
ISBJ Trägerportal

Entwicklerleitfaden für die Dienstschnittstelle

NORTAL AG



April 2026, v2.0.2

Inhaltsverzeichnis

1	Einleitung	4
2	Grundlagen	4
2.1	Fachliche Grundlagen	4
2.2	Organisatorische Grundlagen	5
2.2.1	Beteiligte Personen und Rollen	5
2.3	Technische Grundlagen	6
2.4	Daten und Informationen	7
3	Umgebung des ISBJ Trägerportals und der zugehörigen Dienstschnittstelle	7
3.1	Umgebungen	7
3.1.1	Abnahmeumgebung	8
3.1.2	Produktivumgebung	8
3.2	Zugang zum ISBJ Trägerportal	8
3.2.1	Daten für die Betreiber des ISBJ Trägerportals	8
3.2.2	Checkliste zur Kommunikation mit der Dienstschnittstelle	9
3.3	Aufruf des Trägerportals im Webbrowser	9
3.4	Anmeldung am Trägerportal als Nutzer der Dienstschnittstelle	9
4	Authentifikation und Autorisation	11
4.1	Authentifizierung	11
4.1.1	1. Faktor: HTTPS-Client-Zertifikat (mTLS)	12
4.1.2	2. Faktor: Message Authentication Code	12
4.2	Autorisation	14
5	Personalmeldungen, Kindzuschläge, Kitaverzeichniseinträge und Vormerkungen	14
	REST API der Dienstschnittstelle anhand des XSD Schema	14
5.1	Datenlieferung und Abfrage	14
5.1.1	Datenbestand	14
5.1.2	Status- und Detailabfragen	15
5.1.3	Asynchrone Datenlieferung	15
5.1.4	Technische Validierung	16
5.1.5	Fachliche Validierung und Protokollabfragen	16
5.1.6	Prüfsummenvalidierung und -berechnung	16
5.2	Schnittstellenendpunkte	21
5.2.1	XSD-Schema	21
5.2.2	URL	22
5.2.3	HTTP-Request-Body	22
5.2.4	Anwendungsfälle und Funktionen	24

5.2.5	Eigenschaften der Dienstschnittstelle	31
6	Vertragsverwaltung RESTful API der Dienstschnittstelle	33
6.1	Authentifikation und Autorisation	33
6.1.1	HMAC	33
6.2	API Endpunkte und Operationen	33
6.3	Einen Gutschein abrufen	33
6.3.1	Anfrageparameter	33
6.3.2	Antworten	33
6.4	Verträge eines Gutscheins abrufen	35
6.4.1	Anfrageparameter	35
6.4.2	Antworten	36
6.5	Vertrag registrieren	37
6.5.1	Anfrageparameter	38
6.5.2	Anfragedaten (request body parameter)	38
6.5.3	Antworten	40
6.6	Verträge abrufen	42
6.6.1	Anfrageparameter	42
6.6.2	Antworten	43
6.7	Einen Vertrag abrufen	44
6.7.1	Anfrageparameter	45
6.7.2	Antworten	45
6.8	Betreuungsumfang erhöhen	47
6.8.1	Anfrageparameter	47
6.8.2	Anfragedaten (request body parameter)	47
6.8.3	Antworten	50
6.9	Betreuungsumfang verringern	51
6.9.1	Anfrageparameter	51
6.9.2	Anfragedaten (request body parameter)	52
6.9.3	Antworten	53
6.10	Laufzeit anpassen	54
6.10.1	Anfrageparameter	55
6.10.2	Anfragedaten (request body parameter)	55
6.10.3	Antworten	56
6.11	Vertrag stornieren	57
6.11.1	Anfrageparameter	57
6.11.2	Anfragedaten (request body parameter)	58
6.11.3	Antworten	58
6.12	Voraussichtliches Vertragsende melden (Schulrückstellung)	59
6.12.1	Anfrageparameter	59

6.12.2	Anfragedaten (request body parameter)	60
6.12.3	Antworten	60
6.13	Voraussichtliches Vertragsende widerrufen	61
6.13.1	Anfrageparameter	61
6.13.2	Anfragedaten (request body parameter)	61
6.13.3	Antworten	62
6.14	Anträge eines Kindes abrufen	63
6.14.1	Anfrageparameter	63
6.14.2	Antworten	63
6.15	Antragdetails abrufen	65
6.15.1	Anfrageparameter	65
6.15.2	Antworten	65
6.16	Gutscheine eines Antrags abrufen	66
6.16.1	Anfrageparameter	66
6.16.2	Antworten	67
7	Posting RESTful API der Dienstschnittstelle	68
7.1	Authentifikation und Autorisation	69
7.1.1	HMAC	69
7.2	API Endpunkte und Operationen	69
7.3	Postings abrufen	69
7.3.1	Anfrageparameter	69
7.3.2	Antworten	70
7.4	Ein Posting abrufen	71
7.4.1	Anfrageparameter	71
7.4.2	Antworten	71

Abbildungsverzeichnis

1	Anmeldemaske des Trägerportals	10
2	Startseite des Trägerportals	11

Tabellenverzeichnis

1	Relevante Datenhaltungsorte für Statusabfragen je nach Anwendungsfall	15
---	---	----

1 Einleitung

Dieses Dokument richtet sich an Entwickler, die eine bestehende Fachanwendung über die Dienstschnittstelle an das ISBJ Trägerportal anbinden möchten. Es bietet eine detaillierte Anleitung zur Vorbereitung einer Clientanwendung für die Kommunikation mit der Dienstschnittstelle. Die Dienstschnittstelle bietet Zugriff auf bestimmte Funktionen des ISBJ Trägerportals, die von der Clientanwendung aufgerufen werden können. Das Aufrufen der Dienstschnittstelle über das Internet erfolgt mittels RESTful¹, welcher als Transportprotokoll HTTP(S) verwendet. Das Nachrichtenformat ist vom Typ XML. Die Form wird durch eine Schemadefinitionsdatei (XSD) vorgegeben.

Der vorliegende Leitfaden befasst sich sowohl mit organisatorischen Grundlagen als auch mit den technischen Aspekten der Anbindung. Er beschreibt, wie die Dienstschnittstelle programmatisch angesprochen wird, welche Daten in welchem Format vorliegen müssen und welche Rückmeldungen zu erwarten sind.

2 Grundlagen

2.1 Fachliche Grundlagen

Die erfolgreiche Anbindung einer bestehenden Fachanwendung an die Dienstschnittstelle erfordert detaillierte Kenntnisse der Fachanwendung und der ISBJ-Anwendungslandschaft.

Besteht über die Fachanwendung oder über die ISBJ-Anwendungslandschaft Informationsbedarf, kann über den Träger-Service Kontakt mit der Senatsverwaltung für Bildung, Jugend und Familie (SenBJF) aufgenommen werden. So können frühzeitig fachliche Fragen beantwortet bzw. relevante fachliche Konzepte oder Auszüge daraus zur Verfügung gestellt werden. Eine

¹REST - *Representational State Transfer*, ein Architekturstil, um APIs über HTTP-Semantik abzubilden.

Implementierung auf rein technischer Ebene ist zwar möglich, dieses Vorgehen kann jedoch zu unnötigen mehrfachen Implementierungsversuchen führen.

2.2 Organisatorische Grundlagen

Da es sich beim Trägerportal und dessen Dienstschnittstelle als Teil des ISBJ-Fachverfahrens um ein komplexes System mit verschiedenen Rollen und Rechten handelt, ist es notwendig, die für die Nutzung der Dienstschnittstelle relevanten Rollen abzugrenzen. Des Weiteren ist es erforderlich, alle relevanten Technologien aufzuzählen, welche bei der Kommunikation mit der Dienstschnittstelle in Verwendung sind. Dies dient dazu, einen Rahmen zu errichten, in welchem die Entwicklung einer externen Clientanwendung stattfinden kann.

2.2.1 Beteiligte Personen und Rollen

Für die Anbindung von Clientanwendungen an die Dienstschnittstelle ist die Kommunikation zwischen den Mandanten und den Betreibern des ISBJ Trägerportals und dessen Dienstschnittstelle ein integraler Bestandteil des Prozesses. Daher ist es wichtig, die Verantwortlichkeiten auf beiden Seiten klar zu definieren. Die unten beschriebenen Rollen müssen nicht auf unterschiedliche Personen verteilt sein.

2.2.1.1 Betreiber des ISBJ Trägerportals und der zugehörigen Dienstschnittstelle

Der Mandanten-Betreuer (ISBJ-Träger-Service) ist die Person, die direkt für den Austausch von Daten und Dateien zwischen Betreibern und Mandanten zuständig ist. Konkret wird diese Aufgabe durch den Träger-Service von SenBJF erfüllt. Der Träger-Service hat die Aufgabe, Mandanten-Informationen entgegenzunehmen und in der Portal-Benutzerverwaltung (Portal-BNV) zu pflegen. Es geht dabei insbesondere um das Erstellen eines Administrators und das Ausstellen des zugehörigen Zertifikats. Gleichzeitig liefert der Träger-Service Informationen, die die Mandanten für die Anbindung benötigen. Dazu gehören die Zugangsdaten des Administrators sowie das Zertifikat.

Weiterhin ist der Betreiber dafür verantwortlich, dass die Firewall mit den Zertifikatsinformationen gepflegt wird, um die Kommunikation zu ermöglichen.

2.2.1.2 Mandanten des ISBJ Trägerportals und der zugehörigen Dienstschnittstelle

Der Prozessverantwortliche (Administrator im Trägerportal) ist eine administrative Rolle im Betrieb des Mandanten, der in Absprache mit dem Mandanten-Betreuer die Nutzerrechte und Zugänge für den jeweiligen Träger definiert.

Dieser Person werden das Zertifikat und die Zugangsdaten für den Administrator im Trägerportal zugewiesen. Anschließend ist der Prozessverantwortliche zuständig für die Einrichtung von Benutzern und von deren Rechten. Es muss für die Clientanwendung mindestens ein Benutzer im Trägerportal eingerichtet sein, da ohne diesen Benutzer nicht mit der Clientanwendung auf die Dienstschnittstelle zugegriffen werden kann.

Dienstschnittstellen-Benutzer haben die Möglichkeit, unterschiedliche Anwendungsfälle über die Dienstschnittstelle auszuführen.

Entwickler sind die Personen in der Organisation des Mandanten, die mit der Erstellung und Pflege der Clientanwendung betraut sind. Sie sind die primäre Zielgruppe dieses Dokuments, da sie für die programmatische Anbindung der Clientanwendung verantwortlich sind. Zu ihren Aufgabenbereichen zählt auch die Einbindung der Zertifikate.

2.3 Technische Grundlagen

Es werden keine Programmierkenntnisse in einer bestimmten Sprache für die Verwendung der Dienstschnittstelle vorausgesetzt, da die Kommunikation über eine REST-Schnittstelle erfolgt. Grundlegende Kenntnisse in Java sind jedoch empfehlenswert, wenn versucht wird, die Kommunikation mithilfe des Referenzclients und anhand von Programmierbeispielen nachzuvollziehen. Ggf. sind Maven-Kenntnisse erforderlich, wenn der Referenzclient gebaut werden soll.

Vorausgesetzt werden allgemeine Kenntnisse in den folgenden Bereichen:

- Grundlegende Kenntnisse der Konzepte von **RESTful APIs** und der verschiedenen Methodenarten sollten vorhanden sein.
- **TLS (HTTPS)** wird zur verschlüsselten Übertragung eingesetzt.
- Grundlegende **XML**-Kenntnisse sind erforderlich.
- In **XSD** genügt es, das vorgegebene Schema und v. a. darin festgelegte Formate und Restriktionen zu verstehen. Für den Fall, dass in einer Sprache Modellklassen aus dem XSD generiert werden sollen, sind ggf. Kenntnisse im Umgang mit entsprechenden Bibliotheken für die jeweilige Sprache nötig. Bei Java sind beispielsweise Erfahrungen im Umgang mit JAXB hilfreich.
- Um ungültige Anfragen aufgrund fachlich falscher Daten zu vermeiden, empfiehlt es sich, diese bereits vor dem Versand abzufangen. Dafür ist ein **fachliches Verständnis** der Daten erforderlich. Anderenfalls werden die fachlich ungültigen Daten mit entsprechenden Fehlercodes, welche später noch beschrieben werden, quittiert.

2.4 Daten und Informationen

Folgende Informationen sind für die Verwendung des Trägerportals über die Dienstschnittstelle erforderlich:

Benutzerzugang	bestehend aus Benutzername und API-Key. Alle Anfragen an die Dienstschnittstelle müssen mit diesen Daten autorisiert werden.
TLS-Client-Zertifikat	Es dient als zusätzliche Absicherung der verschlüsselten Verbindung.
URL des Trägerportals	Über die URL können die Benutzer der Dienstschnittstelle die eigenen Rechte einsehen.
URL des REST-Services	Über diese URL können die verschiedenen Services der Dienstschnittstelle aufgerufen werden.
Übersicht der Services	Sie beinhaltet eine Aufzählung, welche Dienste von der Dienstschnittstelle bereitgestellt werden.
XSD-Schema	Es beschreibt die Form der XML-Dateien, die die Dienstschnittstelle entgegennimmt oder die als Antwort auf Abfragen generiert werden.

Woher diese Informationen stammen und wie sie im Prozess der Dienstschnittstellennutzung verwendet werden, wird in den folgenden Kapiteln vermittelt.

3 Umgebung des ISBJ Trägerportals und der zugehörigen Dienstschnittstelle

Dieses Kapitel beschreibt die verschiedenen Umgebungen des Trägerportals sowie der Dienstschnittstelle und organisatorische Details zu den Umgebungen.

3.1 Umgebungen

Die beiden folgenden Umgebungen werden von SenBJF angeboten und stehen Nutzenden des Trägerportals und der Dienstschnittstelle zur Verfügung.

- Abnahmeumgebung
- Produktivumgebung

Der Beantragungsprozess für die Zugangsdaten des Produktivsystems muss zwischen dem Dienstleister und dessen Auftraggeber abgestimmt werden.

3.1.1 Abnahmeumgebung

Diese Umgebung ist zu verwenden, um echte HTTPS-Zugriffe mit Clientzertifikaten zu testen und die Korrektheit der Clientanwendung gegen eine abzunehmende Version der Dienstschnittstelle zu verifizieren. Positive Tests auf der Abnahmeumgebung sind die Voraussetzung für die Produktivumgebung des Clients.

Die Abnahmeumgebung wird im IT-Dienstleistungszentrum Berlin (ITDZ) betrieben.

Für die Vergabe der Benutzer und Zertifikate ist der ISBJ-Träger-Service zuständig. Zugangsdaten können per E-Mail mit dem Betreff "Dienstschnittstelle" an die Adresse traeger-service@senbjf.berlin.de beantragt werden.

3.1.2 Produktivumgebung

Auch die Produktivumgebung wird im ITDZ betrieben. Hier sollten die Träger bereits Zertifikate und Administratorkennungen beantragt haben, damit weitere Benutzer über das Trägerportal eingerichtet werden können. Für neue Administratorkennungen muss die Rolle Dienstschnittstelle vom Träger-Service SenBJF zugeordnet sein, um sie Benutzern vererben zu können.

