ERROR

Markiert das geschehene Ereignis als **Fehler**.

INFO

Markiert das geschehene Ereignis als **Information**.

TRACE

Verbindet das geschehene Ereignis mit einem Auftrag

LOG

Protokolliert das geschehene Ereignis in einer Datei

DEBUG

Protokolliert das geschehene Ereignis in einer Datei und gibt sie auch in die Konsole aus. Dieser Typ vom LOG, enthält am meisten die Information über das Ereignis.

WARNING

Markiert das geschehene Ereignis als **Warnung**. Dabei handelt es sich noch nicht um einen kritischen Fehler. Das System hat an der Stelle ein unerwartetes Ereignis registriert.

LOG-Path

Das Verzeichnis, in dem der Service die Logdateien schreiben darf.

LOG-E-Mail

Zum Versenden der kritischen Log's per E-Mail, soll die Anmeldung am SmtPServer(Transporter) konfiguriert sein.

CleanUpDays

Diese Parameter, definiert die Anzahl der Tage, nachdem die Befunde vom Server gelöscht werden. Standardmäßig sind 30 Tage vorkonfiguriert. D.h. die Befunde bleiben max. 30 Tage auf dem Server vorhanden.

Wichtig: Wird ein Befund von Patienten erfolgreich abgeholt, wird dieser sofort auf dem Server gelöscht.

PrivateDirectory

Das Datenverzeichnis, welches für Intranet als Freigabe zur Verfügung gestellt wird. Das LIS/MIRTH/etc... haben die Dateien (ack.log & .hl7) mit dem ServiceInside über diese Freigabe auszutauschen. (S. Abbildung „Netzwerkinfrastruktur“)

PublicDirectory

Das Datenverzeichnis, wo die verschlüsselten Befunde zum Abholen bereitgestellt werden.

Wichtig: Der Service-Outside, darf die verschlüsselten Befunde nicht über das Filesystem abholen, sondern nur über die integrierten autorisierten Services-Requests(https).

MaxWorkerThreads

Anzahl der Unterprozesse, die die Anforderungen/Dateien (.ack & .hl7) von einem LIS gleichzeitig abarbeiten sollen. Es ist empfohlen, diesen Parameter nicht höher als 10 zu setzen, da der Prozessor stark belastet werden kann.

ServiceOutsideIP

IP-Adresse des Service-Outsides (Diese IP-Adresse wird als Filter der Abfragen verwendet. Die Abfragen vom Service-Outside werden nur von dieser IP-Adresse entgegengenommen).

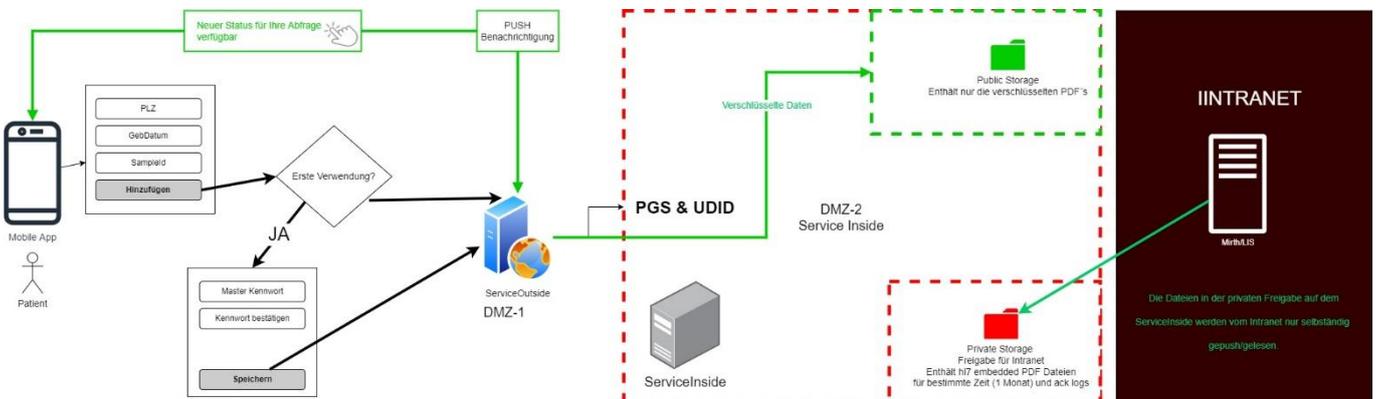
ServiceOutsideURL

Die URL des ServiceOutside für die https Abfragen.

