



Webplattform-API

Schnittstellenbeschreibung

Autor: Stefan Beer

Stand: 10.03.2021

Technischer Kontakt:

webplattform@ratio-software.de

Inhalt

| | |
|---|----|
| Einleitung..... | 3 |
| Nachrichtenformat | 4 |
| Finanzbuchhaltung | 8 |
| Zahlung erfassen | 8 |
| SEPA-Mandate finden | 8 |
| SEPA-Mandate anlegen | 9 |
| Kontingente | 10 |
| Kontingente einer Katalogreise abfragen | 10 |
| Webengine | 12 |
| Katalogreise-Bundle laden | 12 |
| Termin-Bundle laden | 14 |
| Buchung initialisieren | 17 |
| Vorgangsdaten setzen | 18 |
| Teilnehmerdaten setzen..... | 19 |
| Teilnehmer gruppieren..... | 20 |
| Zustieg wählen | 21 |
| Gemeinsamer Zustieg für alle Teilnehmer | 21 |
| Separater Zustieg pro Teilnehmer..... | 21 |
| Sitzplatz wählen..... | 22 |

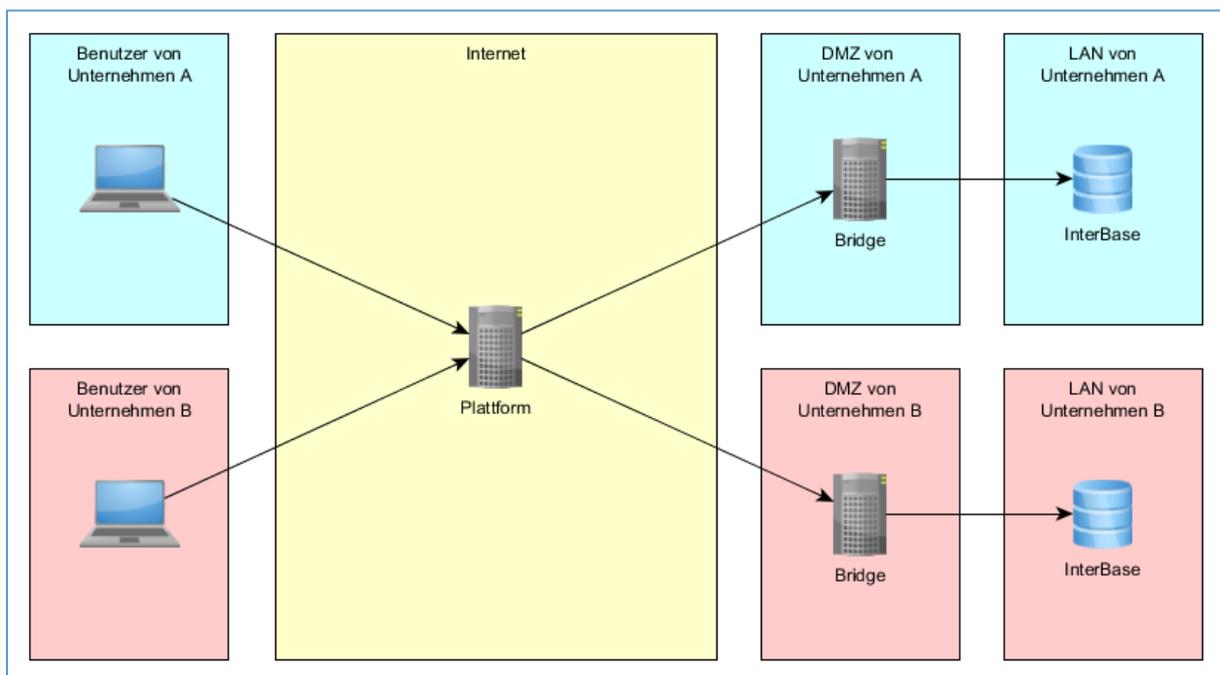
| | |
|--|----|
| Exakten Sitzplatz wählen | 22 |
| Sitzbereich wählen | 23 |
| Kundendaten erfassen | 25 |
| Buchung als Gast | 25 |
| Buchung mit Account | 25 |
| Buchung zusammenfassen | 26 |
| Versicherungen buchen..... | 27 |
| Versicherungen aus den TouPac Stammdaten buchen..... | 27 |
| Versicherungen per Schnittstelle buchen | 30 |
| Gutschein einlösen | 33 |
| Buchung abschließen..... | 34 |
| Zahlungsoptionen..... | 34 |
| Zahlung auf Rechnung | 34 |
| Zahlung per Lastschrift | 34 |
| Zahlung über Dienstleister | 35 |
| Zahlung über Dienstleister in Standardbuchungsmaske..... | 35 |
| Bestätigung erzeugen..... | 37 |
| Gutscheine..... | 37 |
| Zahlung über Dienstleister in Standardbuchungsmaske..... | 37 |

Einleitung

Die Webplattform-API bietet REST-Funktionen zur Entwicklung individueller Erweiterungen für die Ratio-Produktpalette. Die Kommunikation erfolgt per HTTP. Nachrichten werden im JSON-Format übertragen. Voraussetzung zur Verwendung der API ist die Einrichtung eines Mandanten für das Unternehmen in der Webplattform von RATIOSoftware.

Die Webplattform ist eine Hybrid Cloud. Auf einem zentralen Webserver, der von RATIOSoftware verwaltet wird, sind alle Webanwendungen installiert und von überall auf der Welt aufrufbar. Die unternehmensspezifischen Daten verbleiben in der lokalen Datenbank des Unternehmens.

Der Datenaustausch zwischen zentralem Webserver und Unternehmensdatenbank erfolgt über ein Schnittstellenmodul, das einmalig im Unternehmen installiert wird, die sogenannte Bridge. Sobald die Bridge eingerichtet ist, können die verfügbaren Funktionen mit Mandantenkennung und API-Schlüssel aufgerufen werden.



Beim Aufruf einer API-Funktion können Mandantenkennung **[MANDANT]** und API-Schlüssel **[KEY]** auf eine der folgenden Arten übergeben werden:

- HTTP-Header: `x-rs-api-key=[MANDANT]$[KEY]`
- URL-Parameter: `x-rs-api-key=[MANDANT]$[KEY]`
- POST-Request-Body:

```
{
  "data": {...},
  "meta": {
    "credentials_api": {
      "mandant": "[MANDANT]",
      "token": "[KEY]"
    }
  }
}
```

Die Funktionen werden individuell freigeschaltet. Dies richtet sich nach den gekauften Lizenzen. Beim Aufruf einer gesperrten Funktion wird standardmäßig der Fehler **401 Unauthorized** zurückgeliefert.

Jedem Funktionsaufruf muss folgende Basis-URL vorangestellt werden:

<https://www.ratioapp.de/webplattform>

Eine Übersicht aller Funktionen inklusive Live-Test finden Sie hier:

<https://www.ratioapp.de/webplattform/apps/apidoc/>

Nachrichtenformat

Eine JSON-Nachricht besitzt grundsätzlich folgenden Aufbau:

```
{
  "data": {
    "feld_1": "wert_1",
    "feld_2": "wert_2",
    "feld_3": "wert_3"
  },
  "meta": {
    "fields": {
      "feld_1": {
        "table": "t_tabelle",
        "name": "feld_1",
        "type": "char",
        "length1": 40,
        "length2": null
      }
    }
  },
  "errors": [
    {
      "status": 422,
      "code": "feld_1.invalid.error",
      "message": "Ungültiger Wert",
      "value": "",
      "expected_value_1": "",
      "expected_value_2": ""
    }
  ]
}
```

Der Abschnitt **data** enthält die Nutzdaten. Der Inhalt unterscheidet sich für jede API-Funktion und wird in späteren Abschnitten jeweils separat beschrieben. Die Bedeutung von Standardfeldern wird jedoch nicht in jedem Abschnitt wiederholt. Nutzen Sie hierfür unsere Online-Dokumentation des Datenmodells als Referenz: <https://www.ratioapp.de/webplattform/apps2/doc>

Die Elementnamen sind abhängig von der Anzahl der enthaltenen Daten entweder im Singular oder Plural angegeben. Im Deutschen gibt es aber für Singular und Plural oft keine verschiedenen Wörter. Deshalb wird in diesen Fällen für den Plural ein **s** angehängt, z.B. **t_teilnehmers**. Im Datenmodell werden immer die tatsächlichen Tabellennamen aus der Datenbank genannt. Es kann dadurch zu leichten Abweichungen bei den Begrifflichkeiten kommen.