3.2 Zugang zum ISBJ Trägerportal

In diesem Abschnitt wird beschrieben, welche Schritte zur Verwendung der Dienstschnittstelle notwendig sind. Dabei wird an den entsprechenden Stellen auf die Unterschiede zwischen den zwei Umgebungen hingewiesen. Die ersten Schritte dienen dazu zu prüfen, ob von den jeweiligen Rechnern bzw. vom Netzwerk auf das Trägerportal zugegriffen werden kann. Es wird ebenso aufgezeigt, wie ein Benutzer anzulegen ist bzw. wie die Rechte eines Benutzers zu überprüfen sind, bevor versucht wird, mit diesem auf die Dienstschnittstelle zuzugreifen.

3.2.1 Daten für die Betreiber des ISBJ Trägerportals

Zuerst müssen die Nutzerdaten für die jeweilige Umgebung angefordert werden. Diese Informationen werden per E-Mail an den Betreiber übermittelt. Zu übermitteln sind jeweils:

- Nachname
- Vorname
- E-Mailadresse
- Telefon
- Anschrift (Straße, Hausnummer, PLZ, Ort)
- Name und/oder Nummer des Trägers, für den der Zugang eingerichtet werden soll

- Name und/oder Nummern der Einrichtungen, für die der Zugang eingerichtet werden soll
- Auflistung der Rechte, welche der Benutzer für die Einrichtungen benötigt

3.2.2 Checkliste zur Kommunikation mit der Dienstschnittstelle

Bevor mit der Dienstschnittstelle gearbeitet werden kann, sollte sichergestellt sein, dass alle relevanten Zugangsdaten für die Nutzung der Dienstschnittstelle vorhanden sind. Hier ist eine Checkliste aller Voraussetzungen:

- Haben Sie die Nutzerdaten vom Betreiber erhalten?
 - Liegt Ihnen Ihr Nutzernamen vor?
 - Liegt Ihnen Ihr Nutzerkennwort vor?
- Haben Sie ein Zertifikat vom Betreiber zugewiesen bekommen?
 - Liegt Ihnen das Zertifikat in PKCS12-Format (`dateiname.p12`) vor?
 - Liegt Ihnen das Installationskennwort des Zertifikats als Schreiben oder PDF-Dokument vor?
- Haben Sie die aktuelle Version der XML-Schemadefinition (`isbj-dienstschnittstelle-1.24.0.xsd`) der Dienstschnittstelle erhalten?
- Haben Sie die URL für die Zugriffe erhalten?
 - URL des ISBJ Trägerportals
 - URL der Dienstschnittstelle
 - URL der Dienstschnittstelle Anfrageliste

Falls eine oder mehrere Voraussetzungen fehlen, muss der Betreiber kontaktiert werden.

3.3 Aufruf des Trägerportals im Webbrowser

Nach Erhalt der benötigten Zugangsdaten kann das Trägerportal unter der übermittelten URL aufgerufen werden.

Hierbei ist darauf zu achten, dass das übermittelte Zertifikat im verwendeten Browser installiert ist. Unterstützung bei der Installation des Zertifikats kann über den Träger-Service von SenBJF unter traeger-service@senbjf.berlin.de erbeten werden.

3.4 Anmeldung am Trägerportal als Nutzer der Dienstschnittstelle

Das Portal empfängt Sie mit folgender Startseite:

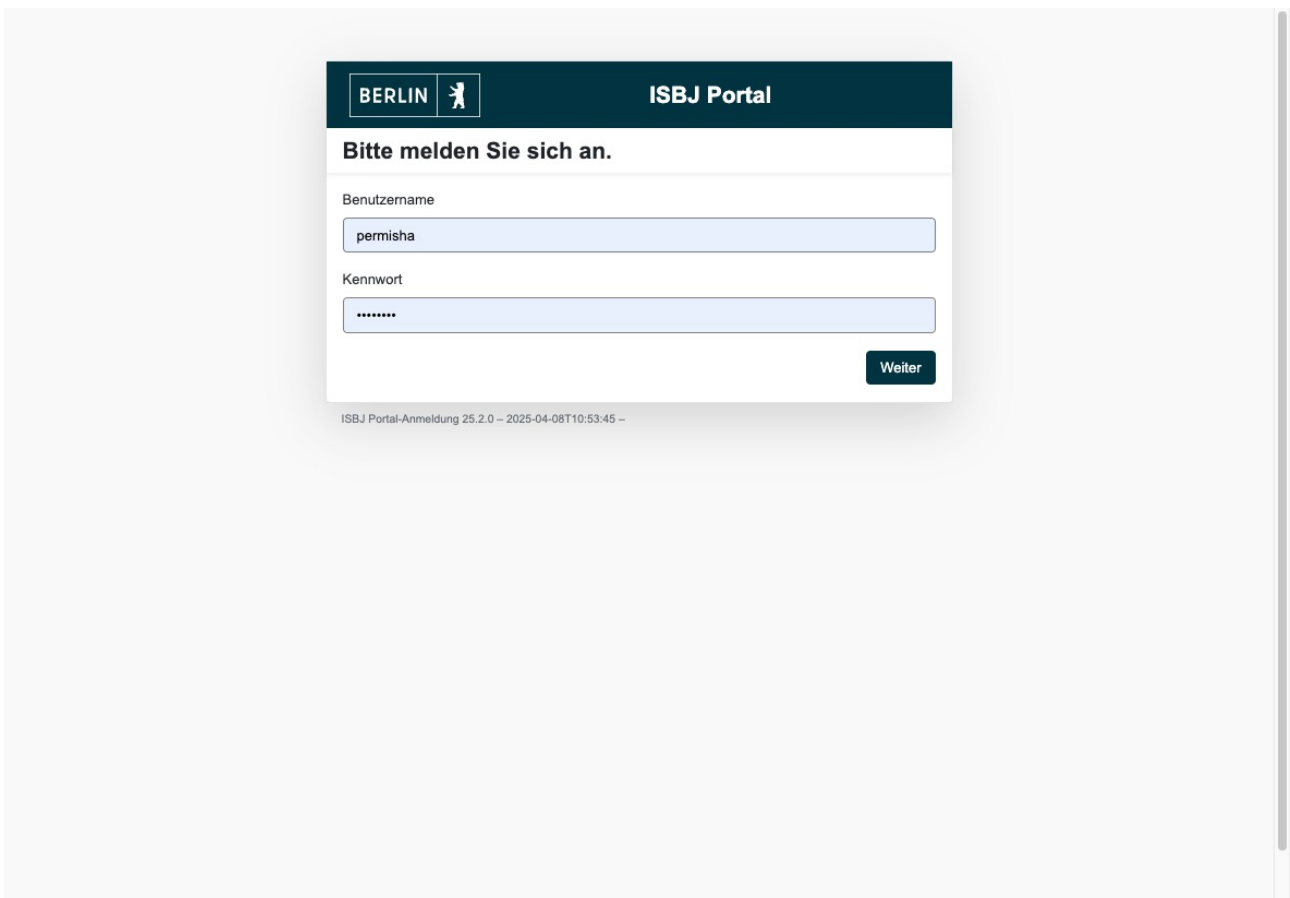


Abbildung 1 – Anmeldemaske des Trägerportals

Tragen Sie in der folgenden Eingabemaske den übermittelten *Benutzernamen* und das zugehörige *Passwort* ein.

Nachdem Sie sich erfolgreich angemeldet haben, wird die Übersicht des Trägerportals angezeigt. Dies dient als Verifikationsmöglichkeit für den Zugang zu Systemen des ISBJ Trägerportals.

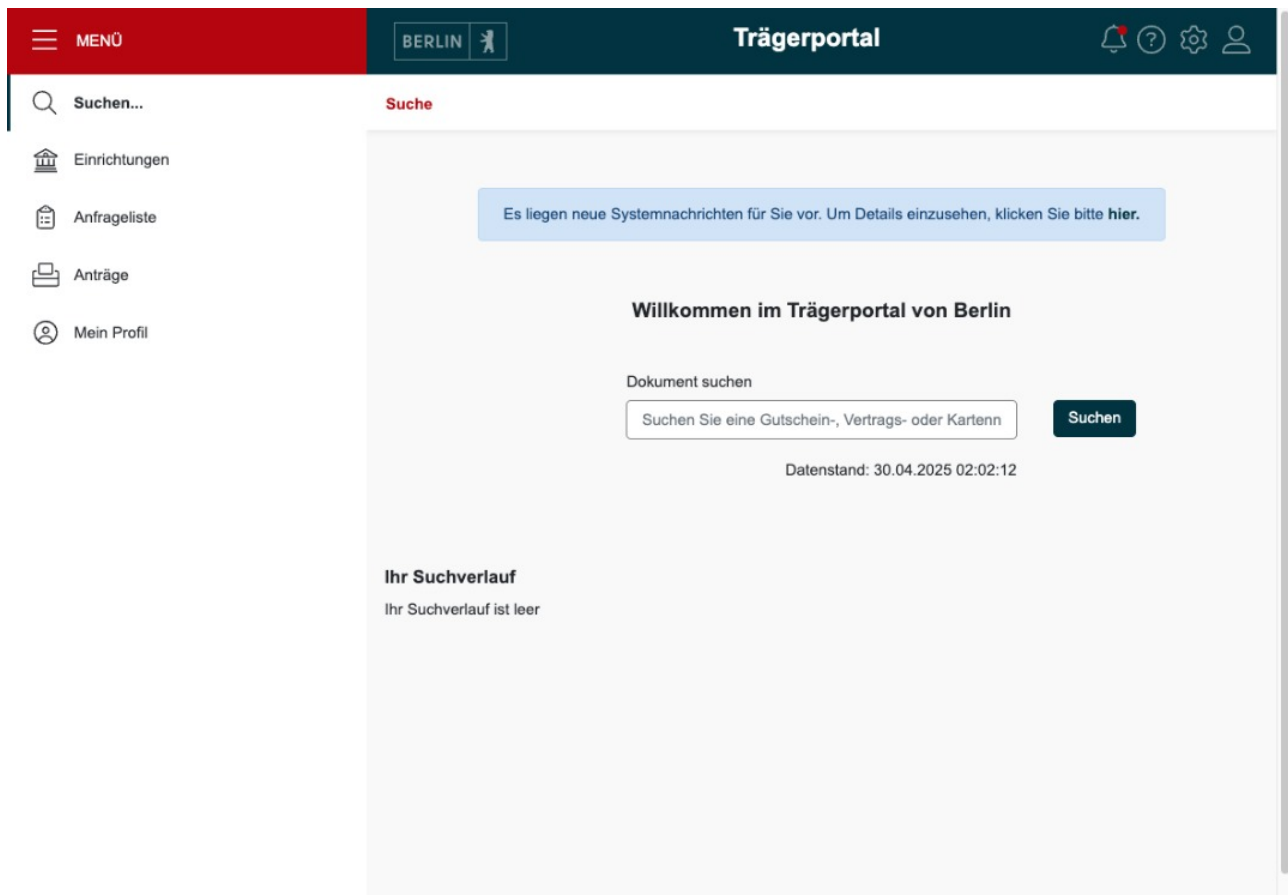


Abbildung 2 – Startseite des Trägerportals

4 Authentifikation und Autorisation

Der Zugriff auf die Dienstschnittstelle kann nur durch autorisierte Benutzer erfolgen. Aus diesem Grund müssen aufrufende Systeme authentifiziert werden.

4.1 Authentifizierung

Die Authentifizierung an der Dienstschnittstelle erfolgt auf Basis von zwei Faktoren. Diese orientieren sich an der Zwei-Faktor-Authentifizierung, die aktuell im ISBJ Trägerportal angewendet wird.

4.1.1 1. Faktor: HTTPS-Client-Zertifikat (mTLS)

Der erste Faktor der Authentifizierung ist ein TLS-Client-Zertifikat, das bei jedem Aufruf verwendet werden muss. Das Verfahren wird als *mutual TLS* (mTLS) bezeichnet. Weitere Informationen dazu sind unter diesem Stichwort online einsehbar; die konkrete Implementierung ist dabei abhängig von der Plattform (Programmiersprache, Frameworks, Betriebssysteme), die für die Trägersoftware zum Einsatz kommen.

Als Zertifikat kommt aktuell das identische Zertifikat zum Einsatz, welches auch für den Zugriff auf das ISBJ Trägerportal im Browser verwendet wird und welches durch den Trägerservice von ISBJ automatisch verwendet wird. Weitere Fragen zu dem Zertifikat kann der zuständige Administrator oder der ISBJ Trägerservice beantworten.

Aktuell ist absehbar, dass das Zertifikat für den mTLS Zugriff auf die Dienstschnittstelle unabhängig vom ISBJ Trägerportalzertifikat wird. In diesem Fall wird der ISBJ Trägerservice eine entsprechende Mitteilung ausgeben. An der technischen Implementierung ändert dies jedoch nichts.

4.1.2 2. Faktor: Message Authentication Code

Der zweite Faktor der Authentifizierung besteht in der Verwendung eines Keyed-Hash Message Authentication Code (HMAC). Dieser wird für jeden Request individuell berechnet und bezieht alle wesentlichen Attribute der Anfrage mit ein.

Für die Berechnung des HMAC ist ein geheimer API-Schlüssel notwendig. Dieser wird dem Träger bei der Freischaltung für die Dienstschnittstelle durch die Senatsverwaltung mitgeteilt. Bei Bedarf kann ein neuer API-Schlüssel über die Anwendung *Portal Selfservice* angefordert werden.

Bei dem API-Schlüssel handelt es sich um einen kryptografischen Schlüssel, der zur Berechnung des HMAC verwendet werden muss. Er fungiert als vorab geteiltes Geheimnis (*shared secret*) zwischen Trägersoftware und Dienstschnittstelle.

Wichtig: Der API-Schlüssel wird *niemals* als Teil einer Anfrage an die Dienstschnittstelle übertragen. Es wird *nur* der HMAC als abgeleiteter Wert übertragen. Aus dem HMAC ist der API-Schlüssel *nicht* rekonstruierbar.

Zur Berechnung des HMAC werden die folgenden Informationen herangezogen:

- HTTP-Methode (HTTP Verb) des Aufrufs, bspw. GET oder POST
- normalisierter URL-Pfad des Aufrufs
- MD5-Prüfsumme des Anfrageinhaltes (kompletter request body)
- Zeitstempel der Anfrage

Diese Informationen werden durch Zeilenumbrüche miteinander verbunden und bilden eine Zeichenkette, die als Eingabe für den HMAC dient. Das Ergebnis des HMAC wird anschließend *base64* enkodiert als HTTP-Anfrage-Header mit übertragen. Dazu wird der Header *Authorization* verwendet. Als Schlüsselwort, das den Autorisationstyp beschreibt (*authorization scheme*), wird *HMAC* verwendet. Anschließend werden Nutzernamen gefolgt von einem Doppelpunkt und dem HMAC angegeben.

Die Berechnung des HMACs erfolgt mittels des SHA256-Algorithmus entsprechend [RFC 2104](#).

Referenzimplementierungen für verschiedene Programmiersprachen sind verfügbar bzw. Teil der Standard-Bibliotheken verschiedener Plattformen (z.B. Java).