ConnectionStrings

Die Verbindungszeichenfolge der Datenbank. Es ist wichtig, dass die gewünschte Verbindungszeichenfolge über default gewählt wird. Für die Produktion soll bei default **prod** eingetragen werden. Die Verbindungszeichenfolge sind als verschlüsselter Wert zu hinterlegen. Zum Verschlüsseln der Verbindungszeichenfolge, verwenden Sie bitte, das mitgelieferte Tool PasswordEncryptor.exe. Diese finden Sie im Hauptverzeichnis vom Service.

Netzwerkinfrastruktur



Konfiguration des Service-Outside

Der Service-Outside soll in einer DMZ-Zone betrieben werden. Es ist empfohlen der Service-Outside in einem IIS zu hosten. Dabei ist es wichtig, dass der Webserver das Framework .NET > 6.0 unterstützt. Darüber hinaus muss der Service-Outside im Internet unter einer Domain erreichbar sein. (S. Liste der verfügbaren Domains). In der Konfigurationsdatei (appsettings.json) vom Service-Outside sollen/können folgende Einstellungen konfiguriert werden. Die Konfigurationsdatei des Service liegt im Hauptverzeichnis vom Service.

appsettings.json (Service-Outside)

```
{
  "Log": {
    "Type": "DEBUG",
    "Path": "c:\\tmp\\outside",
    "SmtpServer": "172.21.101.88",
    "SmtpPort": 25,
    "SmtpSSL": false,
    "SmtpSenderEmail": "ServiceOutside@labor-stein.de",
    "SmtpSenderName": "ServiceOutside",
    "SmtpRecipients": "info@labor-stein.de"
  },
  "IpRateLimiting": {
    "EnableEndpointRateLimiting": true,
    "StackBlockedRequests": false,
    "RealIPHeader": "X-Real-IP",
    "ClientIdHeader": "X-ClientId",
    "HttpStatusCode": 429,
    "IpWhitelist": [ "127.0.0.1", "::1" ],
    "GeneralRules": [
      {
        "Endpoint": "*",
        "Period": "1m",
        "Limit": 20
      }
    ]
  },
  "ServiceInsideIP": "::1",
  "ServiceInsideURL": "https://localhost:444/",
  "ConnectionStrings": {
    "default": "dev",
    "prod": "",
    "dev":
      "gOHCqY/I8uctbqjdg+bF0FIzIYQBBCj3wcTeuRn5VfU30u3NHV1E1XyoD0kKPKsLenNGZG45tChvnEsBKKn0iymyL+tI7/eeq5rwS4gsA47j30YUSMdtjhVpumJt8IuN6Yq6s9M=",
    "test":
      "HJbFsfLbWM/dfmbVGiFTsYmTHR0yABB9MKdPfyTdWgejmZ1bw2P9wmCLBMrDK8acWymDREnvGRrPwALmpNrCt/HdisrmL25GGGkLswvEyoHw9nq5eBxXSS04H1Q5Q+vZNzCaRr77mRpHV/mAOvIrfnvV9EYWj0PCRhq04ksY2S4="
  },
  "PushNotifications": {
    "APNS": {
      "Type": "sandbox",
      "Cert": "C:\\\\certs\\apple_apns.p12",
      "CertPasswordHash":
        "0hXiNg0507v6YWZREfulN+qCfuBvD8V0QCM08b7ffHTmvosBHvcFPLKGQr228YjkGvuJ8rLZsyKozSTo"
    },
    "GOOGLE": {
      "CredentialJson": "C:\\\\certs\\google_credential.json"
    }
  }
}
```

```
}  
}
```

Log-Einstellungen zum Protokollieren der verschiedenen Ereignisse im Service

LOG-Typen:

CRITICAL

Markiert das geschehene Ereignis als **kritisch** und löst eine E-Mail-Benachrichtigung aus. (Falls konfiguriert)

ERROR

Markiert das geschehene Ereignis als **Fehler**.

INFO

Markiert das geschehene Ereignis als **Information**.

TRACE

Verbindet das geschehene Ereignis mit einem Auftrag

LOG

Protokolliert das geschehene Ereignis in einer Datei

DEBUG

Protokolliert das geschehene Ereignis in einer Datei und gibt sie auch in die Konsole aus. Dieser Typ vom LOG, enthält am meisten die Information über das Ereignis.

WARNING

Markiert das geschehene Ereignis als **Warnung**. Dabei handelt es sich noch nicht um einen kritischen Fehler. Das System hat an der Stelle ein unerwartetes Ereignis registriert.