Der Abschnitt **meta** enthält Zusatzinformationen über die Nutzdaten:

| Pfad | Beschreibung |
|---|---|
| meta.fields.[DATA_FIELD].table | Name der Tabelle in der Interbase-Datenbank |
| meta.fields.[DATA_FIELD].name | Name der Tabellenspalte in der Interbase-Datenbank (kann von [DATA_FIELD] abweichen) |
| meta.fields.[DATA_FIELD].type | Typ der Tabellenspalte in der Interbase-Datenbank |
| meta.fields.[DATA_FIELD].length1 | Länge der Tabellenspalte in der Interbase-Datenbank oder Anzahl der Ziffern für den Typ NUMERIC (inklusive Nachkommastellen) |
| meta.fields.[DATA_FIELD].length2 | Anzahl der Nachkommastellen für den Typ NUMERIC |

Der Abschnitt **errors** enthält Fehlermeldungen:

| Pfad | Beschreibung |
|---------------------------------------|---|
| errors[INDEX].status | HTTP-Fehlerstatus: <ul style="list-style-type: none"> • 400 — Bad Request (i.d.R. strukturell ungültige Daten) • 401 — Unauthorized (i.d.R. ungültige Logindaten) • 404 — Not Found (i.d.R. Datensatz nicht gefunden) • 422 — Unprocessable Entity (Validierungsfehler) • 500 — Internal Server Error (Bug! Bitte Nachricht an technischen Kontakt) • 502 — Gateway Error (i.d.R. Problem mit RATIOserver = Bridge-Modul) |
| errors[INDEX].code | Bei einem Validierungsfehler i.d.R. eine Referenz auf ein Datenfeld: <ul style="list-style-type: none"> • [DATA_FIELD].missing.error • [DATA_FIELD].empty.error • [DATA_FIELD].invalid.error • [DATA_FIELD].unknown.error Allgemeine Fehler: <ul style="list-style-type: none"> • request.error • unauthorized.error • unknown.error • gateway.error • error |
| errors[INDEX].message | Beschreibung des Fehlers |
| errors[INDEX].value | Fehlerhafter Wert |
| errors[INDEX].expected_value_1 | Erwarteter Wert, z.B. Minimalwert |
| errors[INDEX].expected_value_2 | Erwarteter Wert, z.B. Maximalwert |

Anmerkungen:

Einige Funktionen liefern Listen mit ergänzender Map zurück, z.B. `t_termine` mit `t_termine_map`. Nutzen Sie die Maps, um mit konstanter Zugriffszeit auf Listenelemente zuzugreifen. Die Schlüssel der Maps bestehen aus der Primärnummer des Datensatzes mit vorangestelltem Unterstrich. Der jeweils zugeordnete Wert entspricht dem Index des Datensatzes in der Liste.

Beispiel:

```
{
  "data": {
    "t_termine": {
      "data": [
        {
          "nr": "392",
          "katalogreisenr": "130",
          "von": "2020-06-12T00:00:00",
          "bis": "2020-06-16T00:00:00"
        },
        { "nr": "393" },
        { "nr": "394" },
        { "nr": "395" }
      ]
    },
    "t_termine_map": {
      "_392": 0,
      "_393": 1,
      "_394": 2,
      "_395": 3
    }
  }
}
```

Weitere Funktionen bieten die Möglichkeit im Request festzulegen, welche Tabellenfelder in der Response enthalten sein sollen und nach welchen Feldern die Ergebnisse sortiert werden sollen.

```
{
  "data": {
    "fields": [
      {
        "name": "kennziffer",
        "alias": "kundennummer"
      }
    ],
    "orderby": [
      {
        "field": "kennziffer",
        "sort": "asc"
      }
    ],
    "limit": "10"
  }
}
```

Felder:

| Name | Beschreibung |
|--------------|--|
| limit | Maximale Anzahl an Datensätzen, die im Response-Paket enthalten sein sollen. Bei der Angabe von -1 erfolgt keine Limitierung. |

fields

| Name | Beschreibung |
|--------------|--|
| name | Name des Tabellenfelds, das in der Response enthalten sein soll. |
| alias | Alternative Bezeichnung des Tabellenfelds in der Response. |

orderby

| Name | Beschreibung |
|--------------|--|
| field | Name des Tabellenfelds, nach dem sortiert werden soll. |
| sort | Sortierungsreihenfolge: asc oder desc |

Welche Features bei welchen Funktionen zur Verfügung stehen ist in der Live-API dokumentiert:

<https://www.ratioapp.de/webplattform/apps/apidoc>

Finanzbuchhaltung

Im Bereich *Finanzbuchhaltung* finden Sie Funktionen zur Zahlungsabwicklung. Beispielsweise können Sie nach einer erfolgreichen Onlinebuchung eine Zahlung, die über einen Dienstleister wie Wirecard, SIX oder Klarna abgewickelt wurde, in das RatioWw-System übernehmen.

Zahlung erfassen

Um eine Zahlung zu erfassen, senden Sie einen **POST**-Request an folgende URL:

```
/services/api/v1/ratio/fibu/save-zahlung
```

Request:

```
{
  "data": {
    "t_vorgang_nr": "200076",
    "soll": "1000",
    "betrag": "90.00",
    "erfasstdurch": "API",
    "text": "Zahlung von Max Mustermann",
    "belegdatum": "2020-05-28T00:00:00"
  }
}
```

Request-Felder:

| Name | Typ | Referenz | Beschreibung |
|---------------------|---------------|-----------------|-------------------------------|
| t_vorgang_nr | INTEGER | T_VORGANG.NR | Vorgangsnummer |
| soll | INTEGER | | SOLL-Konto |
| betrag | NUMERIC(15,2) | | Zahlungsbetrag |
| erfasstdurch | CHAR(20) | USERS.LOGINNAME | Loginname des Users |
| text | CHAR(40) | | <i>optional</i> Belegtext |
| belegdatum | DATE | | <i>optional</i> Belegdatum |

SEPA-Mandate finden

Um die SEPA-Mandate eines Kunden aufzulisten, senden Sie einen **POST**-Request an folgende URL:

```
/services/api/v1/ratio/fibu/find-sepa-mandate
```

Request:

```
{
  "data": {
    "kennziffer": "10263"
  }
}
```

Request-Felder:

| Name | Typ | Referenz | Beschreibung |
|-------------------|---------|---------------------|-----------------------|
| kennziffer | INTEGER | ADRESSEN.KENNZIFFER | Kennziffer des Kunden |

Response:

```
{
  "data": [
    {
      "b2b": "0",
      "beschreibung": "",
      "gesperrt": "NEIN",
      "kennziffer": "10263",
      "letzte_ausfuehrung": "",
      "mandat_datum": "2019-11-26T00:00:00",
      "mandat_id": "10005/10263/01",
      "nr": "10005",
      "refnr": "",
      "reftable": "",
      "sequencetype": "FRST",
      "zweck1": "",
      "zweck2": ""
    }
  ]
}
```

Response-Felder:

siehe Online-Dokumentation der Tabelle [SEPA_MANDATE](#)

SEPA-Mandate anlegen

Um ein neues SEPA-Mandat anzulegen, senden Sie einen **POST**-Request an folgende URL:

```
/services/api/v1/ratio/fibu/create-sepa-mandat
```

Request:

```
{
  "data": {
    "kennziffer": "10263",
    "b2b": "0",
    "sequencetype": "FRST",
    "beschreibung": "Beschreibung",
    "zweck1": "Verwendungszweck 1",
    "zweck2": "Verwendungszweck 2"
  }
}
```

Request-Felder und Response-Felder:

siehe Online-Dokumentation der Tabelle [SEPA_MANDATE](#)

Kontingente

Im Bereich *Kontingente* finden Sie Funktionen zur Abfrage von Paket- und Leistungskontingenten.

Kontingente einer Katalogreise abfragen

Um die Kontingente der Leistungen einer Katalogreise tagesweise abzufragen, senden Sie einen **POST**-Request an folgende URL:

/services/api/v1/ratio/kontingente/find-by-katalogreise

Die Funktion liefert eine Tabelle zurück, in der die angelegten, freien und belegten Kontingente pro Reiseleistung und Tag aufgelistet sind. Die Zeilen der Tabelle enthalten die Leistungen, die Spalten die Kontingente jedes einzelnen Tages im Reisezeitraum. Wird kein Reisezeitraum per Parameter übergeben, wird automatisch der frühesten Abreise- sowie der späteste Rückkehrtermin ermittelt.