Leere Request Bodys, die bspw. bei GET-Anfragen verwendet werden, werden einheitlich mit einer MD5-Prüfsumme `d41d8cd98f00b204e9800998ecf8427e` bewertet.

4.1.2.1 Beispiel

Das folgende Beispiel demonstriert die Authentifikation für einen exemplarischen Aufruf.

Nutzername	dienstschnittstelle-demo-user
API-Schlüssel	6EJV9vmAD/JBEv6n14o0EpZy5ijUk5miF4+T/VVyH34=
HTTP Methode	POST
Normalisierter URL-Pfad	/api/v1/betreuung/gutscheine/GB-123456789-00/vertragregistrieren
kompletten Request-Bodys	b1946ac92492d2347c6235b4d2611184
HTTP Format - RFC1123)	Tue, 12 Jun 2018 15:04:00 GMT; dies <i>MUSS</i> exakt dem Wert entsprechen, der im HTTP-Anfrage-Header <i>Date</i> übertragen wird.

Die Zeichenfolge, für die der HMAC berechnet wird, wäre dementsprechend:

```
POST
/api/v1/betreuung/gutscheine/GB-123456789-00/vertragregistrieren
b1946ac92492d2347c6235b4d2611184
Tue, 12 Jun 2018 15:04:00 GMT
```

Dies entspricht der Verknüpfung der Zeichenketten für

- HTTP-Methode
- Normalisierter URL-Pfad
- MD5-Prüfsumme des Request-Body
- Zeitstempel der Anfrage im HTTP-Format

Nach dem Signieren mit dem Schlüssel `6EJV9vmAD/JBEv6n14o0EpZy5ijUk5miF4+T/VVyH34=` würde sich folgender HMAC für diesen Aufruf ergeben: `49cb6c6c6a359305a352637f1f404874b9f3358f3fc41447ddbe00`

Daraus ergibt sich der folgende HTTP Aufruf:

```
POST /api/v1/betreuung/gutscheine/GB-123456789-00/vertragregistrieren HTTP/1.1
Host: ds.traegerportal.isbj.verwalt-berlin.de
Date: Tue, 12 Jun 2018 15:04:00 GMT
Authorization: HMAC dienstschnittstelle-demo-
user:49cb6c6c6a359305a352637f1f404874b9f3358f3fc41447ddb00aa2380ce31
<... weitere HTTP Header ...>

<... HTTP Body ...>
```

4.2 Autorisation

Die Autorisation des Nutzers erfolgt gemäß den Berechtigungen, die im BNV-Portal durch den Administrator des Trägers vergeben wurden. Der Nutzer wird anhand des übergebenen Nutzernamens ermittelt. Für die Ausführung von Operationen werden dieselben Berechtigungen eingefordert, die für die entsprechenden Operationen im Trägerportal eingefordert werden.

5 Personalmeldungen, Kindzuschläge, Kitaverzeichniseinträge und Vormerkungen REST API der Dienstschnittstelle anhand des XSD Schema

Für die Anwendungsfälle *Personalmeldungen*, *Kindzuschläge* und *Vormerkungen* wird bereits seit einigen Jahren eine Dienstschnittstelle basierend auf den Technologien *RESTful APIs* und *XML* angeboten. Diese Schnittstellen und ihre spezifischen Ausprägungen werden in den folgenden Abschnitten beschrieben.

5.1 Datenlieferung und Abfrage

Im Folgenden wird das Verhalten der Dienstschnittstelle in Bezug auf Datenlieferungen und Abfragen beschrieben. Des Weiteren werden Details zu den verwendeten Datenformaten erläutert, um eine korrekte Verwendung der Dienstschnittstelle zu ermöglichen.

5.1.1 Datenbestand

Der Datenbestand des Trägerportals kann unter bestimmten Bedingungen nicht immer aktuell sein. Das hängt mit dem Abgleich der Daten aus den BeLa-Anwendungen zusammen, bei dem die Daten aus dem gesicherten Netzwerk mit den Daten aus dem Trägerportal abgeglichen werden. Normalerweise sind die Daten des Trägerportals aufgrund dieses regelmäßigen Datenabgleichs

nicht älter als 24 Stunden, die Daten der endgültigen Datenhaltung wiederum nicht älter als 48 Stunden.

5.1.2 Status- und Detailabfragen

Abfragen auf Daten werden unterteilt in Statusabfragen und Detailabfragen. Detailabfragen beinhalten sämtliche fachliche Daten, wie sie auch im Trägerportal einsehbar sind. Statusabfragen liefern hingegen Metainformationen wie beispielsweise die URL und das Datum einer Veröffentlichung. Statusabfragen liefern erst dann ein Ergebnis bzw. das aktuelle Ergebnis zurück, wenn die Daten in der endgültigen Datenhaltung vorhanden sind. Diese unterscheiden sich je nach Anwendungsfall:

Tabelle 1 – Relevante Datenhaltungsorte für Statusabfragen je nach Anwendungsfall

Anwendungsfall für Statusabfrage	Ort der endgültigen Datenhaltung	Replikation
personalplanung	ISBJ E&D Personal	innerhalb 24h
vormerkliste	ISBJ Vormerkliste	innerhalb 24h
kitaverzeichnis	ISBJ Kitaverzeichnis	innerhalb 24h
kindzuschlag	ISBJ Kita	innerhalb 24h

Details zu den verschiedenen Status sowie Detailantworten können der Schemadefinition unter `antwort_type` entnommen werden.

Für alle Abfragen an die Dienstschnittstelle gilt: Es können nur Daten abgefragt werden, für die der abfragende Nutzer die Rechte hat.

5.1.3 Asynchrone Datenlieferung

Datenlieferungen werden asynchron bearbeitet, d. h., die Lieferung wird angenommen und mit einer Vorgangsnummer (auch Trackingnummer genannt) bestätigt, sofern die XML-Daten dem Schema entsprechen. Erst danach werden diese Daten validiert und ggf. an das Trägerportal weitergeleitet, von wo aus diese zu den Backend-Anwendungen weitergegeben werden. Das Ergebnis der Validierung steht wenige Minuten nach der Lieferung über den Aufruf des Anwendungsfalls `protokoll` zur Verfügung. Durch die Abfrage der Protokollinformationen einer Lieferung mithilfe der Trackingnummer kann die Validität der einzelnen Datensätze und die der gesamten Datenlieferung nachvollzogen werden.

Fachliche Antworten bzw. Statusaktualisierungen finden im Rahmen der normalen Replikationen von Backend-Anwendungen zum Trägerportal statt.

5.1.4 Technische Validierung

Vor jeder fachlichen Validierung wird eine technische Validierung durch die Dienstschnittstelle durchgeführt. Dabei werden z. B. die Schemakonformität, UTF-8-Kodierung der Daten, Korrektheit der Nutzerdaten und Gültigkeit des Nutzerzertifikats geprüft.

Sollte die technische Validierung fehlschlagen, wird die gesamte Lieferung abgelehnt. Eine entsprechende Fehlermeldung wird als Antwort auf den Request gesendet.

5.1.5 Fachliche Validierung und Protokollabfragen

Das Trägerportal führt eine Reihe von fachlichen Validierungen durch, die analog zu den Eingabvalidierungen über die Oberfläche des Trägerportals die fachliche Korrektheit von Datensätzen verifizieren. Sind die Daten valide, werden sie unverändert an die Backend-Anwendungen weitergeleitet.

Für den Fall, dass Datensätze in einer Lieferung fachlich nicht valide sind, werden diese Datensätze nicht bearbeitet. Entsprechende Fehlermeldungen und Status einzelner Datensätze können aus den Protokollabfragen zu Datenlieferungen entnommen werden.

5.1.6 Prüfsummenvalidierung und -berechnung

5.1.6.1 Wichtige Informationen

Die Prüfsummenberechnung kann konfigurierbar ein- und ausgeschaltet werden. Dies liegt im Ermessen des Betreibers des ISBJ Trägerportals. Bitte kontaktieren Sie hierzu den Betreiber.

5.1.6.2 Sonderfall *Personalplanung*

Die Prüfsummenvalidierung und -berechnung ist ein Feature, das für den Anwendungsfall *personalplanung* immer deaktiviert ist (seit Version 1.18.0)! In der Lieferung können Dummywerte eingefügt werden, diese müssen weiterhin dem XSD-Schema entsprechen (siehe Kapitel [Technische Definition](#)). Ein Beispiel ist unter [Prüfsumme für Personalplanung](#) zu finden.

5.1.6.3 Technische Definition

Eine Prüfsumme ist ein durch das MD5-Hash Verfahren verschlüsselter String.

Eine Prüfsumme ist 32 Zeichen lang und besteht aus der Menge der Hexadezimal-Zahlen mit lowercase Buchstaben (Regex: `[a-f0-9]{32}`). Voranstehende Nullen werden aufgefüllt. Die XSD-Schema-Typ-Bezeichnung ist `pruefsumme_type`.

5.1.6.4 Berechnung der Prüfsumme

Jede Datenlieferung enthält zwei Arten von MD5-Prüfsummen: jeweils eine Prüfsumme pro Datensatz und eine Prüfsumme für die gesamte Datenlieferung.

Im XML-Header steht die Prüfsumme der gesamten Datenlieferung. Sie wird über die konkatenierten Prüfsummen der einzelnen Datensätze in der Reihenfolge erstellt, in der sie im XML-Code auftreten.

Im XML-Body befinden sich die Prüfsummen für jeden Datensatz. Um sie zu bilden, werden die Einrichtungsnummer, die Empfänger-ID sowie die Inhalte aus den Fachdaten-Tags konkateniert und mit dem MD5-Hashverfahren verschlüsselt. Bei der Konkatenierung ist die Reihenfolge des Auftretens innerhalb des XML-Codes (von oben nach unten) einzuhalten.

Durch die Prüfsummenvalidierung wird die Datenintegrität bei der Übermittlung verifiziert. Des Weiteren werden diese Prüfsummen verwendet, um Dubletten zu erkennen, so dass diese abgelehnt werden. Hierzu werden:

- die Prüfsumme der Datenlieferung gegen vorhandene Datenlieferungsprüfsummen in der Datenbank überprüft. Existiert dieselbe Datenlieferungsprüfsumme wird die gesamte Datenlieferung abgelehnt.
- die Prüfsummen anhand der gelieferten Daten neu berechnet. Unterscheiden sich die gelieferten Prüfsummen von den berechneten, wird die gesamte Datenlieferung als ungültig abgelehnt.

Sind beide Prüfsummen identisch, wird in der Datenbank überprüft, ob bereits ein Datensatz mit derselben Prüfsumme existiert. In diesem Fall wird der Datensatz mit dem Fehler Dublette erkannt abgelehnt. Andernfalls folgt die fachliche Validierung.

Im folgenden Beispiel wird der XML-Code einer Lieferung an die *vormerkung* mit den dazugehörigen Prüfsummen dargestellt und anschließend die Prüfsummenberechnung erläutert:

```
1 <root xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
2   <header xsi:type="header-anfrage_type">
3     <erstellungsdatum>2024-08-19T15:40:38</erstellungsdatum>
4     <software>
5       <hersteller-name>Nortal AG</hersteller-name>
6       <software-name>ISBJ Dienstschnittstelle</software-name>
7       <software-version>2.0.0</software-version>
8     </software>
9     <pruefsumme>a24ea67a7e15fb678a5d97a62fbd0f66</pruefsumme>
10  </header>
11  <body xsi:type="body_type">
12    <traeger nummer="8368">
13      <einrichtung nummer="10231060">
14        <datensatz lfdnummer="1" xsi:type="datensatz-anfrage_type">
```

```
15     <admin-anfrage>
16         <erstellerid>1001</erstellerid>
17         <erstellungsdatum>2024-08-19T15:40:38</erstellungsdatum>
18         <aenderungsdatum>2024-08-19T15:40:38</aenderungsdatum>
19         <aktion>create</aktion>
20         <pruefsumme>b0e1a22906b9747647455743eac8e5e8</pruefsumme>
21     </admin-anfrage>
22     <fachdaten>
23         <vormerkung>
24             <aufnahme-ab>2026-01-01</aufnahme-ab>
25             <platzumfang>HTB</platzumfang>
26             <kontakt>Familie Mustermann</kontakt>
27             <kind>
28                 <vorname>Max</vorname>
29                 <nachname>Mustermann</nachname>
30                 <geburtsdatum>
31                     <jahr>2023</jahr>
32                     <monat>05</monat>
33                     <tag>15</tag>
34                 </geburtsdatum>
35                 <geschlecht>m</geschlecht>
36                 <adresse>
37                     <strasse>Musterstraße</strasse>
38                     <hausnr>123</hausnr>
39                     <plz>10115</plz>
40                     <ort>Berlin</ort>
41                 </adresse>
42             </kind>
43         </vormerkung>
44     </fachdaten>
45 </datensatz>
46 <datensatz lfdnummer="2" xsi:type="datensatz-anfrage_type">
47     <admin-anfrage>
48         <erstellerid>1002</erstellerid>
49         <empfaengerid>2002</empfaengerid>
50         <erstellungsdatum>2024-08-19T15:40:38</erstellungsdatum>
51         <aenderungsdatum>2024-08-19T15:40:38</aenderungsdatum>
52         <aktion>update</aktion>
53         <pruefsumme>05b812c621e4355f92ca2972b84d4d02</pruefsumme>
54     </admin-anfrage>
55     <fachdaten>
56         <vormerkung>
57             <aufnahme-ab>2026-08-01</aufnahme-ab>
58             <platzumfang>GTB</platzumfang>
59             <kontakt>Familie Schmidt</kontakt>
```

```
60         <kind>
61             <vorname>Anna</vorname>
62             <nachname>Schmidt</nachname>
63             <geburtsdatum>
64                 <jahr>2022</jahr>
65                 <monat>12</monat>
66                 <tag>03</tag>
67             </geburtsdatum>
68             <geschlecht>w</geschlecht>
69             <adresse>
70                 <strasse>Beispielweg</strasse>
71                 <hausnr>45</hausnr>
72                 <plz>10117</plz>
73                 <ort>Berlin</ort>
74             </adresse>
75         </kind>
76     </vormerkung>
77 </fachdaten>
78 </datensatz>
79 <datensatz lfdnummer="3" xsi:type="datensatz-anfrage_type">
80     <admin-anfrage>
81         <erstellerid>1003</erstellerid>
82         <empfaengerid>2003</empfaengerid>
83         <erstellungsdatum>2024-08-19T15:40:38</erstellungsdatum>
84         <aenderungsdatum>2024-08-19T15:40:38</aenderungsdatum>
85         <aktion>delete</aktion>
86         <pruefsumme>bfad3b171b73c19718b8e2b5f706fa68</pruefsumme>
87     </admin-anfrage>
88     <fachdaten/>
89 </datensatz>
90 </einrichtung>
91 </traeger>
92 </body>
93 </root>
```

Dieses Beispiel enthält drei Datensätze vom Anwendungsfall Vormerkung mit den Aktionen `create`, `update` und `delete`. Darin gibt es vier Prüfsummen, welche folgendermaßen gebildet werden.