LOG-Path

Das Verzeichnis, in dem der Service die Logdateien schreiben darf.

LOG-E-Mail

Damit die kritischen Log's per E-Mail versendet werden können, soll der SmtpServer(Transporter) konfiguriert werden.

IpRateLimiting

S. die Dokumentation des Herstellers:

<https://github.com/stefanprodan/AspNetCoreRateLimit/wiki/IpRateLimitMiddleware#defining-rate-limit-rules>

ServiceInsideIP

IP-Adresse des Service-Inside (Diese IP-Adresse wird als Filter der Abfragen verwendet. Die Abfragen vom Service-Inside werden nur von dieser IP-Adresse entgegengenommen.

ServiceInsideURL

Die URL des ServiceInside für die https Abfragen.

ConnectionStrings

Die Verbindungszeichenfolge der Datenbank. Es ist wichtig, dass die gewünschte Verbindungszeichenfolge über default gewählt wird. Für die Produktion soll bei default **prod** eingetragen werden. Die Verbindungszeichenfolge ist als verschlüsselter Wert zu hinterlegen. Zum Verschlüsseln der Verbindungszeichenfolge verwenden Sie bitte, das mitgelieferte Tool PasswordEncryptor.exe. Diese finden Sie im Hauptverzeichnis vom Service.

PushNotifications

Konfiguration für die Push-Benachrichtigungen.

APNS

Konfiguration der Push Benachrichtigungen für die iOS Geräte.

APNS-Type:

Typ des APNS-Servers. Mögliche Werte: **Sandbox** zum Testen, **Production** für das produktive System.

APNS-Cert

Der Pfad zum PUSH-Zertifikat vom developer.apple.com.

INFO: Standardmäßig liegt das PUSH-Zertifikat im Hauptverzeichnis vom Service-Outside. Hier soll nur der lokale Pfad angepasst werden.

APNS- CertPasswordHash

INFO: Für das mitgelieferte Zertifikat soll das Passwort in der Konfigurationsdatei nicht verändert werden. Es ist empfohlen für die PUSH-Konfiguration nur die lokalen Pfade der Zertifikate anzupassen. Die restlichen Einstellungen für die PUSH-Benachrichtigungen sollen möglich unverändert übernommen werden. (App gibt nur einmal für alle Standorte)

GOOGLE

Konfiguration der Push Benachrichtigungen für die Android Geräte.

CredentialJson

Der Pfad zur Google **google_credential.json** Datei.

INFO: Standardmäßig liegt die **google_credential.json** Datei im Hauptverzeichnis vom Service-Outside. Hier soll nur der lokale Pfad angepasst werden.

Freigabe vom ServiceInside (Datenverzeichnis)

Der ServiceInside soll das private Verzeichnis (S. appsettings.json vom ServiceInside) für das interne Netz freigeben, damit das LIS/Mirth die Dateien dort speichern bzw. lesen zu können. Das freigegebene Verzeichnis beinhaltet folgende Struktur: (S. Abbildung)



Die hl7 Dateien (S. hl7 Definition) mit originalen Befunde werden vom LIS/Mirth aus Intranet im Hauptverzeichnis der Freigabe(private) abgelegt. Diese werden nun vom ServiceInside verschlüsselt und dem ServiceOutside im public Verzeichnis zur Verfügung gestellt.

Die ack Log Dateien (S. ack Log Definition) werden vom ServiceInside für LIS/Mirth im Unterordner „ack“ der Freigabe(private) erstellt. Diese Log Dateien sollen vom LIS/Mirth nach dem Lesen ins done Verzeichnis verschoben werden. Dabei ändert sich die Erweiterung der ack Log Datei im done Verzeichnis entweder auf .imp(erfolgreich) oder .non(fehlerhaft) .

HL7 Dateien

Die HL7 Dateien werden von einem LIS dem Service-Inside zur Verfügung gestellt. Diese sollen im Privaten Verzeichnis (S. appsettings.json vom – Service-Inside) abgelegt werden. Die HL7 Dateien beinhaltet folgende Schema/Informationen:

```

MSH|^~\&|[LIS]|||[ZEITSTEMPEL_DATEI]||MDM^TO1||P|2.3|||AL|NE|DE
EVN|TO1|[ZEITSTEMPEL_BEFUND]||[STATUS "COMPLETED", "REJECTED"]
PID|1|[PATIENT_ID]||[GEB_DATUM]||[PLZ]||[GERAETE_UDID]
TXA|1|[AUFTRAGSNUMMER]||[PGS]
OBX||||[BASE64_PDF_BEFUND]
  
```

[LIS]

- Name des LIS

[ZEITSTEMPEL_DATEI]

- Das Erstelldatum der hl7 Datei.