Die Informationen der Tabelle entsprechen der Kontingentübersicht in TouPac:

Request:

```
{
  "data": {
    "nr": "130",
    "von": "2021-06-12T00:00:00",
    "bis": "2021-06-16T00:00:00"
  }
}
```

Request-Felder:

| Name | Typ | Referenz | Beschreibung |
|------------|---------|-------------------|----------------------------------|
| nr | INTEGER | T_KATALOGREISE.NR | Katalogreisenummer |
| von | DATE | | <i>optional</i> Abreisedatum |
| bis | DATE | | <i>optional</i> Rückkehrdatum |

Response-Felder:

| Name | Typ | Referenz | Beschreibung |
|------------------------|-----------|----------------------|--|
| leistungen.info | | | |
| typ | INTEGER | | Leistungstyp |
| nr | INTEGER | T_LEISTUNG.NR | Leistungsnummer |
| r1_nr | INTEGER | T_REISELEISTUNGEN.NR | Reiseleistungsnummer |
| lt_nr | INTEGER | T_LTRAEGER.NR | Leistungsträgernummer |
| lt_bezeichnung | CHAR(40) | | Leistungsträgerbezeichnung |
| bezeichnung | CHAR(120) | | Leistungsbezeichnung |
| kontingentcode | CHAR(20) | | Kontingentcode |
| max_angelegt | INTEGER | | Größter Wert der angelegten Kontingente im Zeitraum |
| is_angelegt | BOOLEAN | | Gibt an, ob für die Leistung mindestens ein Kontingent angelegt wurde. |
| leistungen.tage | | | |
| is_angelegt | BOOLEAN | | Gibt an, ob für die Leistung für diesen Tag ein Kontingent angelegt wurde. |
| is_global | BOOLEAN | | Gibt an, ob ein globales oder reisespezifisches Kontingent angelegt wurde. |
| frei | INTEGER | | Anzahl der freien Plätze |
| belegt | INTEGER | | Anzahl der belegten Plätze |
| auf_option | INTEGER | | Anzahl der belegten Plätze mit Optionsdatum |
| angelegt | INTEGER | | Anzahl der insgesamt angelegten Kontingente |
| personen | INTEGER | | Anzahl der gebuchten Personen. |
| verfallsdatum | DATE | | Optionsdatum |

Webengine

Diese Funktionen dienen zur Entwicklung einer individuellen Onlinebuchungstrecke für TouPac Touristik oder TouPac Incoming. Um den vollen Funktionsumfang auszuschöpfen, können jedoch weitere Funktionen anderer Bereiche eingesetzt werden, z.B. Registrierung, Kontingente, Reports.

Katalogreise-Bundle laden

Zum Abruf aller Eckdaten einer Katalogreise senden Sie einen **GET**-Request an folgende URL:

```
/services/api/v1/ratio/webengine-toupac/load-katalogreise-bundle
```

Die Funktion liefert alle Informationen zurück, die speziell für die angegebene Katalogreise gültig sind. Dazu zählen insbesondere die möglichen Reiseternine. Details zu Reiseleistungen und Paketen müssen gesondert per Termin-Bundle (siehe unten) abgerufen werden, da sich Verfügbarkeiten und Preise abhängig vom Reisezeitraum unterscheiden können.

Parameter:

| Name | Typ | Referenz | Beschreibung |
|-------------------------|----------|---------------------|-------------------|
| katalogcode | CHAR(40) | T_KATALOG.CODE | Katalog-Code |
| katalogreisecode | CHAR(20) | T_KATALOGREISE.CODE | Katalogreise-Code |

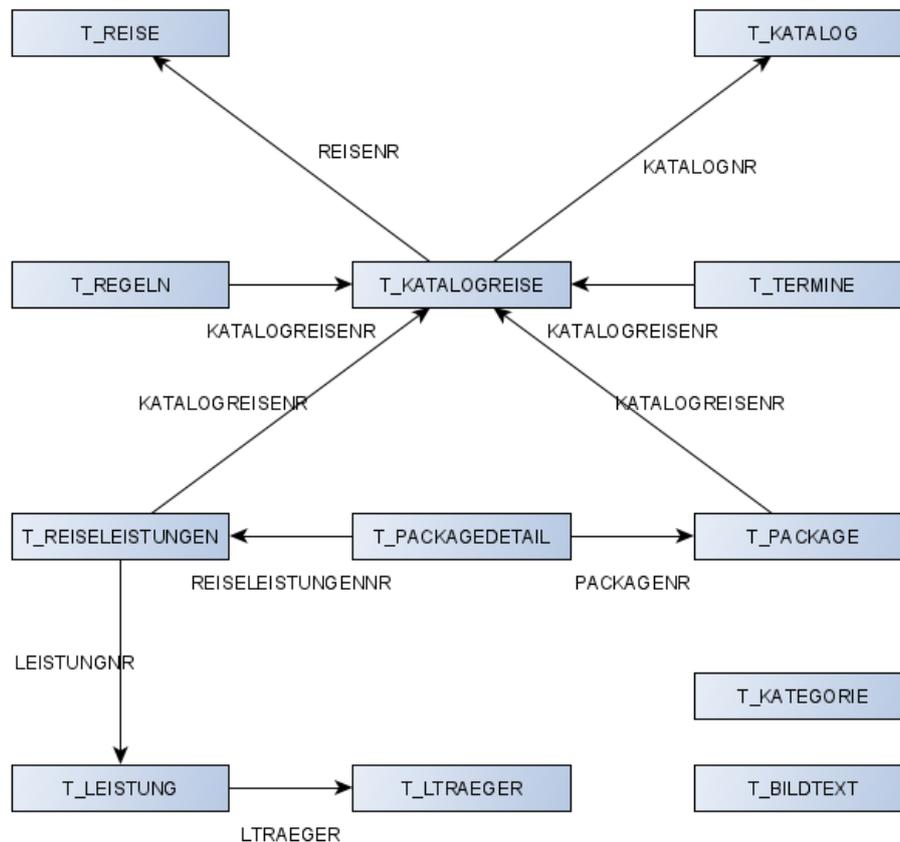
Beispiel:

```
/services/api/v1/ratio/webengine-toupac/load-katalogreise-bundle?katalogcode=K2013&katalogreisecode=ST-BER
```

Response:

```
{
  "data": {
    "t_katalog": {...},
    "t_reise": {...},
    "t_katalogreise": {...},
    "t_termin": {...},
    "t_termin_map": {...},
    "t_termin_vw": {...},
    "t_termin_vw_max": {...},
    "t_reiseleistungen": {...},
    "t_reiseleistungen_map": {...},
    "t_leistungen": {...},
    "t_leistungen_map": {...},
    "t_packages": {...},
    "t_packages_map": {...},
    "t_packagedetails": {...},
    "t_packagedetails_map": {...},
    "t_ltraegers": {...},
    "t_ltraegers_map": {...},
    "t_bildtexte": {...},
    "t_bildtexte_map": {...},
    "t_kategorien": {...},
    "t_kategorien_map": {...},
    "t_regeln": {...},
    "t_regeln_map": {...}
  },
  "meta": {},
  "errors": []
}
```

Response-Datenmodell:



Response-Felder:

t_katalogreise

| Name | Beschreibung |
|---------------------------|--|
| derived_zusatzinfo | <p>Name-Wert-Paare mit Zusatzinformationen zur Katalogreise</p> <p>Name: WEB Pauschalreise Wert: JA JA_TAGES NEIN <i>Gibt an, ob die Reise eine Pauschal- oder Tagespauschalreise ist.</i></p> <p>Name: WEB Passdaten Wert: JA NEIN <i>Gibt an, ob für die Reise die Passdaten der Teilnehmer erfasst werden müssen oder nicht.</i></p> <p>Beispiel:</p> <pre> "derived_zusatzinfo": [{ "name": "WEB Pauschalreise", "wert": "JA_TAGES" }, { "name": "WEB Passdaten", "wert": "NEIN" }] </pre> |

t_termine_vw

Die Felder haben dieselbe Bedeutung wie die Felder von **t_termine**. Der Unterschied bezüglich der Daten besteht darin, dass anhand der Angabe zu den Verlängerungswochen (**vw**) die tatsächlich buchbaren Termine abgeleitet wurden. Diese Termine können in einer Buchungstrecke einzeln zur Auswahl angeboten werden.

t_termine_vw

Die Felder haben dieselbe Bedeutung wie die Felder von **t_termine_vw**. Der Unterschied bezüglich der Daten besteht darin, dass von den abgeleiteten Terminen nur diejenigen Termine mit maximalem Reisezeitraum berücksichtigt sind.

t_packages

| Name | Beschreibung |
|---------------------------------|---|
| derived_zusatzleistungen | Gibt an, ob in dem Paket einzeln buchbare Zusatzleistungen enthalten sind (JA) oder nicht (NEIN). Handelt es sich um keine Zusatzleistungen, muss das Paket als Ganzes gebucht werden. Pro Teilnehmer kann und muss genau ein solches Paket gebucht werden. |

Termin-Bundle laden

Zum Abruf aller Eckdaten eines Termins senden Sie einen **GET**-Request an folgende URL:

```
/services/api/v1/ratio/webengine-toupac/load-termin-bundle
```

Die Funktion liefert alle Informationen zurück, die speziell im angegebenen Reisezeitraum gültig sind. Dazu zählen Leistungspreise, Zustiegsmöglichkeiten und Sitzplatzbelegungen. Je nach Datenumfang kann die Preisermittlung einige Sekunden in Anspruch nehmen. Um ein besseres Benutzererlebnis zu realisieren, kann es sich darum anbieten die Daten aus dem Cache zu laden. Ist der Cache leer, wird automatisch die originäre Preisermittlung durchgeführt.