Die Prüfsummen der einzelnen Datensätze im XML-Body ergeben sich aus den konkatenierten Werten der Einrichtungsnummer, der Empfänger-ID sowie der Fachdaten des jeweiligen Datensatzes, wobei es im `create`-Fall keine Empfänger-ID und im `delete`-Fall keine Fachdaten gibt:

Prüfsumme 1. Datensatz für Aktion: `create`:

b0e1a22906b9747647455743eac8e5e8 = MD5("102310602026-01-01HTBFamilie
↪ MustermannMaxMustermann2023051523mMusterstraße12310115Berlin")

Prüfsumme 2. Datensatz mit Aktion: update:

05b812c621e4355f92ca2972b84d4d02 = MD5("1023106020022026-08-01GTBFamilie
↪ SchmidtAnnaSchmidt2022120322wBeispielweg4510117Berlin")

Prüfsumme 3. Datensatz mit Aktion delete:

bfad3b171b73c19718b8e2b5f706fa68 = MD5("102310602003")

Die Prüfsumme der gesamten Datenlieferung steht im XML-Header und ergibt sich aus dem MD5-Hash der Prüfsummen der einzelnen Datensätze in der Reihenfolge, wie sie im XML-Code auftreten:

Prüfsumme XML-Header:

```
a24ea67a7e15fb678a5d97a62fbd0f66 = MD5(  
    "b0e1a22906b9747647455743eac8e5e8" +  
    "05b812c621e4355f92ca2972b84d4d02" +  
    "bfad3b171b73c19718b8e2b5f706fa68"  
)
```

5.1.6.5 Prüfsumme für Personalplanung

Für den Anwendungsfall *personalplanung* wird die Prüfsumme nicht validiert. Bei der Datenlieferung sind die Prüfsummen frei wählbar gemäß dem XSD-Schema (siehe Kapitel [Technische Definition](#)).

Im folgenden Beispiel wird der XML-Code einer Lieferung an die *personalplanung* mit frei wählbaren Prüfsummen dargestellt:

```
1 <root xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">  
2   <header xsi:type="header-anfrage_type">  
3     <erstellungsdatum>2022-12-06T10:52:40</erstellungsdatum>  
4     <software>  
5       <hersteller-name>Hersteller</hersteller-name>  
6       <software-name>DSS</software-name>  
7       <software-version>1</software-version>  
8     </software>  
9     <pruefsumme>000000000000000000000000f0001000a</pruefsumme>  
10  </header>  
11  <body xsi:type="body_type">  
12    <traeger nummer="0001">  
13      <einrichtung nummer="01020050">  
14        <datensatz lfdnummer="1" xsi:type="datensatz-anfrage_type">  
15          <admin-anfrage>
```

```
16         <erstellerid>0001</erstellerid>
17         <erstellungsdatum>2022-12-06T10:52:40</erstellungsdatum>
18         <aenderungsdatum>2022-12-06T10:52:40</aenderungsdatum>
19         <aktion>create</aktion>
20         <pruefsumme>000000000000000000000000f0001000a</pruefsumme>
21     </admin-anfrage>
22     <fachdaten>
23         <personalplanung>
24             <quereinsteiger>
25                 <name>Lorem-Ipsum</name>
26                 <vorname>Lorem-Ipsum</vorname>
27                 <geburtsdatum>1990-11-25</geburtsdatum>
28                 <geschlecht>w</geschlecht>
29                 <wochenarbeitszeiten>
30                     <anzahlstunden>26.0</anzahlstunden>
31                     <gueltigab>2022-03-01</gueltigab>
32                 </wochenarbeitszeiten>
33                 <gueltigab>2022-03-01</gueltigab>
34                 <quereinsteigerArt>QA</quereinsteigerArt>
35                 <sozialassistent>true</sozialassistent>
36                 <besitztSchulbescheinigung>true</besitztSchulbescheinigung>
37                 <zeitvertrag>>false</zeitvertrag>
38                 <fernstudium>true</fernstudium>
39             </quereinsteiger>
40         </personalplanung>
41     </fachdaten>
42 </datensatz>
43 </einrichtung>
44 </traeger>
45 </body>
46 </root>
```

5.2 Schnittstellenendpunkte

Im Folgenden wird die Endpunkte der REST-Schnittstelle der Dienstschnittstelle genauer beschrieben.

5.2.1 XSD-Schema

Das XSD-Schema ist ein fundamentaler Bestandteil der REST-Schnittstelle. Es beschreibt die XML-Datentypen für Request- & Response-Bodies. Gelieferte Daten werden zunächst gegen das XSD-Schema validiert, bevor ihr Inhalt weiter fachlich geprüft wird. Im Fall invalider Daten wird zusätzlich zum Response-Code auch mit einer beschreibenden Fehlermeldung geantwortet.

Um unnötigen Traffic zu vermeiden, empfiehlt es sich, die zu liefernden Daten bereits vor der Lieferung zu validieren.

Für die aktuelle Version der Dienstschnittstelle wird das XSD-Schema `isbj-dienstschnittstelle-1.24.0.xsd` verwendet.

5.2.2 URL

```
URL           = BasisURL, "/", Anwendungsfall { , Funktion { Parameter } };
BasisURL      = "https://", Host { , ":", Port }, "/", Anwendungsname, "/rest";
Funktion      = "lieferung" | "abfragequote" | "abfragedetail" |
  ↪ "abfrageeinrichtungohnepersonal";
ParameterListe = Parameter { , "&", Parameter };
Parameter     = Wert, "=", Wert;
```

Die Verwendung der Schnittstelle erfolgt stets über eine URL nach zuvor beschriebener Form. Host, Port und Anwendungsname für die Basis-URL können beim Betreiber der Dienstschnittstelle angefragt werden. Die Anwendungsfälle mit ihren möglichen Funktionen und Parametern sind dem Abschnitt REST-WebServices zu entnehmen.

5.2.2.1 Beispiel für eine vollständige URL

```
https://dss-url.de/portal-ws/rest/personalplanung/abfragedetail?traeger=1234&einrichtung=12345678
```

5.2.3 HTTP-Request-Body

Ein Request-Body muss nur bei Datenlieferungen (mit der Funktion `lieferung`) übermittelt werden. Er enthält XML-Code, dessen Struktur über das Schema definiert ist.

Der XML-Code im Request-Body unterteilt sich in zwei Abschnitte, den XML-Header und den XML-Body. Im folgenden Beispiel handelt es sich um einen Datensatz des Anwendungsfalls Personalplanung.

5.2.3.1 XML-Header

Im XML-Header werden die Informationen über das Erstellungsdatum, die verwendete Software und die Gesamtprüfsumme über die Prüfsummen der einzelnen Fachdatensätze übermittelt (siehe Kapitel [Prüfsummenvalidierung und -berechnung](#)). Genauere Informationen können im Schema unter `header_abstract_type` und den abgeleiteten Typen entnommen werden.

```
1 <header xsi:type="header-anfrage_type">
2   <erstellungsdatum>2022-12-06T10:52:40</erstellungsdatum>
```

```
3 <software>
4   <hersteller-name>Hersteller</hersteller-name>
5   <software-name>DSS</software-name>
6   <software-version>1</software-version>
7 </software>
8 <pruefsumme>00000000000000000000000000000000f0001000a</pruefsumme>
9 </header>
```

5.2.3.2 XML-Body

Der XML-Body besitzt die Struktur *traeger* → *einrichtung* → *datensatz*. Pro Lieferung ist ein Träger mit maximal 200 Einrichtung sowie jeweils 1.000 Datensätzen pro Einrichtung erlaubt. Genauere Informationen können dem Schema unter *body_type* und den abgeleiteten Typen entnommen werden.

Ein Datensatz besteht jeweils aus der *admin-anfrage* und den *fachdaten*. Die Schema-Definition ist *datensatz-anfrage_type*.

Die *admin-anfrage* ist für alle Anwendungsfälle identisch (*admin-anfrage_type*). Sie enthält folgende Informationen:

erstellerid ID des Fremdsystems, welche dem Fremdsystemersteller ermöglicht, die von ihm erstellten Datensätze in den ISBJ-Anwendungen wiederzufinden.
Die Ersteller-ID gehört dem Fremdsystem und wird in die ISBJ-Anwendungen nur übernommen, um dem Fremdsystem bei Detail- und Statusabfragen die Zuordnung der Datensätze zu vereinfachen.
Die Prüfung der Ersteller-ID soll nur die Speicherung dieses Feldes in den ISBJ-Anwendungen garantieren und verlangt daher:

- Die Ersteller-ID muss bei Lieferungen vorhanden sein.
- Die Ersteller-ID muss die Form des *id_type* haben (ganze Zahl)
- Das Anlegen von zwei unterschiedlichen Datensätzen mit derselben Ersteller-ID ist erlaubt.

empfaengerid ID des Zieldatensatzes, unter der die Daten in der Backend-Anwendung eindeutig hinterlegt sind.

- Die Empfänger-ID wird bei einer Datenlieferung bei der Aktion *create* generiert und kann über die Funktion *abfragedetails* oder *abfragestatus* abgerufen werden.
- Der Datensatz lässt sich über die vergebene Ersteller-ID identifizieren.
- Für die eindeutige Vergabe der Ersteller-ID ist der Nutzer der Dienstschnittstelle verantwortlich.

erstellungdatum Datum der Erstellung des Datensatzes

aenderungsdatum	Datum der letzten Änderung des Datensatzes
aktion	Aktion zur Übermittlung der Fachdaten: <code>create</code> , <code>update</code> oder <code>delete</code>
pruefsumme	Prüfsumme über den Inhalt der Fachdaten

Die *fachdaten* unterscheiden sich je nach Anwendungsfall. Informationen und Definitionen hierzu können im XSD-Schema unter `fachdaten_type` nachgeschlagen werden.

Ein Beispiel einer kompletten Datenlieferung ist in Abbildung "8 Prüfsumme Beispiel 1 (real)" und Abbildung "9 Prüfsumme Beispiel 2 (frei gewählt)" unter Kapitel [Berechnung der Prüfsumme](#) zu finden.

5.2.4 Anwendungsfälle und Funktionen

Für einen Anwendungsfall kann es bis zu vier verschiedene Funktionen geben. Die Funktion *lieferung* ermöglicht das Einpflegen von Daten mittels einer HTTP-POST-Methode und einem XML-Body mit den einzupflegenden Daten auf Basis des XSD-Schemas. Die Funktionen *abfragestatus* und *abfragedetail* ermöglichen das Abrufen von Informationen über die HTTP-GET-Methode. Wie im Kapitel [Status- und Detailabfragen](#) beschrieben, können mit diesen Funktionen Metainformationen oder Fachdaten abgefragt werden.

Im Einzelnen werden die Anwendungsfälle und Funktionen in Kapitel [REST-WebServices](#) beschrieben.

5.2.4.1 REST-WebServices

Im folgenden werden die Anwendungsfälle genauer beschrieben (Funktion, HTTP-Methode, URL-Parameter, Beschreibung). Falls URL-Parameter für einen Anwendungsfall oder eine Funktion aufgelistet ist, wird ebenfalls die Parameter-Validierung anhand von Referenzen zu XSD-Schema `simpleType` hinterlegt. Für die Funktion *lieferung* wird auf Referenzen zu XSD-Schema `complexType` hingewiesen, welche für die *fachdaten* (`fachdaten_type`) benötigt werden.

5.2.4.2 Smoketest

Methode	GET
Anwendungsfall	<code>smoketest</code>
Funktion	<i>leer</i>
Parameter	<i>keine</i>

Der Smoketest meldet zurück, ob die Dienste aktiv sind und validiert die Benutzerrechte.

5.2.4.3 Kindzuschlag

Methode	GET
Anwendungsfall	kindzuschlag
Funktion	abfragedetail
Parameter	traeger: traegernr_type einrichtung (optional): einrichtungnr_type rechtsgrundlage (optional): rechtsgrundlage_type_enum kartennr (optional): kartenummer_type

Liefert BuT-Kindzuschläge zur abgefragten Einrichtung zurück. Die Ergebnismenge kann mit den optionalen Parametern *einrichtung*, *rechtsgrundlage* und *kartennr* (Berlin Passnummer) weiter eingeschränkt werden.

5.2.4.4 Freiplatzmeldung

(i): Die Funktion *freiplatzmeldung* ist ab der Version 2.0 nicht mehr verfügbar. Die Endpunkte werden aus Kompatibilitätsgründen weiterhin angeboten, liefern aber keine fachlichen Daten sondern einen Fehlercode, der auf diesen Umstand hinweist.

Bei Fragen wenden Sie sich bitte an den ISBJ Träger-Service.

5.2.4.5 Kitaverzeichnis

Methode	POST
Anwendungsfall	kitaverzeichnis
Funktion	lieferung
Parameter	<i>keine</i>

Die Datenlieferung im Request-Body wird hochgeladen.

Methode	GET
Anwendungsfall	kitaverzeichnis
Funktion	abfragestatus
Parameter	traeger: traegernr_type einrichtung (optional): einrichtungnr_type

Die Abfrage liefert den Status von Kitaverzeichniseinträgen für einen Träger bzw. eine Einrichtung eines Trägers zurück. Der Parameter *einrichtung* ist optional.

Methode	GET
Anwendungsfall	kitaverzeichnis
Funktion	abfragedetail

Parameter traeger: traegernr_type
einrichtung: einrichtungnr_type
bild (optional): xs:boolean

Die Kitaverzeichnisdaten zur abgefragten Einrichtung werden zurückgeliefert. Der Parameter bild ist optional und erlaubt die Werte true und false. true liefert die Daten mit Bild, sofern eines vorhanden ist. Default = false.

5.2.4.6 Personalplanung

Methode POST
Anwendungsfall personalplanung
Funktion lieferung
Parameter keine

Die Datenlieferung im Request-Body wird hochgeladen.

Methode GET
Anwendungsfall personalplanung
Funktion abfragedetail
Parameter traeger: traegernr_type
einrichtung: einrichtungnr_type

Eine Liste der Personalmeldungen für die abgefragte Einrichtung wird zurückgeliefert.

Methode GET
Anwendungsfall personalplanung
Funktion abfragequote
Parameter traeger: traegernr_type
einrichtung: einrichtungnr_type

Eine Liste der Personalquoten und der Personalsituation der abgefragten Einrichtung wird zurückgeliefert.

Methode GET
Anwendungsfall personalplanung
Funktion abfrageeinrichtungohnepersonal
Parameter traeger: traegernr_type

Eine Liste der Einrichtungen mit unzureichender Personalsituation wird zurückgeliefert.

5.2.4.7 Stellenangebot

(i): Die Funktion `stellenangebot` ist ab der Version 2.0 nicht mehr verfügbar. Die Endpunkte werden aus Kompatibilitätsgründen weiterhin angeboten, liefern aber keine fachlichen Daten sondern einen Fehlercode, der auf diesen Umstand hinweist.

Bei Fragen wenden Sie sich bitte an den ISBJ Träger-Service.

5.2.4.8 Vertrag

Methode	GET
Anwendungsfall	vertrag
Funktion	abfragedetail
Parameter	traeger: traegernr_type einrichtung: einrichtungnr_type

Liefert Vertragsinformationen zu einem Träger zurück.