[ZEITSTEMPEL_BEFUND]

- Das Fertigstellungsdatum vom Befund

[STATUS]

- Der Status vom Befund.

Mögliche Werte:

- COMPLETED
- REJECTED

- Der Befund ist fertig und kann abgeholt werden

- Der Befund kann vom LIS nicht bereitgestellt werden

[PATIENT_ID]	- Interne ID des Patienten (Optional)
[GEB_DATUM]	- Das Geburtsdatum des Patienten
[PLZ]	- Die PLZ des Patienten
[GERAETE_UDID]	- Die eindeutigen Geräte-ID von der ersten Registrierung
[AUFTRAGSNUMMER]	- Die eindeutige Auftragsnummer aus LIS
[PGS]	- Der SHA512 Hash von PLZ+Geburtsdatum+Auftragsnummer

Wichtig:

Format für Geb. Datum: (YYYY-MM-DD)

Beispiel:

PLZ: 41238
 Geb.Datum: 1987-01-04
 Auftragsnummer: 0000000001
 Kombiniertes Wert: **412381987-01-040000000001**

Der SHA512 Hash:

502bdf2e4af6d7f7b75e4b8b10700917d7de6bcdea78456e871b52892471d487c858c0219af6da12b99ae21c2fd244d669f7e79f4ce8e4a25a9930837f0566d7

[BASE64_PDF_BEFUND]	- Der PDF-Befund als Base64 Kodierung
---------------------	---------------------------------------

Beispiel hl7 Datei

```

MSH|^~\&|MOLIS|||202220130143010||MDM^T01||P|2.3||AL|NE|DE
EVN|T01|2202220130123010|COMPLETED
PID|1|777|1987-01-04|41238|fx2133123123-31dasd23124-ds23113
TXA|1|0524242211|
OBX|||BASE_64XXXXXX
  
```

.ack Log Dateien

Die .ack Log Dateien werden vom Service-Inside für das LIS im Verzeichnis ack(Unterverzeichnis vom Privatverzeichnis) erzeugt. Diese Dateien beinhalten als Information, den Typ der Anforderung, PLZ, Geb.Datum und die Auftragsnummer. An Hand dieser Dateien kann das LIS auf die Anforderungen der mobilen Applikation entsprechend reagieren.

Abonniert sich ein Patient für seinen Befund, erzeugt der Service-Inside eine ack. Log-Dateien mit folgendem Inhalt für das LIS:

TYPE: SUBSCRIBE**UDID:**[GeräteID] „Beispiel: 9799ab0b3e7ba625_47084651-4182-4d9c-a14c-bc808fe84b05“**PGS:**[PGS] Hash vom PLS+Geb.Datum+Auftragsnummer) „Beispiel: 9799ab0b3e7ba625.....“**ZIP:**[Postleitzahl des Patienten]**BIRTHDATE:**[Geburtsdatum des Patienten Format beachten: YYYY-MM-DD]

ORDER_ID:[Auftragsnummer]
CREATED:[Erstelldatum der log Datei]

Erhält das LIS eine .ack Log Datei vom Service-Inside mit dem Typ: **SUBSCRIBE** soll er die mit gelieferte GeräteID des Patienten in seine Patienten Stammdaten speichern/aktualisieren. Darüber hinaus soll das LIS dabei überprüfen, ob der angeforderte Auftrag bereits fertig ist und diesen mit der GeräteID an den Service-Inside als eine hl7(S. Definition) Datei zurückgeben. Die hl7 Dateien sollen im privaten Verzeichnis vom Service-Inside abgelegt werden. (S. appsettings.json – Service-Inside).

WICHTIG: Nach dem Einlesen der ack Log-Dateien, sind diese vom LIS in das ack/done Verzeichnis zu verschieben. Das verhindert das doppelte Einlesen der gleichen Dateien. Darüber hinaus ist es wichtig, dass LIS die Auftragsnummer eindeutig identifizieren kann. Sollte die Auftragsnummer nicht eindeutig sein(LG-Auftrag?) so hat das LIS die Registrierung mit dem STATUS REJECTED abzulehnen. (S. HL7 Datei STATUS)

Erhält das LIS eine .ack Log Datei vom Service-Inside mit dem Typ: **UNSUBSCRIBE** so kann es die Zustellung(falls noch nicht geschehen) ignorieren. Der Patient hat den abonnierten Befund auf seinem Gerät entfernt. (Die GeräteID soll an der Stelle nicht in Stammdaten des Patienten entfernt werden. Dieser kann noch für die zukünftigen Befunde benötigt werden).