Parameter:

| Name | Typ | Referenz | Beschreibung |
|------------------|----------------|---------------------|--|
| terminnr | INTEGER | T_TERMINE.NR | Terminnummer (Primärschlüssel) |
| von | DATE | - | <i>optional</i> Reisebeginn abweichend von Termin Beispiel: 2021-06-12T00:00:00 Achtung! Die Doppelpunkte müssen URL-codiert werden: %3A |
| bis | DATE | - | <i>optional</i> Reiseende abweichend von Termin Beispiel: 2021-06-16T00:00:00 Achtung! Die Doppelpunkte müssen URL-codiert werden: %3A |
| use_cache | BOOLEAN | - | <i>optional; Standard: false</i> Daten aus Cache laden: true false |

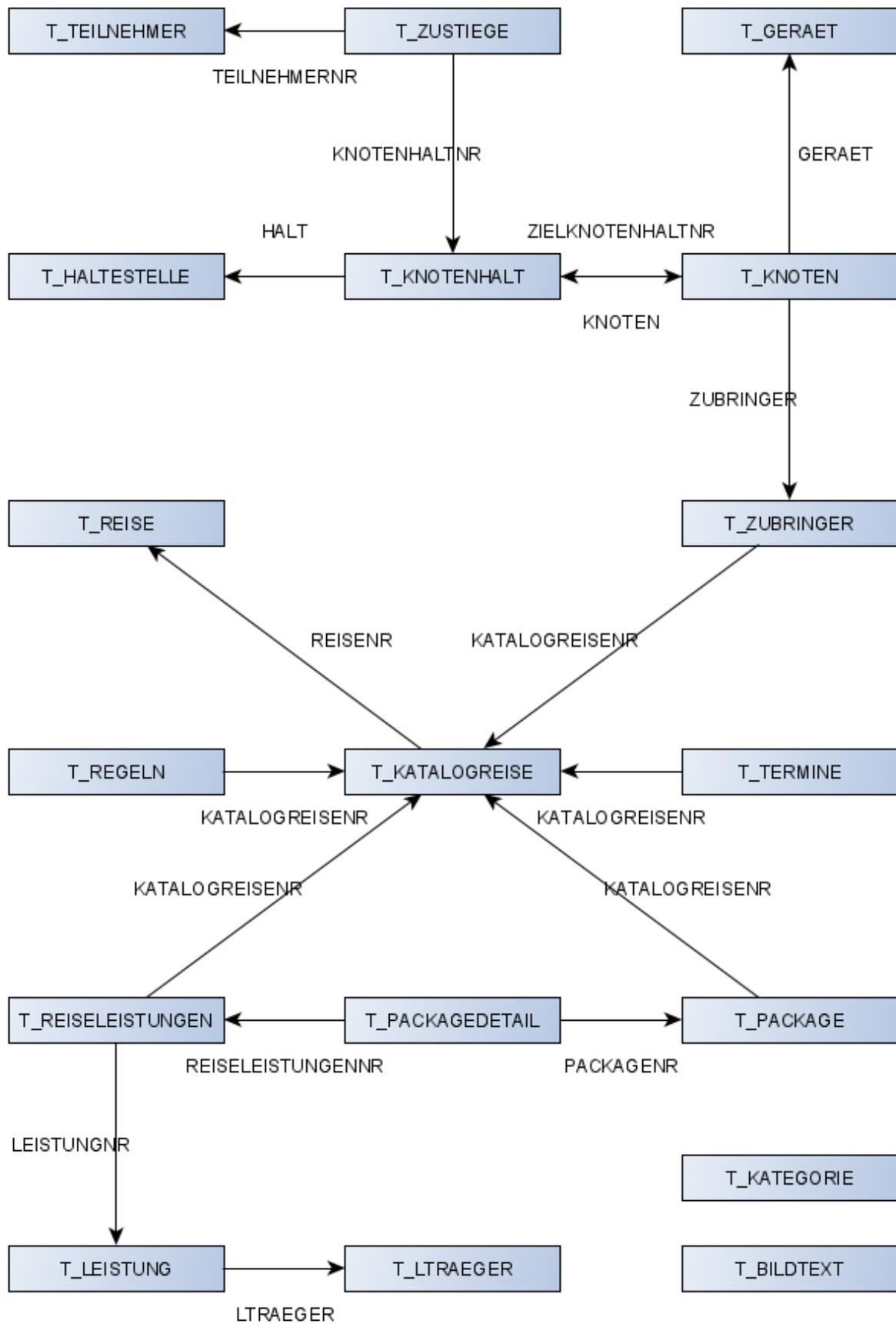
Beispiel:

```
/services/api/v1/ratio/webengine-toupac/load-termin-bundle?terminnr=390&von=2021-06-12T00%3A00%3A00&bis=2021-06-16T00%3A00%3A00&use_cache=true
```

Response:

```
{
  "data": {
    "terminnr": "390",
    "von": "2021-06-12T00:00:00",
    "bis": "2021-06-16T00:00:00",
    "t_termin": {},
    "t_katalogreise": {},
    "t_reise": {},
    "t_kategorien": {},
    "t_packages": {},
    "t_packages_map": {},
    "t_packagedetails": {},
    "t_packagedetails_map": {},
    "t_reiseleistungen": {},
    "t_reiseleistungen_map": {},
    "t_regeln": {},
    "t_regeln_map": {},
    "t_leistungen": {},
    "t_leistungen_map": {},
    "t_ltraegers": {},
    "t_ltraegers_map": {},
    "t_zubringer_hin": {},
    "t_knotens_hin": {},
    "t_knotens_hin_map": [],
    "t_knotenhalte_hin": {},
    "t_knotenhalte_hin_map": [],
    "t_haltestellen_hin": {},
    "t_haltestellen_hin_map": [],
    "t_geraete_hin": {},
    "t_geraete_hin_map": [],
    "t_zustiege_hin": {},
    "t_zustiege_hin_map": [],
    "sitzplatzbelegungen_hin": {},
    "sitzplatzbelegungen_hin_map": [],
    "t_zubringer_rueck": {},
    "t_knotens_rueck": {},
    "t_knotens_rueck_map": [],
    "t_knotenhalte_rueck": {},
    "t_knotenhalte_rueck_map": [],
    "t_haltestellen_rueck": {},
    "t_haltestellen_rueck_map": [],
    "t_geraete_rueck": {},
    "t_geraete_rueck_map": [],
    "t_zustiege_rueck": {},
    "t_zustiege_rueck_map": [],
    "sitzplatzbelegungen_rueck": {},
    "sitzplatzbelegungen_rueck_map": []
  },
  "meta": {},
  "errors": []
}
```

Response-Datenmodell:



Response-Felder:

t_katalogreise

| Name | Beschreibung |
|---------------------------|--|
| derived_zusatzinfo | <p>Name-Wert-Paare mit Zusatzinformationen zur Katalogreise</p> <p>Name: WEB Pauschalreise Wert: JA JA_TAGES NEIN <i>Gibt an, ob die Reise eine Pauschal- oder Tagespauschalreise ist.</i></p> <p>Name: WEB Passdaten Wert: JA NEIN <i>Gibt an, ob für die Reise die Passdaten der Teilnehmer erfasst werden müssen oder nicht.</i></p> <p>Beispiel:</p> <pre>"derived_zusatzinfo": [{ "name": "WEB Pauschalreise", "wert": "JA_TAGES" }, { "name": "WEB Passdaten", "wert": "NEIN" }]</pre> |

t_packages

| Name | Beschreibung |
|---------------------------------|--|
| derived_zusatzleistungen | <p>Gibt an, ob in dem Paket einzeln buchbare Zusatzleistungen enthalten sind (JA) oder nicht (NEIN). Handelt es sich um keine Zusatzleistungen, muss das Paket als Ganzes gebucht werden. Pro Teilnehmer kann und muss genau ein solches Paket gebucht werden.</p> |

t_reiseleistungen

| Name | Beschreibung |
|----------------------|---|
| derived_preis | <p>Regulärer Preis ohne Rabattierung für die Reiseleistung im Reisezeitraum des Termins bzw. im Reisezeitraum, der beim Abruf des Termin-Bundles angegeben wurde.</p> |

Buchung initialisieren

Um eine neue Buchung zu beginnen, senden Sie einen **GET**-Request an folgende URL:

```
/services/api/v1/ratio/webengine-toupac/init-buchung
```

Der Aufruf liefert eine Datenstruktur zurück, die Sie ergänzen und im weiteren Buchungsverlauf an die Funktionen zur Erzeugung einer Zusammenfassung und zur abschließenden Buchung übergeben können. Betrachten Sie die Datenstruktur als den Buchungskontext. Alle relevanten Daten sind hier enthalten. Welche Felder Sie zu welchem Zweck setzen können wird in den folgenden Abschnitten erläutert.

Response:

```
{
  "data": {
    "status": "",
    "buchender_modus": "",
    "zustieg_hin_modus": "BUCHUNG",
    "versicherung_modus": "BUCHUNG",
    "versicherung_extern": false,
    "schlichtungsstelle_checked": false,
    "agb_checked": false,
    "datenschutz_checked": false,
    "pauschalreiserichtlinien_checked": false,
    "frame_ancestor_qualified_url": "",
    "t_teilnehmers_versicherung_extern": [false...]
  ],
  "t_vorgang": {},
  "t_vorgang_buchender": {},
  "t_vorgang_buchender_original": {},
  "t_vorgang_gutscheine": {},
  "t_vorgang_zahlung": {},
  "t_teilnehmers": {},
  "t_positionen": {},
  "t_teilnehmerpositionen": {},
  "t_zustieg_hin": {}
},
"meta": {},
"errors": []
}
```

Vorgangsdaten setzen

Die einzigen Pflichtangaben sind die Terminnummer und der Vorgangstatus. Darüber hinaus können Sie aber auch einen beliebigen Reisezeitraum übergeben. Um textuelle Informationen im Vorgang zu speichern, können Sie das Feld **derived_bemerkung** verwenden. Dies bietet Ihnen beispielsweise die Möglichkeit individuelle Features in Ihrem Frontend umzusetzen, die von der Buchungsstrecke nicht unterstützt werden. Natürlich bedeutet das, dass die erfassten Informationen im Nachgang manuell vom zuständigen Bearbeiter in TouPac korrekt eingepflegt werden müssen.

```
{
  "data": {
    "t_vorgang": {
      "data": {
        "status": "4",
        "terminnr": "390"
      }
    }
  }
}
```

Anmerkungen:

Wenn in der Response gemeldet wird, dass der Buchungsstatus ungültig ist, kann das daran liegen, dass der Status in der Grundkonfiguration der Webplattform für Buchungen nicht freigeschaltet ist.