5.2.4.9 Vormerkung

Methode	POST
Anwendungsfall	vormerkung
Funktion	lieferung
Parameter	<i>keine</i>

Die Datenlieferung im Request-Body wird hochgeladen.

Methode	GET
Anwendungsfall	vormerkung
Funktion	abfragestatus
Parameter	traeger: traegernr_type einrichtung (optional): einrichtungnr_type

Die Abfrage liefert den Status von Vormerkungen für einen Träger bzw. eine Einrichtung zurück. Der Parameter `einrichtung` ist optional.

Methode	GET
Anwendungsfall	vormerkung
Funktion	abfragedetail
Parameter	traeger: traegernr_type einrichtung: einrichtungnr_type

Eine Liste der Vormerkungen für die abgefragte Einrichtung wird zurückgeliefert. Grundsätzlich sind nur Kita-Träger und Kita-Einrichtungen als Aufruf-Parameter erlaubt (d. h. mit einer Träger-ID aus 4 Ziffern und einer Einrichtungs-ID aus 8 Ziffern).

Bei der Funktion *lieferung* enthält jeder Datensatz eine *create*-, *update*- oder *delete*-Aktion. Ein Datensatz wird mit *create* neu angelegt, mit *update* bearbeitet und mit *delete* gelöscht. Im letzten Fall stehen keine Fachdaten im XML-Body zur Verfügung.

Definition des Geburtsdatums bei Vormerkung:

Für die erfolgreiche Verarbeitung der Datenlieferung ist es zwingend erforderlich, für das Geburtsdatum neben dem Geburtsjahr auch Tag und Monat anzugeben, da alle ISBJ-Anwendungen bisher nur mit vollständigem Geburtsdatum arbeiten können.

5.2.4.10 Aktionen für die Funktion *lieferung*

Die folgenden Aktionen sind für die verschiedenen Anwendungsfälle verfügbar:

Personalplanung und Vormerkung:

<code>create</code>	Datensatz wird neu erstellt
<code>update</code>	Datensatz wird bearbeitet
<code>delete</code>	Datensatz wird gelöscht (keine Fachdaten im XML-Body erforderlich)

Kitaverzeichnis:

<code>update</code>	Datensatz wird bearbeitet (einzige verfügbare Aktion)
---------------------	---

(i): Die Anwendungsfälle `freiplatzmeldung` und `stellenangebot` sind ab Version 2.0 nicht mehr verfügbar.

5.2.4.11 Identifikation einer Personalmeldung bei *personalplanung*/*lieferung*

Bei der Datenlieferung einer Person bei dem Anwendungsfall *personalplanung* ist es wichtig, dass eine Person eindeutig identifizierbar ist. Die Identifizierung von Personal findet auf Einrichtungsebene statt, d. h. eine Person wird im Backend mit der Zuordnung zu ihrer Einrichtung lokalisiert. Vielmehr ist ein Personaldatensatz als ein Person-Einrichtung-Vertrag zu betrachten. Prinzipiell kann eine einzelne Person in mehreren Einrichtungen eingepflegt sein. Es liegt in der Verantwortung der Träger, die Arbeitszeiten und Gültigkeiten bei Personalwechsel von Einrichtung A nach Einrichtung B entsprechend zu pflegen. Das bedeutet, dass der Träger hier zwei Datensätze für eine natürliche Person zu pflegen hat.

Bei der Implementierung der Dienstschnittstelle vor v1.18.0 wurde zur Identifizierung einer Person die *laufendeNr* aus dem XSD-Schema-Typ `personstamm_abstract_type` verwendet.

Seit v1.18.0 wurde dies geändert hin zu einer Identifizierung mittels Kombination aus *name*, *vorname*, *geburtsdatum* aus dem XSD-Schema-Typ `personstamm_abstract_type`. Das Feld *laufendeNr* ist nun kein Pflichtfeld mehr und wird vom Backend bei der Datenlieferung ignoriert.

Seit v1.20.0 ist nun die Identifikation mittels *personalnummer* neben der Kombination aus *name*, *vorname*, *geburtsdatum* aus dem XSD-Schema-Typ *personstamm_abstract_type* ebenfalls möglich. Somit können nun *name*, *vorname*, *geburtsdatum* für einen bestehenden Datensatz ebenfalls über die Dienstschnittstelle angepasst werden, sofern eine *personalnummer* vergeben ist.

5.2.4.12 Setzen der Personalnummer bei Personalplanung/Lieferung

Seit v1.20.0 sind die *personalnummer* & *personalnummerNeu* Teil des XSD-Schema-Typs *personstamm_abstract_type*. Beide Elemente sind keine Pflichtfelder.

Initial haben alle Personaldatensätze keine *personalnummer*. Die Träger können diese frei wählen.

Zum Setzen der *personalnummer* wird die *personalnummerNeu* verwendet. Wird bei der *personalplanung/lieferung* für einen Datensatz die *personalnummerNeu* mitgeliefert, wird die *personalnummer* entsprechend mit dem Wert der *personalnummerNeu* gesetzt, sofern diese valide ist und auf Einrichtungsebene noch nicht vergeben ist.

WICHTIG: Wird die *personalnummerNeu* mit einem leeren String verschickt, wird die *personalnummer* für diesen Datensatz auf null gesetzt. Damit ist es möglich, eine *personalnummer* jederzeit für einen Datensatz zurückzusetzen.

Folgende Regeln gelten bei der Validierung der *personalnummer*:

- Erlaubt sind alle Unicode-Zeichen außer C0- & C1 Control-Characters
- Eine *personalnummer* muss einzigartig auf Einrichtungsebene sein. Dies hat Auswirkungen auf die Identifizierung eines Personaldatensatzes (siehe Kapitel "[Identifikation einer Personalmeldung bei Personalplanung/Lieferung](#)").
- Auf Trägerebene darf eine *personalnummer* mehrfach vergeben werden. Hierbei gilt die bekannte fachliche Validierung, dass Personalmeldungen mit der gleichen *personalnummer* akkumuliert nicht mehr als eine bestimmte Anzahl an Wochenarbeitsstunden überschreiten dürfen.
- Die *personalnummerNeu* sollte nur bei der Datenlieferung verschickt werden, wenn die *personalnummer* neu gesetzt werden möchte.

5.2.4.13 Protokoll

Methode	GET
Anwendungsfall	protokoll
Funktion	Abfrage von Statusinformationen einzelner Datensätze nach einer Lieferung
Parameter	trackingnr: Vorgangsnummer (Trackingnummer) der Lieferung

Für jede technisch fehlerfreie Lieferung wird durch die Dienstschnittstelle eine Vorgangsnummer (Trackingnummer) erzeugt. Die Abfrage liefert die einzelnen Status der Lieferung zurück. In der Protokollantwort lässt sich über das Attribut *lfdnummer* im Tag <datensatz> der Datensatz aus der Lieferung identifizieren.

Status einzelner Datensätze:

- OK** Der Datensatz wurde von der Dienstschnittstelle angenommen und hat die fachliche Validierung im Trägerportal bestanden. Dieser Datensatz wird an die zuständige Fachanwendung in ISBJ weitergeleitet.
- WARNING** Der Datensatz weist ein Problem auf, das eine Verarbeitung aber nicht verhindert (z. B. ein Update ohne eine Änderung). Dieser Datensatz wird ggf. an die zuständige Fachanwendung in ISBJ weitergeleitet. Details zu dem Problem werden in den Fehlermeldungen aufgeführt.
- ERROR** Der Datensatz ist fehlerhaft und kann nicht verarbeitet werden. Details werden in den Fehlermeldungen aufgeführt.

Status einer Lieferung:

Für die gesamte Lieferung steht ebenfalls eine Statusanzeige zur Verfügung. Diese dient der Entlastung externer Systeme, da die einzelnen Datensätze im OK-Fall nicht geprüft werden müssen.

- OK** Alle Datensätze in der Datenlieferung haben den Status OK.
- WARNING** Nicht alle Datensätze in der Datenlieferung haben den Status OK. Hier muss im Einzelfall geprüft werden, welche Datensätze fehlerhaft sind.
- ERROR** Die gesamte Datenlieferung ist fehlerhaft. Alle Datensätze in der Datenlieferung sind fehlerhaft.

5.2.4.14 Response-Codes

- 200 OK** Die Daten wurden erfolgreich übermittelt bzw. die Request wurde verstanden und ohne Fehler durchgeführt.
Achtung: HTTP-200 wird auch für Ergebnisse verwendet, die fachliche Probleme haben (z. B. einzelne Datensätze werden abgelehnt, weil diese Duplikate sind).
- 400 Bad Request** Die Clientanwendung hat einen Fehler gemacht, z. B. falsche HTTP-Methode, XML-Daten die nicht schemakonform sind usw.
- 401 Unauthorized** Das Zertifikat ist nicht für diesen Anwender. Benutzername oder Kennwort sind falsch oder der Basic-Authentication Header ist nicht gesetzt bzw. nicht lesbar.
- 403 Forbidden** Keine Berechtigung für diese Aktion, obwohl die Anmeldung erfolgreich war. Z. B. Datenabfrage für einen Träger, für den der Benutzer keine Rechte hat.

404 Not Found Falsche URL

500 Internal Server Error In der Dienstschnittstelle ist ein unbehandelter Fehler aufgetreten. Dieser Fehler muss zusammen mit der Uhrzeit, dem Benutzernamen und weiteren sinnvollen Details an den entsprechenden Helpdesk oder dem Träger-Service gemeldet werden.

503 Service Unavailable In der Dienstschnittstelle bzw. dem Trägerportal stehen Wartungsarbeiten an und die Anwendung ist nicht verfügbar.

5.2.5 Eigenschaften der Dienstschnittstelle

5.2.5.1 Datenlieferung

Ab der Version 1.3.0 der Dienstschnittstelle ist es nicht mehr vorgesehen, dass vollständige Lieferungen des extern gehaltenen Datenbestandes an die ISBJ-Dienstschnittstelle vorgenommen werden. Es ist zwar weiterhin möglich, vollständige Lieferungen zu tätigen, um alle beteiligten Systeme zu entlasten, wird jedoch davon abgeraten. Stattdessen wird empfohlen, nur die geänderten Datensätze zu liefern.

5.2.5.2 Reaktionszeiten auf verschiedene Aktionen bei Lieferungen

Grundsätzlich können Datensätze, die über die Dienstschnittstelle angelegt oder bearbeitet werden, erst dann erneut geändert werden, nachdem sie von der dem Anwendungsfall zugehörigen Backendkomponente freigegeben wurden. Dies kann je nach angegebener Aktion und Größe der gelieferten Datei sowie der allgemeinen Last auf den ISBJ-Systemen mehrere Minuten dauern. Die Aktion *delete* stellt hierbei einen Sonderfall dar.

Reaktionsverhalten auf Aktionen:

create & update Die Datensätze sollten innerhalb weniger Minuten über die Dienstschnittstelle auslesbar und erneut bearbeitbar sein.

delete Gelöschte Datensätze werden im Trägerportal als "in Bearbeitung" markiert und einmal in 24 Stunden im Rahmen des Datenabgleichs zwischen Trägerportal und Backendkomponente aus dem Datenbestand des Trägerportals entfernt.

5.2.5.3 Zusätzliche Informationen zum XSD-Schema

Mit dem Schema werden aus folgenden Gründen Datentypen definiert und Einschränkungen vorgenommen:

Datumstypen:

Es existieren zwei Standarddatumstypen:

`xs:date` wird in der XML-Datei repräsentiert durch `<datum>yyyy-MM-dd</datum>`
`xs:dateTime` wird in der XML-Datei repräsentiert durch `<datum>yyyy-MM-ddThh:mm:ss</datum>`

Diese Typen wurden gewählt, um die Kommunikation aller am Fachverfahren beteiligten Komponenten sicherzustellen.

Einschränkungen der Datensätze innerhalb einer Lieferung:

Derzeit ist eine Datenlieferung durch folgende Kardinalitäten begrenzt:

- 1 Träger pro Lieferung
- 200 Einrichtungen pro Träger
- 1000 Datensätze pro Einrichtung

Diese Einschränkungen wurden unter Berücksichtigung der derzeit in der Praxis vorkommenden Anwendungsfälle festgelegt. Sollten diese Begrenzungen überschritten werden, muss der Träger-Service über diesen Anwendungsfall informiert werden. Hierbei sei auf Kapitel [Datenlieferung](#) verwiesen. Vollständige Updates sind nicht mehr vorgesehen und können ggf. in mehrere kleinere Lieferungen unterteilt werden.

Einschränkung der Wertebereiche:

Die Dienstschnittstelle akzeptiert nur eingeschränkte Wertebereiche, die durch die Längendefinitionen der Datenbankschemata in anderen Komponenten vorgegeben sind. Anwendungsfälle, die diese Einschränkungen überschreiten, sind dem Träger-Service mitzuteilen.

5.2.5.4 Fehlermeldungen und deren Weitergabe

Grundsätzlich muss entschieden werden, ob der Fehler technische oder fachliche Ursachen hat. Fehlermeldungen, die darauf hindeuten, dass die Daten nicht schemakonform sind, kann kein Anwender beheben. Diese und weitere technische Fehler müssen an die Entwickler der externen Anwendung weitergeleitet werden. Fachliche Fehlermeldungen sollten deshalb dem Anwender im externen System dargestellt werden, sodass die Daten korrigiert werden können.

Es wird dringend empfohlen, Fehlermeldungstexte aus der Dienstschnittstelle nicht vor der Darstellung zu verändern, andernfalls wird eine Überprüfung seitens des Träger-Services oder der Nortal AG erschwert.

Sollte eine Anfrage an die Dienstschnittstelle gerichtet werden, bei der im Header ein falscher Benutzername oder ein falsches Passwort übergeben wird, erstellt die Dienstschnittstelle zwei unterschiedliche Fehlermeldungen. Dies ist ein gewünschtes Verhalten. Sofern die Fehlermeldungen im externen System gleichgestellt werden sollen, können über den folgenden regulären Ausdruck beide Fehlermeldungen ausgewertet werden:

```
^(Der Benutzer).( oder das Passwort).( falsch\.)$
```

6 Vertragsverwaltung RESTful API der Dienstschnittstelle

Die Dienstschnittstelle stellt eine RESTful-API bereit, über die Operationen der Vertragsverwaltung durchgeführt werden können. Diese Operationen werden im folgenden beschrieben.

6.1 Authentifikation und Autorisation

Die API unterstützt die folgenden Verfahren zur Authentifikation.

6.1.1 HMAC

Für weitere Details zur HMAC-Authentifizierung siehe die Entwicklerdokumentation Kapitel 4.

6.2 API Endpunkte und Operationen

6.3 Einen Gutschein abrufen

GET /api/v1/betreuung/gutscheine/{gutscheinnummer}

Liefert die Details eines Gutscheins, welcher über die Gutscheinnummer identifiziert wird.

6.3.1 Anfrageparameter

Name	In	Typ	Erforderlich	Beschreibung
gutscheinnummer	path	string	ja	Die Gutscheinnummer des abzurufenden Gutscheins.