TYPE: **UNSUBSCRIBE**
UDID:[GeräteID] „Beispiel: 9799ab0b3e7ba625_47084651-4182-4d9c-a14c-bc808fe84b05“
PGS:[PGS] Hash vom PLS+Geb.Datum+Auftragsnummer) „Beispiel: 9799ab0b3e7ba625.....“
ZIP:[Postleitzahl des Patienten]
BIRTHDATE:[Geburtsdatum des Patienten Format beachten: YYYY-MM-DD]
ORDER_ID:[Auftragsnummer]
CREATED:[Erstelldatum der log Datei]

Erhält das LIS eine .ack Log Datei vom Service-Inside mit dem Typ: **PICKEDUP**, so kann es den Befund als abgeholt markieren(z.B. in Auftragsjournal)

TYPE: **PICKEDUP**
UDID:[GeräteID] „Beispiel: 9799ab0b3e7ba625_47084651-4182-4d9c-a14c-bc808fe84b05“
PGS:[PGS] Hash vom PLS+Geb.Datum+Auftragsnummer) „Beispiel: 9799ab0b3e7ba625.....“
ZIP:[Postleitzahl des Patienten]
BIRTHDATE:[Geburtsdatum des Patienten Format beachten: YYYY-MM-DD]
ORDER_ID:[Auftragsnummer]
CREATED:[Erstelldatum der log Datei]

Erhält das LIS eine .ack Log Datei vom Service-Inside mit dem Typ: **DELETE_DEVICE**, so soll es die GeräteID aus den Stammdaten des Patienten entfernen. Der Patient hat seinen Account gelöscht und es dürfen keine weiteren Befunde per App zugestellt werden. (Auch auf den Services werden alle zugehörigen Daten des Patienten bereinigt).

TYPE: **DELETE_DEVICE**
UDID:[GeräteID] „Beispiel: 9799ab0b3e7ba625_47084651-4182-4d9c-a14c-bc808fe84b05“
PGS:[PGS] Hash vom PLS+Geb.Datum+Auftragsnummer) „Beispiel: 9799ab0b3e7ba625.....“
ZIP:[Postleitzahl des Patienten]

BIRTHDATE:[Geburtsdatum des Patienten Format beachten: YYYY-MM-DD]

ORDER_ID:[Auftragsnummer]

CREATED:[Erstelldatum der log Datei]

Erhält das LIS eine .ack Log Datei vom Service-Inside mit dem Typ: **ERROR**, bedeutet, dass der Befund(hl7 Datei) fehlerhaft war und dieser vom Service-Inside nicht verarbeitet werden konnte. Dazu erzeugt der Service-Inside eine log Datei, welche die Information über den ausgelösten Fehler beinhaltet.

TYPE: **DELETE_DEVICE**

UDID:[GeräteID] „Beispiel: 9799ab0b3e7ba625_47084651-4182-4d9c-a14c-bc808fe84b05“

PGS:[PGS] Hash vom PLS+Geb.Datum+Auftragsnummer) „Beispiel: 9799ab0b3e7ba625.....“

ZIP:[Postleitzahl des Patienten]

BIRTHDATE:[Geburtsdatum des Patienten Format beachten: YYYY-MM-DD]

ORDER_ID:[Auftragsnummer]

FILENAME:[Fehlerhafte Datei]

REASON:[Fehler Beschreibung]

CREATED:[Erstelldatum der log Datei]

MySQL Datentabelle

device – Tabelle

Die Tabelle device beinhaltet die Informationen über die Patienten-Geräte.

udid	-	Die Eindeutige GeräteID des Patienten
device_token	-	Device Token für den PUSH-Service
verificator_hash	-	HMAC Hash für das Benutzerpasswort (Eine Zurückrechnung des Passwortes ist nicht möglich)
public_key	-	Der letzte bekannte Öffentliche Schlüssel der mobilen App von Patienten
pin	-	PIN zum Zurücksetzen vom Benutzerpasswort
device_type	-	Typ vom Gerät (iOS oder Android)
created	-	Erstelldatum vom Tabelleneintraf
modified	-	Datum der letzten Aktualisierung vom Tabelleneintrag