Für den Fall, dass Sie selbst keinen Zugriff auf die Grundkonfiguration haben, setzen Sie sich bitte mit dem technischen Ansprechpartner in Verbindung.

Teilnehmerdaten setzen

Im Element `t_teilnehmers` setzen Sie die Informationen der Teilnehmer. Welche Felder angegeben werden müssen hängt von der Konfiguration und der Reise ab. Es kann eine Konfiguration für Reisen, bei denen keine Passdaten erforderlich sind, und eine zweite Konfiguration für Reisen mit Passdaten angelegt werden. Die Information um was für eine Reise es sich handelt wird Ihnen im Katalogreise-Bundle zurückgeliefert (`derived_zusatzinfo`). Welche Felder in den jeweiligen Fällen relevant sind, muss individuell abgestimmt werden. Grundsätzlich können Sie aber die Felder der Datenbanktabelle `T_TEILNEHMER` übergeben.

Zusätzlich zu den Personendaten übergeben Sie im Element `derived_t_package`, welches Paket für den Teilnehmer gebucht werden soll. Die Webengine verlangt genau ein Hauptpaket pro Teilnehmer. Der Grundgedanke ist, dass ein Paket nur sinnvoll gebündelte Reiseleistungen enthält.

Individuelle Zusatzleistungen können Sie unter Einsatz des Elements `derived_t_packages_zusatz` buchen. Mit Hilfe der einzeln buchbaren Zusatzleistungen können auch komplexe Buchungsstrecken aufgebaut werden, für die das Prinzip mit den Hauptpaketen nicht greift. Insbesondere im Incoming-Bereich ist dies durchaus üblich.

```
{
  "data": {
    "t_teilnehmers": {
      "data": [{
        "anrede": "H",
        "name": "Max Mustermann",
        "geburtstag": "1981-01-01T00:00:00",
        "derived_t_package": {
          "data": {
            "nr": "1"
          }
        }
      ]},
    "derived_t_packages_zusatz": {
      "data": [{
        "nr": "3",
        "derived_t_packagedetails": {
          "data": [{
            "reiseleistungennr": "283"
          },
          {
            "reiseleistungennr": "249"
          }
        ]
      }
    ]}
  ]}
}
```

Teilnehmer gruppieren

Sie können die Zusammenlegung von Teilnehmern durch Angabe der Gruppennummer beeinflussen. Teilnehmer mit gleicher Gruppennummer werden nach Möglichkeit in einem gemeinsamen Zimmer untergebracht.

```
{
  "data": {
    "t_teilnehmers": {
      "data": [{
        "gruppe": "1",
        "name": "Mustermann, Max"
      },
      {
        "gruppe": "1",
        "name": "Mustermann, Martha"
      },
      {
        "gruppe": "2",
        "name": "Mustermann, Michael"
      }
    ]
  }
}
```

Im Response-Paket der Zusammenfassung (siehe unten) finden Sie die ermittelte Zusammenlegung inklusive der Angabe gemeinsamer Reiseleistungen im Element `t_teilnehmers_gruppen`.

Zustieg wählen

Wurde in TouPac eine Fahrtenplanung für die Katalogreise hinterlegt, können Sie den gewünschten Zustieg für die Hinfahrt übergeben. Der Ausstieg sowie Angaben zur Rückfahrt können derzeit noch nicht definiert werden. Eine Zustiegsangabe ist je nach Konfiguration verpflichtend oder nicht.

Gemeinsamer Zustieg für alle Teilnehmer

Um einen gemeinsamen Zustieg für alle Teilnehmer zu buchen, setzen Sie **zustieg_hin_modus** auf den Wert **BUCHUNG**. Unter Angabe der Knotenhaltnummer bestimmen Sie den Zustiegsort.

```
{
  "data": {
    "zustieg_hin_modus": "BUCHUNG",
    "t_zustieg_hin": {
      "data": {
        "knotenhaltnr": "1637"
      }
    }
  }
}
```

Separater Zustieg pro Teilnehmer

Sie können für jeden Teilnehmer einen eigenen Zustieg wählen. Setzen Sie dazu **zustieg_hin_modus** auf den Wert **TEILNEHMER** und geben Sie bei jedem Teilnehmer den Zustiegsort über die Knotenhaltnummer an.

```
{
  "data": {
    "zustieg_hin_modus": "TEILNEHMER",
    "t_teilnehmers": {
      "data": [{
        "name": "Mustermann, Max",
        "derived_t_zustieg_hin": {
          "data": {
            "knotenhaltnr": "1637"
          }
        }
      },
      {
        "name": "Mustermann, Martha",
        "derived_t_zustieg_hin": {
          "data": {
            "knotenhaltnr": "1638"
          }
        }
      }
    ]
  }
}
```

Sitzplatz wählen

In Ergänzung zur Zustiegsauswahl kann dem Kunden auch die Möglichkeit gegeben werden sich einen Sitzplatz auszusuchen. Zur Auswahl eines exakten Sitzplatzes muss in TouPac bei der Fahrtenplanung ein entsprechender Sitzplatzspiegel hinterlegt sein. Alternativ kann auch ein grober Sitzplatzwunsch abgefragt werden. Diese Variante bietet sich an, wenn die tatsächliche Auslastung der Reise nicht preisgegeben werden soll.

Exakten Sitzplatz wählen

Im Termin-Bundle wird Ihnen die aktuelle Sitzplatzbelegung zurückgeliefert:

```
{
  "data": {
    "sitzplatzbelegungen_hin": {
      "data": [{
        "knotennr": "689",
        "geraetnr": "106",
        "sitze": [{
          "txt": "01",
          "x": 36,
          "y": 100,
          "belegt": false
        }]
      }]
    }
  }
}
```

Felder:

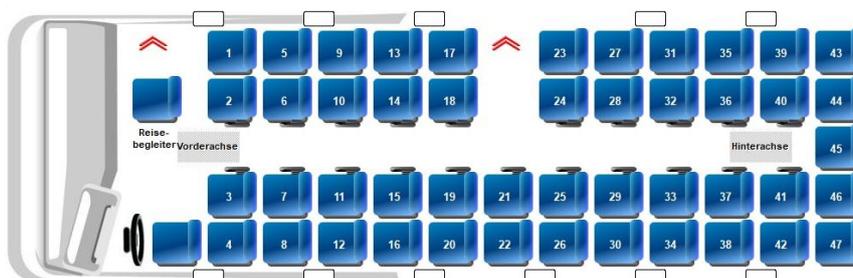
| Name | Referenz | Beschreibung |
|-----------------|-------------|--|
| knotennr | T_KNOTEN.NR | Abfahrtsort des Fahrzeugs |
| geraetnr | T_GERAET.NR | Nummer des Transportmittels und dessen Sitzplatzspiegels |

sitze

| Name | Beschreibung |
|---------------|--|
| txt | Bezeichnung des Sitzplatzes |
| x | Position des Sitzplatzes auf dem Sitzplatzspiegel in Pixel von links |
| y | Position des Sitzplatzes auf dem Sitzplatzspiegel in Pixel von oben |
| belegt | Sitzplatz bereits belegt: true oder false |

Zur Darstellung des Sitzplatzspiegels können Sie zusätzlich die zugehörige Grafik über folgende API-Funktion aus dem Bereich *Fahrtenplanung* abrufen:

```
/services/api/v1/ratio/fahrtenplanung/load-sitzplatzspiegel
```



Geben Sie den Sitzplatz jedes Teilnehmers im Element `derived_t_zustieg_hin` an. Verwenden Sie als Wert die Bezeichnung des Sitzplatzes aus dem Element `txt`.

```
{
  "data": {
    "t_teilnehmers": {
      "data": [{
        "name": "Mustermann, Max",
        "derived_t_zustieg_hin": {
          "data": {
            "platznummer": "01"
          }
        }
      }]
    }
  }
}
```

Der Sitzplatz wird serverseitig in jedem Fall nochmals geprüft. Wenn ein Sitzplatz zwischenzeitlich an einen anderen Reisegast vergeben wurde, enthält die Response eine entsprechende Fehlermeldung.