6.3.2 Antworten

HTTP Status-Code	Status-Text	Beschreibung
200	OK	OK

HTTP Status-Code	Status-Text	Beschreibung
401	Unauthorized	Der Header für die Authentisierung ist nicht vorhanden oder konnte nicht interpretiert werden.
403	Forbidden	Der Benutzer konnte nicht im System gefunden werden oder besitzt nicht die erforderlichen Rechte für das Abrufen von Daten von der Dienstschnittstelle.
404	Not Found	Der angefragte Gutschein mit der entsprechenden Gutscheinnummer wurde nicht gefunden.
500	Internal Server Error	Bei der Verarbeitung ist es zu einem unvorhergesehenen Problem gekommen. Die Verarbeitung der Nachricht hat nicht stattgefunden.

6.3.2.1 Beispielantwort (erfolgreich)

```
{
  "antrag": {
    "antragsnummer": "string",
    "url": "string"
  },
  "kindId": 0,
  "ausstellungsdatum": "2019-08-24",
  "gutscheinnummer": "string",
  "gutscheintyp": "string",
  "gueltigkeitszeitraumBeginn": "2019-08-24",
  "gueltigkeitszeitraumEnde": "2019-08-24",
  "betreuungsumfang": "H",
  "zustaendigkeitBezirksamt": "string",
  "zustaendigkeitBearbeiterAnrede": "string",
  "zustaendigkeitBearbeiterNachname": "string",
  "zustaendigkeitBearbeiterEmail": "string",
```

```
"zustaendigkeitBearbeiterTelefon": "string",
"einloesbar": true,
"einloesbarkeitszeitraeume": [
  {
    "gueltigAb": "2019-08-24",
    "gueltigBis": "2019-08-24"
  }
],
"elternbeteiligungen": [
  {
    "gueltigAb": "2019-08-24",
    "typ": "K",
    "kostenBetreuung": 0,
    "kostenEssen": 0
  }
],
"zuschlagsberechtigungen": [
  {
    "typ": "QM",
    "gueltigVon": "2019-08-24",
    "gueltigBis": "2019-08-24"
  }
]
}
```

6.4 Verträge eines Gutscheins abrufen

GET /api/v1/betreuung/gutscheine/{gutscheinnummer}/vertraege

Liefert eine paginierte Liste aller Verträge eines Gutscheins.

6.4.1 Anfrageparameter

Name	In	Typ	Erforderlich	Beschreibung
gutscheinnum- mer	path	string	ja	Die Gutschein- nummer, deren Verträge abgerufen werden sollen.

Name	In	Typ	Erforderlich	Beschreibung
aktiv	query	boolean	nein	Wird diese Flagge gesetzt, werden nur die nicht stornierten Verträge behandelt.
max	query	integer	nein	Die maximale Anzahl an Einträgen, die zurückgegeben werden sollen.
start	query	integer(int64)	nein	Der Index des ersten Eintrags, der zurückgegeben werden soll.

6.4.2 Antworten

HTTP Status-Code	Status-Text	Beschreibung
200	OK	Der Auftrag wurde erfolgreich entgegengenommen und verarbeitet. Eine strukturierte Antwort mit den angeforderten Daten wird an den Client zurückgegeben.
400	Bad Request	Einer oder mehrere Parameter sind falsch angegeben worden.
401	Unauthorized	Der Header für die Authentisierung ist nicht vorhanden oder konnte nicht interpretiert werden.

HTTP Status-Code	Status-Text	Beschreibung
403	Forbidden	Der Benutzer konnte nicht im System gefunden werden oder besitzt nicht die erforderlichen Rechte für das Abrufen von Daten von der Dienstschnittstelle.
404	Not Found	Der angefragte Gutschein mit der entsprechenden Gutscheinnummer wurde nicht gefunden.
500	Internal Server Error	Bei der Verarbeitung ist es zu einem unvorhergesehenen Problem gekommen. Die Verarbeitung der Nachricht hat nicht stattgefunden.

6.4.2.1 Beispielantwort (erfolgreich)

```
{
  "metainformationen": {
    "paginierung": {
      "start": 0,
      "max": 0,
      "eintraegeGesamt": 0
    }
  },
  "eintraege": [
    "string"
  ]
}
```

6.5 Vertrag registrieren

POST /api/v1/betreuung/gutscheine/{gutscheinnummer}/vertragRegistrieren

Registriert einen neuen Vertrag für einen Gutschein.

6.5.1 Anfrageparameter

Name	In	Typ	Erforderlich	Beschreibung
gutscheinum- mer	path	string	ja	Die Nummer des Gutscheins, für den der Vertrag registriert werden soll.

6.5.2 Anfragedaten (request body parameter)

Es werden die Informationen zum Vertrag übertragen. Dabei müssen die folgenden Validierungsregeln beachtet werden:

Nachname des Kindes:

- Pflichtfeld (max. 255 Zeichen)
- Voraussetzung für die Identifizierung des Kindes in Kita-Familienverwaltung

Vorname des Kindes:

- Pflichtfeld (max. 255 Zeichen)
- Voraussetzung für die Identifizierung des Kindes in Kita-Familienverwaltung

Geburtsdatum des Kindes:

- Pflichtfeld (Datum)
- Voraussetzung für die Identifizierung des Kindes in Kita-Familienverwaltung

Vertragsabschluss:

- Pflichtfeld (Datum)
- Vertragsabschluss > Vertragsbeginn: "Bitte geben Sie ein Datum ein, das kleiner oder gleich ist."
- Vertragsabschluss liegt mehr als 9 Monate vor dem Beginn der Einlösbarkeit des Gutscheins: "Der Vertragsabschluss muss frühestens auf den fallen."

Vertragsbeginn:

- Pflichtfeld (Datum)

- Zeit zwischen Gültig ab des Gutscheins und Vertragsbeginn > 7 Monate bei Basisgutschein mit Betreuungsumfang Ganztags oder Erweitert: "Der Vertragsbeginn darf maximal sieben Monate nach dem Gültigkeitsbeginn des Gutscheins liegen oder nach dem Ende des vorherigen Vertrags."
- Vertragslaufzeit > Einlösbarkeit des Gutscheins: "Der Betreuungszeitraum muss vollständig innerhalb der Einlösbarkeit des Gutscheins liegen: bis ."
- Vertragsbeginn außerhalb Öffnungszeitraum der Einrichtung: "Der Vertragsbeginn muss innerhalb des Öffnungszeitraums der gewählten Einrichtung liegen."
- Registrierung nach 31.3. bei Vertragsbeginn im Vorjahr: "Verträge, die teilweise im vergangenen Jahr liegen, können nur bis zum 31. März des Folgejahres registriert werden."
- Vertragsbeginn im Vor-Vorjahr: "Die Vertragslaufzeit im vorvergangenen Jahr kann nicht mehr registriert werden."
- Überschneidung mit bestehendem Vertrag: "Der Betreuungszeitraum überschneidet sich mit demjenigen des Vertrags <(Vertragsbeginn * Vertragsende)>."

Vertragsende:

- Pflichtfeld (Datum)
- Vertragsbeginn > Vertragsende: "Bitte geben Sie ein Datum größer als an."
- Vertragslaufzeit > Gültigkeit des Gutscheins: "Der Betreuungszeitraum muss vollständig innerhalb der Einlösbarkeit des Gutscheins liegen: bis ."
- Nur Basisgutschein: Vertragsende < Monatsletzter des Vertragsbeginns: "Das Vertragsende muss mindestens auf den Letzten des Monats fallen, in dem der Vertrag beginnt."
- Vertragsende außerhalb Öffnungszeitraum der Einrichtung: "Das Vertragsende muss innerhalb des Öffnungszeitraums der gewählten Einrichtung liegen."
- Überschneidung mit bestehendem Vertrag: "Der Betreuungszeitraum überschneidet sich mit demjenigen des Vertrags <(Vertragsbeginn * Vertragsende)>."

Betreuungsumfang:

- Pflichtfeld (Auswahl)
- Kitavertrag auf Basisgutschein: Alle Betreuungsumfänge verfügbar, die geringer als der bisherige sind
 - halbtags (mindestens 4 bis höchstens 5 Stunden)
 - teilzeit (über 5 bis höchstens 7 Stunden)
 - ganztags (über 7 bis höchstens 9 Stunden)
 - erweitert (über 9 Stunden)
- Kitavertrag auf Sprachfördergutschein: Umfang auf 7 Stunden täglich festgesetzt
 - Sprachförderung

Einrichtung:

- Pflichtfeld (Auswahl)
- Nur Einrichtungen mit Berechtigung "Vertragsregistrierung" verfügbar
- Tagesdatum vor 31.3.: Nur Einrichtungen mit Schließdatum im Vorjahr oder später
- Tagesdatum nach 31.3.: Nur Einrichtungen mit Schließdatum im aktuellen Jahr oder später
- Basisgutschein: Nur Einrichtungen vom Typ Kita
- Sprachfördergutschein: Nur Einrichtungen mit Sprachförderangebot

Essen:

- Boolean-Wert
- Nur bei Betreuungsumfang "halbtags" editierbar
- Bei Betreuungsumfang > halbtags: Immer aktiv, nicht editierbar

```
{
  "kindNachname": "string",
  "kindVorname": "string",
  "kindGeburtsdatum": "2020-01-01",
  "vertragsabschluss": "2019-07-31",
  "vertragsbeginn": "2019-07-31",
  "vertragsende": "2019-07-31",
  "betreuungsumfang": "HALBTAGS",
  "mitEssen": true,
  "einrichtungsnummer": "string"
}
```

Die Anfragedaten müssen dem Typ `vertragRegistrieren` entsprechen.

6.5.3 Antworten

HTTP Status-Code	Status-Text	Beschreibung
200	OK	Der Auftrag wurde erfolgreich entgegengenommen und verarbeitet.
400	Bad Request	Einer oder mehrere Parameter sind falsch angegeben worden.
401	Unauthorized	Der Header für die Authentisierung ist nicht vorhanden oder konnte nicht interpretiert werden.

HTTP Status-Code	Status-Text	Beschreibung
403	Forbidden	Der Benutzer konnte nicht im System gefunden werden oder besitzt nicht die erforderlichen Rechte für das Abrufen von Daten von der Dienstschnittstelle.
404	Not Found	Der angefragte Gutschein mit der entsprechenden Gutscheinnummer wurde nicht gefunden.
409	Conflict	Der Vertrag kann nicht registriert werden. Bitte prüfen Sie, ob der Gutschein, auf den Sie den Vertrag registrieren möchten, ein Basis- oder Sprachförderungsgutschein ist und ob der Gutschein einlösbar ist.
423	Locked	Der Datensatz wird derzeit von einer anderen Anwendung bearbeitet und ist daher für die Bearbeitung gesperrt.
500	Internal Server Error	Bei der Verarbeitung ist es zu einem unvorhergesehenen Problem gekommen. Die Verarbeitung der Nachricht hat nicht stattgefunden.

6.5.3.1 Beispielantwort (erfolgreich)

```
{  
  "postingnummer": "PortalWs-2026042105225017-1"  
}
```

6.6 Verträge abrufen

GET /api/v1/betreuung/vertraege

Liefert eine paginierte Liste aller Verträge für welche der aktuelle Nutzer das Vertrags-Änderungs-Recht hat. Bei Selektion durch einen Benutzer, der Zugriff auf alle Einrichtungen hat (z. B. Imports), werden nur die Berlin- und Brandenburgverträge zurückgeliefert.

6.6.1 Anfrageparameter

Name	In	Typ	Erforderlich	Beschreibung
aktiv	query	boolean	nein	Wird diese Flagge gesetzt, werden nur die nicht stornierten Verträge behandelt.
datumAb	query	string	nein	Ein Datum ab dem die Verträge gelesen werden sollen (Vertragsende bzw. Finanzierungszeitraum (eines der beiden Daten muss im Zeitraum liegen))

Name	In	Typ	Erforderlich	Beschreibung
datumBis	query	string	nein	Ein Datum bis zu dem die Verträge gelesen werden sollen (Vertragsende bzw. Finanzierungszeitraum (eines der beiden Daten muss im Zeitraum liegen))
max	query	integer	nein	Die maximale Anzahl an Einträgen, die zurückgegeben werden sollen.
start	query	integer(int64)	nein	Der Index des ersten Eintrags, der zurückgegeben werden soll.

6.6.2 Antworten

HTTP Status-Code	Status-Text	Beschreibung
200	OK	Der Auftrag wurde erfolgreich entgegengenommen und verarbeitet. Eine strukturierte Antwort mit den angeforderten Daten wird an den Client zurückgegeben.
400	Bad Request	Einer oder mehrere Parameter sind falsch angegeben worden.

HTTP Status-Code	Status-Text	Beschreibung
401	Unauthorized	Der Header für die Authentisierung ist nicht vorhanden oder konnte nicht interpretiert werden.
403	Forbidden	Der Benutzer konnte nicht im System gefunden werden oder besitzt nicht die erforderlichen Rechte für das Abrufen von Daten von der Dienstschnittstelle.
500	Internal Server Error	Bei der Verarbeitung ist es zu einem unvorhergesehenen Problem gekommen. Die Verarbeitung der Nachricht hat nicht stattgefunden.

6.6.2.1 Beispielantwort (erfolgreich)

```
{
  "metainformationen": {
    "paginierung": {
      "start": 0,
      "max": 0,
      "eintraegeGesamt": 0
    }
  },
  "eintraege": [
    "string"
  ]
}
```

6.7 Einen Vertrag abrufen

GET /api/v1/betreuung/vertraege/{vertragsnummer}

Liefert die Dateien eines Vertrages, welcher über die Vertragsnummer identifiziert wird.

6.7.1 Anfrageparameter

Name	In	Typ	Erforderlich	Beschreibung
vertragsnummer	path	string	ja	

6.7.2 Antworten

HTTP Status-Code	Status-Text	Beschreibung
200	OK	Der Auftrag wurde erfolgreich entgegengenommen und verarbeitet. Eine strukturierte Antwort mit den angeforderten Daten wird an den Client zurückgegeben.
400	Bad Request	Einer oder mehrere Parameter sind falsch angegeben worden.
401	Unauthorized	Der Header für die Authentisierung ist nicht vorhanden oder konnte nicht interpretiert werden.
403	Forbidden	Der Benutzer konnte nicht im System gefunden werden oder besitzt nicht die erforderlichen Rechte für das Abrufen von Daten von der Dienstschnittstelle.
404	Not Found	Der angefragte Vertrag mit der entsprechenden Vertragsnummer wurde nicht gefunden.

HTTP Status-Code	Status-Text	Beschreibung
500	Internal Server Error	Bei der Verarbeitung ist es zu einem unvorhergesehenen Problem gekommen. Die Verarbeitung der Nachricht hat nicht stattgefunden.