results – Tabelle

pgs	-	PGS Hash (SHA512 – PLZ + Geb.Datum + Auftragsnummer)
pgs_hash	-	AES Verschlüsselter Wert von PGS. Dieser Wert kann vom Servicelnise entschlüsselt werden somit kann er die Auftragsnummer, PLZ und Geb.Datum für LIS zurückrechnen.
udid	-	Die Eindeutige GeräteID des Patienten
client_public_key	-	Öffentlicher Schlüssel der mobilen App von Patienten Damit werden die Befunde verschlüsselt
server_public_key	-	Öffentlicher Schlüssel des Servers Damit werden die Befunde verschlüsselt
available	-	Flag ob der Befund verfügbar ist
available_ts	-	Zeitstempel der Verfügbarkeit vom Befund
picked_up	-	Flag ob der Befund abgeholt wurde
picked_up_ts	-	Zeitstempel der Abholung vom Befund
file_checksum	-	Prüfsumme des Inhalts vor der Verschlüsselung
notified	-	Flag ob der Patient zum Befund eine PUSH-Benachrichtigung erhalten hat
notified_ts	-	Zeitstempel der PUSH-Benachrichtigung
modified	-	Zeitstempel der letzten Aktualisierung vom Tabelleneintrag
status	-	Aktueller Status vom Befund (COMPLETED, REJECTED, EXPIRIED)
created	-	Erstelldatum vom Tabelleneintrag

trace – Tabelle

id	-	Eindeutige ID des Tabelleneintrages
pgs	-	PGS Hash (SHA512 – PLZ + Geb.Datum + Auftragsnummer)
udid	-	Die Eindeutige GeräteID des Patienten
message	-	Nachricht über das Ereignis
pat_hash	-	SHA512 Hash für die interne Patientenid
trace_type	-	Typ vom Trace
created	-	Erstelldatum vom Tabelleneintrag

Liste der verfügbaren Domains für OutsideService

```
LABOR_HEIDELBERG("patientenbefundapp.labor-limbach.de"),
LABOR_LUDWIGSBURG("patientenbefundapp.mvz-labor-lb.de"),
LABOR_ULM("patientenbefundapp.humangenetik-ulm.de"),
LABOR_AACHEN("patientenbefundapp.labor-aachen.de"),
LABOR_BERLIN("patientenbefundapp.mdi-limbach-berlin.de"),
LABOR_COTTBUS("patientenbefundapp.labor-cottbus.de"),
LABOR_DESSAU("patientenbefundapp.laborpraxis-dessau.de"),
LABOR_DORTMUND("patientenbefundapp.labor-dortmund.de"),
LABOR_DRESDEN("patientenbefundapp.labordresden.de"),
LABOR_ERFURT("patientenbefundapp.labor-erfurt.de"),
LABOR_ESSEN("patientenbefundapp.labor-eveld.de"),
LABOR_FREIBURG("patientenbefundapp.labor-clotten.de"),
LABOR_HANNOVER("patientenbefundapp.labor-limbach-hannover.de"),
LABOR_HOFHEIM("patientenbefundapp.labor-hofheim.de"),
LABOR_KARLSRUHE("patientenbefundapp.laborvolkmann.de"),
LABOR_LEIPZIG("patientenbefundapp.labor-leipzig.de"),
LABOR_MOENCHENGLADBACH("patientenbefundapp.labor-stein.de"),
LABOR_MUENCHEN("patientenbefundapp.labor-limbach-muenchen.de"),
LABOR_MUENSTER("patientenbefundapp.labor-muenster.de"),
LABOR_NUERBERG("patientenbefundapp.labor-limbach-nuernberg.de"),
LABOR_PASSAU("patientenbefundapp.labor-passau.de"),
LABOR_RAVENSBURG("patientenbefundapp.labor-gaertner.de"),
LABOR_ROSENHEIM("patientenbefundapp.medlabor.de"),
LABOR_SCHWEINFURT("patientenbefundapp.laboraerzte-schweinfurt.de"),
LABOR_SCHWERIN("patientenbefundapp.labor-schwerin.de"),
LABOR_STRALSUND("patientenbefundapp.labor-stralsund.de"),
LABOR_SUHL("patientenbefundapp.labor-suhl.de"),
LABOR_KASSEL("MISSING"),
LABOR_HANNOVER_MLM("patientenbefundapp.mlh.de"),
LABOR_MAINZ("patientenbefundapp.medgen-mainz.de"),
LABOR_FRANKFURT("patientenbefundapp.laborarztpraxis.de");
```

Beispiel von MOLIS Anbindung