Sitzbereich wählen

Wenn kein exakter Sitzplatz gewählt werden soll, kann in der Grundkonfiguration auch eine grobe Sitzplatzwahl aktiviert werden. In dem Fall können Wünsche über den Sitzplatzbereich im Fahrzeug erfasst werden: Stock, Reihe, Seite und Platz. Die jeweiligen Feldwerte sind nicht festgelegt.

Gemeinsamen Sitzplatzbereich für alle Teilnehmer wählen

Die Bestimmung eines gemeinsamen Sitzplatzbereichs für alle Teilnehmer funktioniert nur in Kombination mit der gemeinsamen Zustiegsauswahl. Geben Sie im Element `t_zustieg_hin` den gewünschten Bereich an. Sie können Stock, Reihe, Seite und Platz bestimmen. Diese Informationen werden in die Bemerkung des Vorgangs oder in das Feld `INFO2` jedes Teilnehmers eingetragen, je nachdem ob das Feld `INFO2` in der Buchungsstrecke für andere Informationen verwendet wird. Dies ist über die erweiterte Konfiguration der Webplattform steuerbar.

```
{
  "data": {
    "zustieg_hin_modus": "BUCHUNG",
    "t_zustieg_hin": {
      "data": {
        "knotenhaltnr": "1637",
        "derived_stock": "1",
        "derived_reihe": "VORNE",
        "derived_seite": "FAHRERSEITE",
        "derived_platz": "FENSTER"
      }
    }
  }
}
```

Separaten Sitzplatzbereich für jeden Teilnehmer wählen

Wird für jeden Teilnehmer ein separater Zustieg gebucht, kann auch der Sitzplatzbereich für jeden Teilnehmer individuell definiert werden.

```
{
  "data": {
    "zustieg_hin_modus": "TEILNEHMER",
    "t_teilnehmers": {
      "data": [{
        "name": "Mustermann, Max",
        "derived_t_zustieg_hin": {
          "data": {
            "knotenhaltnr": "1637",
            "derived_stock": "1",
            "derived_reihe": "VORNE",
            "derived_seite": "FAHRERSEITE",
            "derived_platz": "GANG"
          }
        }
      }],
      {
        "name": "Mustermann, Martha",
        "derived_t_zustieg_hin": {
          "data": {
            "knotenhaltnr": "1638",
            "derived_stock": "2",
            "derived_reihe": "HINTEN",
            "derived_seite": "TÜRSEITE",
            "derived_platz": "FENSTER"
          }
        }
      }
    ]
  }
}
```

Kundendaten erfassen

Übergeben Sie die Kundendaten im Element `t_vorgang_buchender`. Sie haben die Wahl zwischen der Buchung als Gast und der Buchung mit Account. Die Buchung mit Account hat den großen Vorteil, dass keine Adressdoubletten in der Datenbank erzeugt werden. Außerdem hat der Kunde die Möglichkeit seine Vorgänge im Kundenportal nachträglich einzusehen.

Buchung als Gast

Bei der Buchung als Gast wird in jedem Fall eine neue Adresse in der Datenbank angelegt. Es findet keine Doublettenprüfung statt. Die Pflichtangaben bestehen mindestens aus den Feldern `NAME2` und `EMAIL`. Weitere Felder können in der Grundkonfiguration zu Pflichtfeldern gemacht werden.

```
{
  "data": {
    "buchender_modus": "GAST",
    "t_vorgang_buchender": {
      "data": {
        "name1": "Max",
        "name2": "Mustermann",
        "strasse": "Nobelstraße 17",
        "plz": "89584",
        "ort": "Ehingen",
        "email": "demo@ratio-software.de"
      }
    }
  }
}
```

Buchung mit Account

Hat sich der Kunde registriert, kann er unter Angabe seiner Accountdaten buchen. Abhängig von der Konfiguration werden die Adressangaben beim Buchungsabschluss in der Datenbank aktualisiert.

```
{
  "data": {
    "buchender_modus": "ACCOUNT",
    "t_vorgang_buchender": {
      "data": {
        "name1": "Max",
        "name2": "Mustermann",
        "strasse": "Nobelstraße 17",
        "plz": "89584",
        "ort": "Ehingen",
        "email": "demo@ratio-software.de"
      }
    }
  },
  "meta": {
    "credentials_3rd": {
      "mandant": "demo",
      "email": "demo@ratio-software.de",
      "pw": "demo"
    }
  }
}
```

Buchung zusammenfassen

Um eine Buchung zu prüfen, senden Sie einen **POST**-Request an folgende URL:

```
/services/api/v1/ratio/webengine-toupac/summarize-buchung
```

Übergeben Sie im Request-Body den Buchungskontext, den Sie mit Ihren Angaben ergänzt haben. Die Funktion führt eine Kontingentprüfung durch, ermittelt die Zimmerbelegung und berechnet den Reisepreis. Das Ergebnis dient als Übersicht für den Kunden und bildet die Basis für die Buchung von Versicherungsleistungen. Auch Gutscheine lassen sich jetzt einlösen.

Anmerkung:

Bei diesem Aufruf werden noch keine Daten in der Datenbank gespeichert.

Response:

```
{
  "data": {
    "t_vorgang_preis": {
      "data": {
        "zwischenstufe_vor_rabatten": "689.00",
        "reisepreis": "689.00",
        "zwischenstufe_nach_versicherungen": "689.00",
        "gesamtpreis_vor_gutscheinen": "689.00",
        "anzahlung_vor_gutscheinen": "154.40",
        "restzahlung_vor_gutscheinen": "534.60",
        "gutscheine_gesamt": "10.00",
        "gesamtpreis": "679.00",
        "anzahlung": "144.40",
        "restzahlung": "534.60",
        "restzahlung_faellig_am": "2021-05-13T00:00:00",
        "restzahlung_faellig_sofort": false,
        "zahlung_required": true,
        "zahlung_sofort": "144.40",
        "sonderaktion_berechnet": false
      }
    }
  }
}
```

Response-Felder:

t_vorgang_preis

| Name | Beschreibung |
|--|--|
| zwischenstufe_vor_rabatten | Preis aller gebuchten Reiseleistungen ohne Berücksichtigung von Rabatten. |
| reisepreis | Preis aller gebuchten Reiseleistungen inklusive Rabattierung. Der Reisepreis bildet die Grundlage zur Buchung von Reiseversicherungen. |
| zwischenstufe_nach_versicherungen gesamtpreis_vor_gutscheinen | Preis inklusive gewünschter Versicherungsleistungen (beide Felder enthalten denselben Wert) |
| anzahlung_vor_gutscheinen | Zu bezahlender Zahlungsbetrag ohne Berücksichtigung von Gutscheinen. |

| | |
|------------------------------------|--|
| restzahlung_vor_gutscheinen | Zu bezahlender Restbetrag ohne Berücksichtigung von Gutscheinen. |
| gutscheine_gesamt | Gesamtwert der eingelösten Gutscheine. |
| gesamtpreis | Zu bezahlender Gesamtbetrag nach Abzug von eingelösten Gutscheinen. |
| anzahlung | Zu bezahlender Anzahlungsbetrag nach Abzug von eingelösten Gutscheinen. |
| restzahlung | Zu bezahlender Restbetrag nach Abzug von eingelösten Gutscheinen. |
| restzahlung_faellig_am | Fälligkeitsdatum der Restzahlung |
| restzahlung_faellig_sofort | Gibt an, ob die Restzahlung sofort zu leisten ist. Mögliche Werte: true false |
| zahlung_required | Gibt an, ob eine Zahlung zu leisten ist. Beispiel: Keine Zahlung bei Buchung auf Warteliste Mögliche Werte: true false |
| zahlung_sofort | Zu bezahlender Betrag zum aktuellen Zeitpunkt. |
| sonderaktion_berechnet | Gibt an, ob eine Sonderaktion berechnet wurde. Mögliche Werte: true false |

Versicherungen buchen

In der Grundkonfiguration kann eingestellt werden, ob passende Versicherungen aus dem TouPac Leistungsstamm oder über die Versicherungsschnittstelle ermittelt und gebucht werden sollen. Die richtige Variante ist mit dem Unternehmen abzustimmen.