6.7.2.1 Beispielantwort (erfolgreich)

```
{
  "vertragsnummer": "string",
  "registrierungDatum": "2020-01-01",
  "vertragsabschluss": "2020-01-01",
  "laufzeitBeginn": "2020-01-01",
  "laufzeitEnde": "2020-01-01",
  "finanzierungszeitraumBeginn": "2020-01-01",
  "finanzierungszeitraumEnde": "2020-01-01",
  "betreuungsumfang": "string",
  "einrichtung": {
    "nummer": "string",
    "name": "string",
    "adresse": {
      "strasse": "string",
      "postleitzahl": "string",
      "ort": "string"
    },
    "telefon": "string"
  },
  "saldo": 0,
  "zustaendigkeitBezirksamt": "string",
  "zustaendigkeitBearbeiterAnrede": "string",
  "zustaendigkeitBearbeiterNachname": "string",
  "zustaendigkeitBearbeiterEmail": "string",
  "zustaendigkeitBearbeiterTelefon": "string",
  "storniert": true,
  "gutschein": "string",
  "voraussichtlicheVertragsenden": [
    {
      "voraussichtlichesVertragsende": "2019-07-31",
      "aenderungsgrund": "SCHULRUECKSTELLUNG"
    }
  ],
}
```

```
"mitEssen": true  
}
```

6.8 Betreuungsumfang erhöhen

POST /api/v1/betreuung/vertraege/{vertragsnummer}/betreuungsumfangErhoehen

Initiiert die Erhöhung des Betreuungsumfangs für einen Vertrag.

6.8.1 Anfrageparameter

Name	In	Typ	Erforderlich	Beschreibung
vertragsnummer	path	string	ja	

6.8.2 Anfragedaten (request body parameter)

Es werden die Informationen zum Vertrag übertragen. Dabei müssen die folgenden Validierungsregeln beachtet werden:

Gutscheinnummer:

- Pflichtfeld (max. 17 Zeichen)
- Format: GB-
- Gutschein existiert nicht: "Der Gutschein existiert nicht, bitte geben Sie eine andere Gutscheinnummer ein."
- Gutschein ist storniert: "Der Gutschein existiert nicht, bitte geben Sie eine andere Gutscheinnummer ein."
- Ausstellungsdatum neuer Gutschein < Ausstellungsdatum aktueller Gutschein: "Der neue Gutschein muss nach dem aktuellen Gutschein ausgestellt worden sein."
- Zuordnung des Gutscheins zu einem anderen Kind: "Gutschein muss demselben Kind zugeordnet sein"
- Falscher Gutscheinotyp: "Bitte geben Sie den Wert im folgenden Format an: GB-."
- Zuordnung des Gutscheins zu aktuellem Vertrag: "Der Gutschein darf nicht dem aktuellen Vertrag zugeordnet sein."
- Gültig ab des Gutscheins < Gültig ab des aktuell genutzten Gutscheins: "Der Gutschein darf nicht älter als der aktuell genutzte Gutschein sein."
- Neuer Vertragszeitraum überschreitet den Zeitraum des Erhöhungsgutschein: "Der Gutschein ist nicht für den kompletten Erhöhungszeitraum gültig."

- Neuer Betreuungsumfang überschreitet den Betreuungsumfang des Erhöhungsgutschein: "Der gewählte Betreuungsumfang überschreitet den Betreuungsumfang aus dem Erhöhungsgutschein."
- Fehlender Erstellungsgrund des Gutscheins: "Der Gutschein muss einen Erstellungsgrund haben - Betreuungsumfang erhöht."

Nachname des Kindes:

- Pflichtfeld (max. 255 Zeichen)
- Voraussetzung für die Identifizierung des Kindes in Kita-Familienverwaltung

Vorname des Kindes:

- Pflichtfeld (max. 255 Zeichen)
- Voraussetzung für die Identifizierung des Kindes in Kita-Familienverwaltung

Geburtsdatum des Kindes:

- Pflichtfeld (Datum)
- Voraussetzung für die Identifizierung des Kindes in Kita-Familienverwaltung

Bisheriger Betreuungsumfang:

- Pflichtfeld (max. 255 Zeichen)
- Bereits vorbelegt, wird aus dem Vertrag übernommen
- Nicht editierbar
- Mögliche Werte:
 - halbtags (mindestens 4 bis höchstens 5 Stunden)
 - teilzeit (über 5 bis höchstens 7 Stunden)
 - ganztags (über 7 bis höchstens 9 Stunden)

Neuer Betreuungsumfang:

- Pflichtfeld (Auswahl)
- Vorbelegt - es stehen alle Betreuungsumfänge zur Verfügung, die höher als der bisherige sind
- Mögliche Werte:
 - teilzeit (über 5 bis höchstens 7 Stunden)
 - ganztags (über 7 bis höchstens 9 Stunden)
 - erweitert (über 9 Stunden)

Mit Essen:

- Boolean (optional)

- Nur bei Betreuungsumfang "halbtags" editierbar
- Bei Betreuungsumfang > halbtags: Immer aktiv, nicht editierbar

Vertragsabschluss:

- Pflichtfeld (Datum)
- Vertragsabschluss > Vertragsbeginn: "Der Vertragsabschluss darf nicht nach dem Vertragsbeginn liegen."

Neuer Betreuungsumfang beginnt am:

- Pflichtfeld (Datum)
- Start neuer Betreuungsumfang außerhalb Vertragslaufzeit: "Das Datum liegt außerhalb der Laufzeit (bis des Vertrages"
- Tagesdatum > Stichtag 31.03.: "Der Beginn des Betreuungsumfangs muss im aktuellen Kalenderjahr liegen."
- Vertragsbeginn liegt mehr als 9 Monate nach Vertragsabschluss: "Vertragsabschluss und Vertragsbeginn dürfen nicht mehr als 9 Monate auseinander liegen."
- Start neuer Betreuungsumfang = Start bisheriger Vertragsbeginn: "Der Beginn des neuen Betreuungsumfangs muss nach dem aktuellen Vertragsbeginn liegen."
- Neuer Betreuungsumfang beginnt im Vor-Vorjahr: "Die Vertragslaufzeit im vorvergangenen Jahr kann nicht mehr registriert werden."

Neuer Betreuungsumfang endet am:

- Pflichtfeld (Datum)
- Vertragsbeginn > Vertragsende: "Das Vertragsende muss nach dem Vertragsbeginn liegen."
- Vertragsende < Monatsletzter des Vertragsbeginns: "Das Vertragsende muss mindestens auf den Letzten des Monats fallen, in dem der Vertrag beginnt."
- Ende neuer Betreuungsumfang > aktuelles Vertragsende: "Das Ende des neuen Betreuungsumfangs muss auf das Datum des aktuellen Vertragsendes fallen oder vor diesem Datum liegen."

```
{  
  "gutscheinnummer": "string",  
  "kindNachname": "string",  
  "kindVorname": "string",  
  "kindGeburtsdatum": "2020-01-01",  
  "betreuungsumfangNeu": "HALBTAGS",  
  "vertragsabschluss": "2020-01-01",  
  "erhoehungStart": "2020-01-01",  
  "erhoehungEnde": "2020-01-01"  
}
```

Die Anfragedaten müssen dem Typ `betreuungsumfangErhoehen` entsprechen.

6.8.3 Antworten

HTTP Status-Code	Status-Text	Beschreibung
200	OK	Der Auftrag wurde erfolgreich entgegengenommen und verarbeitet.
400	Bad Request	Einer oder mehrere Parameter sind falsch angegeben worden.
401	Unauthorized	Der Header für die Authentisierung ist nicht vorhanden oder konnte nicht interpretiert werden.
403	Forbidden	Der Benutzer konnte nicht im System gefunden werden oder besitzt nicht die erforderlichen Rechte für das Erstellen von Daten mit Hilfe der Dienstschnittstelle.
404	Not Found	Der angefragte Vertrag mit der entsprechenden Vertragsnummer wurde nicht gefunden.

HTTP Status-Code	Status-Text	Beschreibung
409	Conflict	Der Betreuungsumfang kann nicht erhöht werden. Bitte prüfen Sie, ob der Vertrag nicht gekündigt ist, das Finanzierungs- oder Laufzeitende im rückwirkend abrechenbaren Zeitraum liegt, der Betreuungsumfang nicht bereits auf dem höchsten Niveau 'Erweitert' ist und der Vertrag auf einen Basisgutschein registriert ist.
500	Internal Server Error	Bei der Verarbeitung ist es zu einem unvorhergesehenen Problem gekommen. Die Verarbeitung der Nachricht hat nicht stattgefunden.

6.8.3.1 Beispielantwort (erfolgreich)

```
{
  "postingnummer": "PortalWs-2026042105225017-1"
}
```

6.9 Betreuungsumfang verringern

POST /api/v1/betreuung/vertraege/{vertragsnummer}/betreuungsumfangVerringern

6.9.1 Anfrageparameter

Name	In	Typ	Erforderlich	Beschreibung
vertragsnummer	path	string	ja	

6.9.2 Anfragedaten (request body parameter)

Initiiert die Verringerung des Betreuungsumfangs für einen Vertrag. Dabei müssen die folgenden Validierungsregeln beachtet werden:

Nachname des Kindes:

- Pflichtfeld (max. 255 Zeichen)
- Voraussetzung für die Identifizierung des Kindes in Kita-Familienverwaltung

Vorname des Kindes:

- Pflichtfeld (max. 255 Zeichen)
- Voraussetzung für die Identifizierung des Kindes in Kita-Familienverwaltung

Geburtsdatum des Kindes:

- Pflichtfeld (Datum)
- Voraussetzung für die Identifizierung des Kindes in Kita-Familienverwaltung

Bisheriger Betreuungsumfang:

- Pflichtfeld (max. 255 Zeichen)
- Bereits vorbelegt, wird aus dem Vertrag übernommen
- Nicht editierbar
- Mögliche Werte:
 - teilzeit (über 5 bis höchstens 7 Stunden)
 - ganztags (über 7 bis höchstens 9 Stunden)
 - erweitert (über 9 Stunden)

Neuer Betreuungsumfang:

- Pflichtfeld (Auswahl)
- Vorbelegt - es stehen alle Betreuungsumfänge zur Verfügung, die geringer als der bisherige sind
- Mögliche Werte:
 - halbtags (mindestens 4 bis höchstens 5 Stunden)
 - teilzeit (über 5 bis höchstens 7 Stunden)
 - ganztags (über 7 bis höchstens 9 Stunden)

Mit Essen:

- Boolean (optional)
- Nur bei Betreuungsumfang "halbtags" editierbar
- Bei Betreuungsumfang > halbtags: Immer aktiv, nicht editierbar

Vertragsabschluss:

- Pflichtfeld (Datum)
- Vertragsabschluss > Vertragsbeginn: "Der Vertragsabschluss darf nicht nach dem Vertragsbeginn liegen."

Neuer Betreuungsumfang beginnt am:

- Pflichtfeld (Datum)
- Start neuer Betreuungsumfang außerhalb Vertragslaufzeit: "Das Datum liegt außerhalb der Laufzeit (bis des Vertrages"
- Start neuer Betreuungsumfang = Start bisheriger Vertragsbeginn: "Der Beginn des neuen Betreuungsumfangs muss nach dem aktuellen Vertragsbeginn liegen."

```
{  
  "kindNachname": "string",  
  "kindVorname": "string",  
  "kindGeburtsdatum": "2020-01-01",  
  "betreuungsumfangNeu": "HALBTAGS",  
  "vertragsabschluss": "2020-01-01",  
  "verringerungStart": "2020-01-01",  
  "mitEssen": true  
}
```

Die Anfragedaten müssen dem Typ `betreuungsumfangVerringern` entsprechen.

6.9.3 Antworten

HTTP Status-Code	Status-Text	Beschreibung
200	OK	Der Auftrag wurde erfolgreich entgegengenommen und verarbeitet.
400	Bad Request	Einer oder mehrere Parameter sind falsch angegeben worden.
401	Unauthorized	Der Header für die Authentisierung ist nicht vorhanden oder konnte nicht interpretiert werden.

HTTP Status-Code	Status-Text	Beschreibung
403	Forbidden	Der Benutzer konnte nicht im System gefunden werden oder besitzt nicht die erforderlichen Rechte für das Erstellen von Daten mit Hilfe der Dienstschnittstelle.
404	Not Found	Der angefragte Vertrag mit der entsprechenden Vertragsnummer wurde nicht gefunden.
409	Conflict	Der Betreuungsumfang kann nicht verringert werden. Bitte prüfen Sie, ob der Vertrag nicht gekündigt ist, das Finanzierungs- oder Laufzeitende im rückwirkend abrechenbaren Zeitraum liegt, der Betreuungsumfang nicht bereits zu gering ist und der Vertrag auf einen Basisgutschein registriert ist.
500	Internal Server Error	Bei der Verarbeitung ist es zu einem unvorhergesehenen Problem gekommen. Die Verarbeitung der Nachricht hat nicht stattgefunden.

6.9.3.1 Beispielantwort (erfolgreich)

```
{  
  "postingnummer": "PortalWs-2026042105225017-1"  
}
```

6.10 Laufzeit anpassen

POST /api/v1/betreuung/vertraege/{vertragsnummer}/laufzeitAnpassen

Laufzeit eines Vertrages anpassen

6.10.1 Anfrageparameter

Name	In	Typ	Erforderlich	Beschreibung
vertragsnummer	path	string	ja	

6.10.2 Anfragedaten (request body parameter)

Es werden die Informationen zum Vertrag übertragen. Dabei müssen die folgenden Validierungsregeln beachtet werden:

Nachname des Kindes:

- Pflichtfeld (max. 255 Zeichen)
- Voraussetzung für die Identifizierung des Kindes in Kita-Familienverwaltung

Vorname des Kindes:

- Pflichtfeld (max. 255 Zeichen)
- Voraussetzung für die Identifizierung des Kindes in Kita-Familienverwaltung

Geburtsdatum des Kindes:

- Pflichtfeld (Datum)
- Voraussetzung für die Identifizierung des Kindes in Kita-Familienverwaltung

Vertragsende:

- Pflichtfeld (Datum)
- Wird mit dem Vertragsende des aktuell laufenden Vertrages vorbelegt, ist editierbar
- Vertragsbeginn < Vertragsende: "Das Gültigkeitsende muss nach dem Gültigkeitsbeginn liegen."
- Vertragslaufzeit > Einlösbarkeit des Gutscheins: "Der Betreuungszeitraum muss vollständig innerhalb der Einlösbarkeit des Gutscheins liegen: bis ."
- Nur bei Basisgutschein: Vertragsende < Monatsletzter des Vertragsbeginns: "Das Vertragsende muss mindestens auf den Letzten des Monats fallen, in dem der Vertrag beginnt."
- Neues Vertragsende = altes Vertragsende: "Das neue Vertragsende muss geändert werden."
- Vertragsende nach Öffnungszeitraum der Einrichtung: "Die Vertragslaufzeit muss innerhalb des Öffnungszeitraums der ausgewählten Einrichtung liegen."

Sprachförderbedarf:

- Boolean (optional)

Vertragskündigung:

- Boolean (optional)

```
{  
  "kindNachname": "string",  
  "kindVorname": "string",  
  "kindGeburtsdatum": "2020-01-01",  
  "vertragsende": "2019-07-31",  
  "sprachfoerderbedarf": true,  
  "vertragskuendigung": true  
}
```

Die Abfragedaten müssen dem Typ `vertragslaufzeitAnpassen` entsprechen.

6.10.3 Antworten

HTTP Status-Code	Status-Text	Beschreibung
200	OK	Der Auftrag wurde erfolgreich entgegengenommen und verarbeitet.
400	Bad Request	Einer oder mehrere Parameter sind falsch angegeben worden.
401	Unauthorized	Der Header für die Authentisierung ist nicht vorhanden oder konnte nicht interpretiert werden.
403	Forbidden	Der Benutzer konnte nicht im System gefunden werden.
404	Not Found	Der angefragte Vertrag mit der entsprechenden Vertragsnummer wurde nicht gefunden.