Versicherungen aus den TouPac Stammdaten buchen

Bei der Erzeugung der Buchungszusammenfassung werden anhand der übergebenen Daten passende Versicherungen aus dem TouPac Leistungsstamm ermittelt. Es werden Versicherungsleistungen aufgelistet, die sich im übergreifenden Katalog oder in der Katalogreise befinden.

```
{
  "data": {
    "t_vorgang_versicherungen": {
      "data": {
        "t_reiseleistungen": {},
        "t_leistungen": {},
        "t_ltraegers": {},
        "t_bildtext_leistungen": {}
      },
      "t_teilnehmers_versicherungen": {},
      "t_teilnehmers_versicherungen_gruppen": {}
    }
  }
}
```

Gemeinsame Versicherung aller Teilnehmer

Um alle Teilnehmer gemeinsam zu versichern, übergeben Sie die Nummer der zu buchenden Reiseleistung im Element `derived_t_reiseleistungen_versicherungen`. Wenn in der Grundkonfiguration die Kombination von Versicherungen erlaubt ist, können Sie auch mehrere Versicherungen buchen.

```
{
  "data": {
    "versicherung_modus": "BUCHUNG",
    "versicherung_extern": false,
    "t_vorgang": {
      "data": {
        "terminnr": "390",
        "status": "4",
        "derived_t_reiseleistungen_versicherungen": {
          "data": [{
            "nr": "186"
          }]
        }
      }
    }
  }
}
```

Separate Versicherung jedes Teilnehmers

Um jeden Teilnehmer separat mit einer Versicherung auszustatten, übergeben Sie die Nummern der zu buchenden Reiseleistungen in den Elementen `derived_t_reiseleistungen_versicherungen` auf Teilnehmerebene. Setzen Sie außerdem für jeden Teilnehmer das entsprechende indexbasierte Flag im Element `t_teilnehmers_versicherung_extern` auf den Wert `false`.

```
{
  "data": {
    "versicherung_modus": "TEILNEHMER",
    "t_teilnehmers_versicherung_extern": [false, false],
    "t_teilnehmers": {
      "data": [{
        "derived_t_reiseleistungen_versicherungen": {
          "data": [{
            "nr": "183"
          }]
        }
      },
      {
        "derived_t_reiseleistungen_versicherungen": {
          "data": [{
            "nr": "184"
          }]
        }
      }
    ]
  }
}
```

Versicherung von Teilnehmergruppen

Teilnehmer können gruppenweise versichert werden. Legen Sie im ersten Schritt die Einteilung der Teilnehmer fest und erzeugen Sie die Zusammenfassung per **summarize-buchung**:

```
{
  "data": {
    "versicherung_modus": "GRUPPEN",
    "versicherung_extern": false,
    "t_teilnehmers": {
      "data": [{
        "name": "Mustermann, Max",
        "derived_versicherung_gruppe": "1"
      },
      {
        "name": "Mustermann, Martha",
        "derived_versicherung_gruppe": "1"
      },
      {
        "name": "Mustermann, Michael",
        "derived_versicherung_gruppe": "2"
      }
    ]
  }
}
```

Im Response-Paket werden Ihnen im Element **t_teilnehmers_versicherungen_gruppen** passende Versicherungen für jede Gruppe aufgelistet. Im zweiten Schritt können Sie in demselben Element die gewünschten Reiseleistungen für jede Gruppe angeben:

```
{
  "data": {
    "versicherung_modus": "GRUPPEN",
    "versicherung_extern": false,
    "t_teilnehmers": {},
    "t_teilnehmers_versicherungen_gruppen": {
      "data": [{
        "derived_t_reiseleistungen": {
          "data": [{
            "nr": "186"
          }
        ]
      },
      {
        "derived_t_reiseleistungen": {
          "data": [{
            "nr": "183"
          }
        ]
      }
    ]
  }
}
```

Versicherungen per Schnittstelle buchen

Bei aktivierter Versicherungsschnittstelle werden in der Response der Buchungszusammenfassung passende Versicherungen zurückgeliefert, aus denen der Kunde wählen kann. In der Regel werden das nur zwei oder drei verschiedene Produkte sein, um den Kunden nicht zu überfordern:

```
{
  "data": {
    "t_vorgang_versicherungen_extern": {
      "data": {
        "produkte": {
          "data": {
            "ergo": {
              "data": [{
                "code": "VK19RSM",
                "bezeichnung": "TEST-RundumSorglos-Schutz...",
                "preis": "49.00",
                "multi": "NEIN",
                "angebot_id": "146109551",
                "bedingungen_url": "https://www.reisevers...",
                "info_url": "https://www.reiseversicherun...",
                "tarif_list": {
                  "data": [{
                    "produkt_id": 2,
                    "code": "PBM106",
                    "bezeichnung": "TEST-RundumSorglos-Sc...",
                    "preis": "49.00",
                    "ipid_url": "https://egate2.erv.de/es..."
                  }]
                }
              }]
            }
          }
        }
      }
    },
    "t_teilnehmers_versicherungen_extern": {}
  }
}
```

Response-Felder:

t_vorgang_versicherungen_extern

Dieses Element enthält passende Produkte zur gemeinsamen Versicherung aller Teilnehmer.

t_teilnehmers_versicherungen_extern

Dieses Element enthält für jeden Teilnehmer die für ihn passenden Produkte, wenn jeder Teilnehmer separat versichert werden soll.

Achtung:

- Versicherungsnehmer ist immer der buchende Kunde, da nur für ihn alle relevanten Adressdaten erfasst werden können.
- Es können nur alle Teilnehmer gemeinsam oder jeder Teilnehmer separat versichert werden. Eine Versicherung von flexiblen Teilnehmergruppen ist über die Schnittstelle nicht möglich.
- Die Buchung muss im Status 4 (Festbuchung) gespeichert werden.

Abfolge der Versicherungsbuchung:

1. Erzeugung der Zusammenfassung per **summarize-buchung**
2. Darstellung des Reisepreises in der Buchungsstrecke
3. Angabe der gewünschten Versicherungen
4. Erneute Erzeugung der Zusammenfassung mit Übergabe der Versicherungen
5. Darstellung des Reisepreises und der ausgewählten Versicherungen
6. Abschluss der Buchung per **submit-buchung**

Für jeden per Schnittstelle angebotenen Versicherungsanbieter wird in der Response ein eigener Knoten zurückgeliefert. Der Knotenname wird auch bei der anschließenden Versicherungsbuchung zur Unterscheidung der Anbieter verwendet:

- **ergo** = ERGO Reiseversicherung Deutschland
- **erv_at** = Europäische Reiseversicherung Österreich (geplant)

Gemeinsame Versicherung aller Teilnehmer

Geben Sie den Code der gewünschten Versicherung im Element **derived_versicherungen_extern** an. Theoretisch können auch mehrere Versicherungen gebucht werden, praktisch wird das jedoch im Regelfall nicht sinnvoll sein.

```
{
  "data": {
    "versicherung_modus": "BUCHUNG",
    "versicherung_extern": true,
    "t_vorgang": {
      "data": {
        "status": "4",
        "derived_versicherungen_extern": {
          "data": [{
            "code": "VK19RSM",
            "provider": "ergo"
          }]
        }
      }
    }
  }
}
```

Separate Versicherung jedes Teilnehmers

Geben Sie den Code der gewünschten Versicherung im Element `derived_versicherungen_extern` jedes Teilnehmers an. Setzen Sie außerdem für jeden Teilnehmer, für den eine Versicherung über die Schnittstelle gebucht werden soll, in der Liste `t_teilnehmers_versicherung_extern` das Element auf den Wert `true`, dessen Index dem zu versichernden Teilnehmer entspricht.

```
{
  "data": {
    "versicherung_modus": "TEILNEHMER",
    "t_teilnehmers_versicherung_extern": [true, true],
    "t_teilnehmers": {
      "data": [{
        "anrede": "H",
        "name": "Mustermann, Max",
        "geburtstag": "1981-01-01T00:00:00",
        "derived_versicherungen_extern": {
          "data": [{
            "code": "VK19RRVRABM",
            "provider": "ergo"
          }]
        }
      }],
      },
      {
        "anrede": "F",
        "name": "Mustermann, Martha",
        "geburtstag": "1982-02-02T00:00:00",
        "derived_versicherungen_extern": {
          "data": [{
            "code": "VK19RSM",
            "provider": "ergo"
          }]
        }
      }
    ]
  }
}
```

Gutschein einlösen

Gutscheine aus der RatioWw Gutscheinverwaltung können in der Buchungsstrecke eingelöst werden. Übergeben Sie dazu den Gutscheincode im Element `t_vorgang_gutscheine`. Abhängig von der Einstellung in der Grundkonfiguration können auch mehrere Gutscheine zusammen eingelöst werden. Voraussetzung zur Einlösung ist, dass der Gutschein vollständig bezahlt wurde. Außerdem kann ein Gutschein nur als Ganzes eingelöst werden. Teilbeträge werden nicht unterstützt.

```
{
  "data": {
    "t_vorgang_gutscheine": {
      "data": [{
        "code": "2ZPGXXGAMD"
      }]
    }
  }
}
```

Abfolge zur Einlösung von Gutscheinen:

1. Erzeugung der Zusammenfassung per `summarize-buchung`
2. Darstellung des Reisepreises in der Buchungsstrecke
3. Eingabe eines Gutscheincode durch den Kunden
4. Erneute Erzeugung der Zusammenfassung mit Übergabe des Gutscheincode
5. Darstellung des Reisepreises und des eingelösten Gutscheins
6. Abschluss der Buchung per `submit-buchung`

Response:

```
{
  "data": {
    "t_vorgang_gutscheine": {
      "data": [
        {
          "nr": "10007",
          "code": "2ZPGXXGAMD",
          "betrag": "10.00"
        }
      ]
    }
  }
}
```

Response-Felder:

t_vorgang_gutscheine

| Name | Typ | Referenz | Beschreibung |
|---------------------|----------------------------|-----------------------------|---------------------|
| <code>nr</code> | <code>INTEGER</code> | <code>GUTSCHEIN.NR</code> | Primärschlüssel |
| <code>code</code> | <code>CHAR(20)</code> | <code>GUTSCHEIN.CODE</code> | Gutscheincode |
| <code>betrag</code> | <code>NUMERIC(15,2)</code> | | Wert des Gutscheins |

Buchung abschließen

Um eine Buchung abzuschließen, senden Sie einen **POST**-Request an folgende URL:

```
/services/api/v1/ratio/webengine-toupac/submit-buchung
```

Übergeben Sie denselben Buchungskontext, den Sie zuvor an die Funktion **summarize-buchung** geschickt haben. Mit diesem Aufruf werden nochmals alle Kontingente geprüft, die Buchung in TouPac erfasst und Versicherungen bei aktivierter Schnittstelle gebucht.