HTTP Status-Code	Status-Text	Beschreibung
409	Conflict	Die Vertragslaufzeit kann nicht angepasst werden. Bitte prüfen Sie, ob der Vertrag nicht gekündigt ist, das Finanzierungs- oder Laufzeitende im rückwirkend abrechenbaren Zeitraum liegt und der Vertrag auf einen Basis- oder Sprachfördergutschein registriert ist.
500	Internal Server Error	Bei der Verarbeitung ist es zu einem unvorhergesehenen Problem gekommen. Die Verarbeitung der Nachricht hat nicht stattgefunden.

6.10.3.1 Beispielantwort (erfolgreich)

```
{  
  "postingnummer": "PortalWs-2026042105225017-1"  
}
```

6.11 Vertrag stornieren

POST /api/v1/betreuung/vertraege/{vertragsnummer}/stornieren

Storniert einen Vertrag.

6.11.1 Anfrageparameter

Name	In	Typ	Erforderlich	Beschreibung
vertragsnummer	path	string	ja	

6.11.2 Anfragedaten (request body parameter)

Es werden die Informationen zum Vertrag übertragen. Dabei müssen die folgenden Validierungsregeln beachtet werden:

Nachname des Kindes:

- Pflichtfeld (max. 255 Zeichen)
- Voraussetzung für die Identifizierung des Kindes in Kita-Familienverwaltung

Vorname des Kindes:

- Pflichtfeld (max. 255 Zeichen)
- Voraussetzung für die Identifizierung des Kindes in Kita-Familienverwaltung

Geburtsdatum des Kindes:

- Pflichtfeld (Datum)
- Voraussetzung für die Identifizierung des Kindes in Kita-Familienverwaltung

```
{  
  "kindNachname": "string",  
  "kindVorname": "string",  
  "kindGeburtsdatum": "2020-01-01"  
}
```

Die Anfragedaten müssen dem Typ `vertragStornieren` entsprechen.

6.11.3 Antworten

HTTP Status-Code	Status-Text	Beschreibung
200	OK	Der Auftrag wurde erfolgreich entgegengenommen und verarbeitet.
400	Bad Request	Einer oder mehrere Parameter sind falsch angegeben worden.
401	Unauthorized	Der Header für die Authentisierung ist nicht vorhanden oder konnte nicht interpretiert werden.

HTTP Status-Code	Status-Text	Beschreibung
403	Forbidden	Der Benutzer konnte nicht im System gefunden werden oder besitzt nicht die erforderlichen Rechte für das Abrufen von Daten von der Dienstschnittstelle.
404	Not Found	Der angefragte Vertrag mit der entsprechenden Vertragsnummer wurde nicht gefunden.
409	Conflict	Der Vertrag kann nicht storniert werden. Bitte prüfen Sie die Laufzeit, den Finanzierungszeitraum sowie den Saldo des Vertrags.
500	Internal Server Error	Bei der Verarbeitung ist es zu einem unvorhergesehenen Problem gekommen. Die Verarbeitung der Nachricht hat nicht stattgefunden.

6.11.3.1 Beispielantwort (erfolgreich)

```
{  
  "postingnummer": "PortalWs-2026042105225017-1"  
}
```

6.12 Voraussichtliches Vertragsende melden (Schulrückstellung)

POST /api/v1/betreuung/vertraege/{vertragsnummer}/voraussichtlichesVertragsende

Melde das voraussichtliche Vertragsende (z. B. eine Schulrückstellung) an KiTa.

6.12.1 Anfrageparameter

Name	In	Typ	Erforderlich	Beschreibung
vertragsnummer	path	string	ja	

6.12.2 Anfragedaten (request body parameter)

```
{  
  "voraussichtlichesVertragsende": "2019-07-31",  
  "aenderunggrund": "SCHULRUECKSTELLUNG"  
}
```

Die Anfragedaten müssen dem Typ `voraussichtlichesVertragsende` entsprechen.

6.12.3 Antworten

HTTP Status-Code	Status-Text	Beschreibung
200	OK	Der Auftrag wurde erfolgreich entgegengenommen und verarbeitet.
400	Bad Request	Einer oder mehrere Parameter sind falsch angegeben worden.
401	Unauthorized	Der Header für die Authentisierung ist nicht vorhanden oder konnte nicht interpretiert werden.
403	Forbidden	Der Benutzer konnte nicht im System gefunden werden oder besitzt nicht die erforderlichen Rechte für das Abrufen von Daten von der Dienstschnittstelle.
404	Not Found	Der Vertrag existiert nicht oder ist als gelöscht gekennzeichnet.

HTTP Status-Code	Status-Text	Beschreibung
409	Conflict	Für den Vertrag ist bereits eine Schulrückstellung markiert.
500	Internal Server Error	Bei der Verarbeitung ist es zu einem unvorhergesehenen Problem gekommen. Die Verarbeitung der Nachricht hat nicht stattgefunden.

6.12.3.1 Beispielantwort (erfolgreich)

```
{  
  "postingnummer": "PortalWs-2026042105225017-1"  
}
```

6.13 Voraussichtliches Vertragsende widerrufen

DELETE /api/v1/betreuung/vertraege/{vertragsnummer}/voraussichtlichesVertragsende

Widerruft das zuvor gemeldete voraussichtliche Vertragsende (z. B. eine Schulrückstellung) an KiTa.

6.13.1 Anfrageparameter

Name	In	Typ	Erforderlich	Beschreibung
vertragsnummer	path	string	ja	

6.13.2 Anfragedaten (request body parameter)

```
{  
  "voraussichtlichesVertragsende": "2019-07-31",  
  "aenderunggrund": "SCHULRUECKSTELLUNG"  
}
```

Die Anfragedaten müssen dem Typ `voraussichtlichesVertragsende` entsprechen.

6.13.3 Antworten

HTTP Status-Code	Status-Text	Beschreibung
200	OK	Der Auftrag wurde erfolgreich entgegengenommen und verarbeitet.
400	Bad Request	Einer oder mehrere Parameter sind falsch angegeben worden.
401	Unauthorized	Der Header für die Authentisierung ist nicht vorhanden oder konnte nicht interpretiert werden.
403	Forbidden	Der Benutzer konnte nicht im System gefunden werden oder besitzt nicht die erforderlichen Rechte für das Abrufen von Daten von der Dienstschnittstelle.
404	Not Found	Der Vertrag existiert nicht oder ist als gelöscht gekennzeichnet.
409	Conflict	Um eine Schulrückstellung aufzuheben muss eine Schulrückstellung für den Vertrag markiert sein. Für den Vertrag ist keine Schulrückstellung markiert.
500	Internal Server Error	Bei der Verarbeitung ist es zu einem unvorhergesehenen Problem gekommen. Die Verarbeitung der Nachricht hat nicht stattgefunden.

6.13.3.1 Beispielantwort (erfolgreich)

```
{  
  "postingnummer": "PortalWs-2026042105225017-1"  
}
```

6.14 Anträge eines Kindes abrufen

GET /api/v1/betreuung/kinder/{kindId}/antraege

Liefert eine paginierte Liste aller Anträge für ein Kind, welches über die Kind-ID identifiziert wird. Die Antwort enthält pro Antrag die Antragsnummer sowie eine URL, über die die Detaildaten des jeweiligen Antrags abgerufen werden können.

6.14.1 Anfrageparameter

Name	In	Typ	Erforderlich	Beschreibung
kindId	path	integer(int64)	ja	Die ID des Kindes, dessen Anträge abgerufen werden sollen.
max	query	integer	nein	Die maximale Anzahl an Einträgen, die zurückgegeben werden sollen.
start	query	integer(int64)	nein	Der Index des ersten Eintrags, der zurückgegeben werden soll.

6.14.2 Antworten

HTTP Status-Code	Status-Text	Beschreibung
200	OK	Die Anfrage wurde verarbeitet und es wird eine Liste der Anträge des Kindes geliefert.
401	Unauthorized	Der Header für die Authentisierung ist nicht vorhanden oder konnte nicht interpretiert werden.
403	Forbidden	Der Benutzer konnte nicht im System gefunden werden oder besitzt nicht die erforderlichen Rechte für das Abrufen von Daten von der Dienstschnittstelle.
500	Internal Server Error	Bei der Verarbeitung ist es zu einem unvorhergesehenen Problem gekommen. Die Verarbeitung der Nachricht hat nicht stattgefunden.

6.14.2.1 Beispielantwort (erfolgreich)

```
{
  "metainformationen": {
    "paginierung": {
      "start": 0,
      "max": 0,
      "eintraegeGesamt": 0
    }
  },
  "eintraege": [
    {
      "antragsnummer": "string",
      "url": "string"
    }
  ]
}
```

6.15 Antragdetails abrufen

GET /api/v1/betreuung/antraege/{antragsnummer}

Liefert die Details eines Antrags anhand der Antragsnummer.

6.15.1 Anfrageparameter

Name	In	Typ	Erforderlich	Beschreibung
antragsnummer	path	string	ja	Die Antragsnummer des Antrags, dessen Details abgerufen werden sollen.

6.15.2 Antworten

HTTP Status-Code	Status-Text	Beschreibung
200	OK	Die Anfrage wurde verarbeitet und es werden die Antragdetails geliefert.
401	Unauthorized	Der Header für die Authentisierung ist nicht vorhanden oder konnte nicht interpretiert werden.
403	Forbidden	Der Benutzer konnte nicht im System gefunden werden oder besitzt nicht die erforderlichen Rechte für das Abrufen von Daten von der Dienstschnittstelle.

HTTP Status-Code	Status-Text	Beschreibung
404	Not Found	Es konnte kein Antrag mit der angegebenen Antragsnummer gefunden werden.
500	Internal Server Error	Bei der Verarbeitung ist es zu einem unvorhergesehenen Problem gekommen. Die Verarbeitung der Nachricht hat nicht stattgefunden.

6.15.2.1 Beispielantwort (erfolgreich)

```
{
  "nummer": "string",
  "kindId": 0,
  "antragsdatum": "2019-08-24",
  "antragstyp": "string"
}
```

6.16 Gutscheine eines Antrags abrufen

GET /api/v1/betreuung/antraege/{antragsnummer}/gutscheine

Liefert eine paginierte Liste aller Gutscheine, die zu einem Antrag gehören. Der Antrag wird über die Antragsnummer identifiziert.

6.16.1 Anfrageparameter

Name	In	Typ	Erforderlich	Beschreibung
antragsnummer	path	string	ja	Die Antragsnummer des Antrags, dessen Gutscheine abgerufen werden sollen.
max	query	integer	nein	Die maximale Anzahl an Einträgen, die zurückgegeben werden sollen.
start	query	integer(int64)	nein	Der Index des ersten Eintrags, der zurückgegeben werden soll.

6.16.2 Antworten

HTTP Status-Code	Status-Text	Beschreibung
200	OK	Die Anfrage wurde verarbeitet und es wird eine paginierte Liste der Gutscheine des Antrags geliefert.
401	Unauthorized	Der Header für die Authentisierung ist nicht vorhanden oder konnte nicht interpretiert werden.

HTTP Status-Code	Status-Text	Beschreibung
403	Forbidden	Der Benutzer konnte nicht im System gefunden werden oder besitzt nicht die erforderlichen Rechte für das Abrufen von Daten von der Dienstschnittstelle.
404	Not Found	Es konnte kein Antrag mit der angegebenen Antragsnummer gefunden werden.
500	Internal Server Error	Bei der Verarbeitung ist es zu einem unvorhergesehenen Problem gekommen. Die Verarbeitung der Nachricht hat nicht stattgefunden.

6.16.2.1 Beispielantwort (erfolgreich)

```
{
  "metainformationen": {
    "paginierung": {
      "start": 0,
      "max": 0,
      "eintraegeGesamt": 0
    }
  },
  "eintraege": [
    {
      "gutscheinnummer": "string",
      "url": "string"
    }
  ]
}
```

7 Posting RESTful API der Dienstschnittstelle

Die Dienstschnittstelle stellt eine RESTful-API für Postings bereit.

7.1 Authentifikation und Autorisation

Die API unterstützt die folgenden Verfahren zur Authentifikation.

7.1.1 HMAC

Für weitere Details zur HMAC-Authentifizierung siehe die Entwicklerdokumentation Kapitel 4.

7.2 API Endpunkte und Operationen

7.3 Postings abrufen

GET /api/v1/postings

Liefert eine paginierte Liste von Postings des angemeldeten Benutzers.

7.3.1 Anfrageparameter

Name	In	Typ	Erforderlich	Beschreibung
start	query	integer(int64)	nein	Der Index des ersten Eintrags, der zurückgegeben werden soll.
max	query	integer	nein	Die maximale Anzahl an Einträgen, die zurückgegeben werden sollen.
status	query	array[string]	nein	
von	query	string(date)	nein	Beginn des Zeitraums (basierend auf dem Erstellungsdatum des Postings)

Name	In	Typ	Erforderlich	Beschreibung
bis	query	string(date)	nein	

7.3.2 Antworten

HTTP Status-Code	Status-Text	Beschreibung
200	OK	OK
400	Bad Request	Einer oder mehrere Parameter sind falsch angegeben worden.
401	Unauthorized	Der Header fuer die Authentisierung ist nicht vorhanden oder konnte nicht interpretiert werden.
403	Forbidden	Der Benutzer konnte nicht im System gefunden werden oder besitzt nicht die erforderlichen Rechte.
500	Internal Server Error	Bei der Verarbeitung ist es zu einem unvorhergesehenen Problem gekommen.

7.3.2.1 Beispielantwort (erfolgreich)

```
{
  "metainformationen": {
    "paginierung": {
      "start": 0,
      "max": 25,
      "eintraegeGesamt": 125
    }
  },
  "eintraege": [
    {
      "trackingnummer": "PortalWs-2026042105225017-1",

```

```
    "status": "ERLEDIGT",  
    "meldung": "Posting wurde korrekt verarbeitet."  
  }  
]  
}
```

7.4 Ein Posting abrufen

GET /api/v1/postings/{trackingnummer}

Liefert ein Posting zur angegebenen Trackingnummer.

7.4.1 Anfrageparameter

Name	In	Typ	Erforderlich	Beschreibung
trackingnummer	path	string	ja	

7.4.2 Antworten

HTTP Status-Code	Status-Text	Beschreibung
200	OK	OK
401	Unauthorized	Der Header fuer die Authentisierung ist nicht vorhanden oder konnte nicht interpretiert werden.
403	Forbidden	Der Benutzer konnte nicht im System gefunden werden oder besitzt nicht die erforderlichen Rechte.
404	Not Found	Das angefragte Posting wurde nicht gefunden.
500	Internal Server Error	Bei der Verarbeitung ist es zu einem unvorhergesehenen Problem gekommen.

7.4.2.1 Beispielantwort (erfolgreich)

```
{  
  "trackingnummer": "PortalWs-2026042105225017-1",  
  "status": "ERLEDIGT",  
  "meldung": "Posting wurde korrekt verarbeitet."  
}
```