Zahlungsoptionen

Die Zahlung kann auf Rechnung, per Lastschrift oder über die Anbindung eines Zahlungsanbieters erfolgen. Zahlungsanbieter müssen immer direkt angebunden werden. Die API-Funktionen der Webplattform bieten hierfür keine direkte Unterstützung.

Zahlung auf Rechnung

Die Zahlung auf Rechnung stellt die einfachste Zahlungsart dar. Es sind keine zusätzlichen Angaben erforderlich. Nach Buchungsabschluss können Sie über die Funktion **generate** aus dem Bereich *Reports* eine Rechnung im PDF-Dokument generieren und dem Kunden per E-Mail schicken. Alternativ kann die Mail auch durch die Engine verschickt werden. Welche Variante zum Einsatz kommen soll, kann in der Grundkonfiguration der Webplattform eingestellt werden.

```
{
  "data": {
    "t_vorgang_zahlung": {
      "data": {
        "zahlungsart": "RECHNUNG"
      }
    }
  }
}
```

Zahlung per Lastschrift

Bei der Zahlung per Lastschrift müssen die Bankdaten des Kunden übermittelt werden. Bei Abschluss der Buchung wird automatisch ein SEPA-Mandat generiert. Der Bankeinzug findet nicht automatisch statt. Dies muss vom zuständigen Bearbeiter manuell durchgeführt werden.

```
{
  "data": {
    "t_vorgang_zahlung": {
      "data": {
        "zahlungsart": "BASIS-LS (B2C)",
        "iban": "DE07123412341234123412",
        "bic": "MARKDEFF",
        "bank": "Testbank",
        "kontoinhaber": "Max Mustermann"
      }
    }
  }
}
```

Nach Buchungsabschluss wird die Mandats-ID im Response-Paket zurückgeliefert:

```
{
  "data": {
    "t_vorgang_zahlung": {
      "data": {
        "mandat_id": "10001/10268/01"
      }
    }
  }
}
```

Zahlung über Dienstleister

Wenn Sie die Zahlung über einen Dienstleister wie Wirecard, SIX oder Klarna abwickeln möchten, können Sie als Zahlungsart **PROVIDER** übergeben. Beim Abschluss der Buchung erfolgt keine weitere Validierung (analog zur Zahlung auf Rechnung). In der Bemerkung des Vorgangs ist aber ersichtlich, dass der Kunde diese Zahlungsart gewählt hat.

```
{
  "data": {
    "t_vorgang_zahlung": {
      "data": {
        "zahlungsart": "PROVIDER"
      }
    }
  }
}
```

Nach Abschluss der Buchung können Sie zur Bezahlseite des Dienstleisters weiterleiten. Verwenden Sie die Funktion **save-zahlung** aus dem Bereich *Finanzbuchhaltung* (siehe oben), um die Zahlung des Kunden in TouPac zu erfassen.

Wichtig: Schließen Sie die Buchung immer zuerst ab bevor Sie zum Zahlungsanbieter verzweigen. Andernfalls kann es passieren, dass der Kunde eine Zahlung veranlasst, die Buchung jedoch nicht mehr abgeschlossen werden kann, weil in der Zwischenzeit nicht mehr ausreichend Kontingente für die gewählten Reiseleistungen zur Verfügung stehen.

Zahlung über Dienstleister in Standardbuchungsmaske

Wird die Standardbuchungsmaske von Ratio per iFrame in eine Website eingebettet, kann die Zahlung trotzdem über einen beliebigen Zahlungsanbieter abgewickelt werden. Hierzu ist es notwendig in der Grundkonfiguration die Option **webbuchung.zahlung_custom_aktiv** auf den Wert **JA** zu setzen.

In diesem Fall hat der Kunde die Möglichkeit auf der zweiten Seite (Zusammenfassung) eine weitere Zahlungsart auszuwählen. Die Bezeichnung sowie der beschreibende Zusatztext können ebenfalls in der Grundkonfiguration bei den Übersetzungen angepasst werden:

- **WEBBUCHUNG.VIEW.ZAHLUNG.CUSTOM.LABEL**
- **WEBBUCHUNG.VIEW.ZAHLUNG.CUSTOM.TEXT**

Die Buchungstrecke wird regulär durchlaufen und nach Klick auf *Zahlungspflichtig buchen* ein neuer Vorgang in TouPac angelegt.

Anschließend werden die relevanten Zahlungsdaten per **postMessage** an das umgebende Parent-Window geschickt. Das MessageEvent enthält in der data-Eigenschaft folgende Werte:

| Name | Typ | Referenz | Beschreibung |
|---------------------------|---------------|-----------------------|---|
| messageId | STRING | | Identifiziert zur Erkennung des Nachrichtentyps für das Parent-Window. Enthält immer den Wert: ratio-custom-zahlung |
| betrag | NUMERIC(15,2) | | Zahlungsbetrag |
| waehrung | VARCHAR(255) | | Währung laut Grundkonfiguration |
| vorgangsnummer | INTEGER | T_VORGANG.NR | TouPac-Vorgangsnummer |
| rechnungsnummer | INTEGER | T_VORGANG.RECHNUNGSNR | TouPac-Rechnungsnummer |
| successUrl | STRING | | URL zur Fortsetzung der Buchungsstrecke nach einer erfolgreichen Zahlung |
| failureUrl | STRING | | URL zur Fortsetzung der Buchungsstrecke nach einer fehlgeschlagenen Zahlung |
| kundeKennziffer | INTEGER | ADRESSEN.KENNZIFFER | Kennziffer des Kunden |
| kundeAnrede | CHAR(20) | | Anrede |
| kundeTitel | CHAR(20) | | Titel |
| kundeName1 | CHAR(30) | | Vorname |
| kundeName2 | CHAR(30) | | Nachname |
| kundeName3 | CHAR(30) | | Zusatz |
| kundeNamenszusatz | CHAR(60) | | Namenszusatz |
| kundeNamenszusatz2 | CHAR(120) | | Namenszusatz (Verlängerung) |
| kundeStrasse | CHAR(30) | | Straße |
| kundeStrasse2 | VARCHAR(80) | | Straße (Verlängerung) |
| kundePlz | CHAR(15) | | Postleitzahl |
| kundeOrt | CHAR(30) | | Ort |
| kundeOrt2 | CHAR(80) | | Ort (Verlängerung) |
| kundeLand | CHAR(15) | | Ländercode |
| kundeRegion | CHAR(20) | | Region |
| kundeTelefon1 | CHAR(25) | | Telefonnummer 1 |
| kundeTelefon2 | CHAR(25) | | Telefonnummer 2 |
| kundeFax | CHAR(25) | | Faxnummer |
| kundeEmail | CHAR(60) | | E-Mail-Adresse |

Diese Informationen können je nach Bedarf an den angebotenen Zahlungsdienstleister übergeben werden. Im Anschluss an die Zahlungsabwicklung kann entweder die **successUrl** oder **failureUrl** aufgerufen werden. In beiden Fällen wird die Buchung abgeschlossen. Bei Aufruf der **successUrl** werden zusätzlich Versicherungen per Schnittstelle gebucht (sofern die Schnittstelle aktiv ist). Bei Aufruf der **failureUrl** erfolgt dies nicht; stattdessen wird ein Hinweis über die fehlgeschlagene Zahlung auf der Bestätigungsseite und in der Bestätigungsmail (sofern aktiv) angezeigt.

Es ist nicht zwingend erforderlich eine der beiden URLs aufzurufen. Stattdessen kann auch eine eigene Bestätigungsseite angezeigt und eine eigene Mail versandt werden. Versicherungen müssen allerdings vom Sachbearbeiter nachträglich in TouPac gebucht werden (sofern die Schnittstelle benutzt wird).

Achtung: Die Zahlung muss in jedem Fall selbst erfasst werden. Dazu kann die API-Funktion **save-zahlung** aus dem Bereich *Finanzbuchhaltung* verwendet werden.

Bestätigung erzeugen

Abhängig von der Grundkonfiguration erhält der Kunde nach Buchungsabschluss automatisch eine Bestätigung per E-Mail. Sie können die Bestätigung unter Verwendung der Funktion **generate** aus dem Bereich *Reports* aber jederzeit selbst erzeugen.

Zur Erzeugung der Bestätigung senden Sie einen **GET**-Request an folgende URL:

```
/services/api/v1/ratio/reports/generate
```

Parameter:

| Name | Beschreibung |
|----------------|--|
| bereich | Bereich des Formulars Beispiel: T-TOURISTIK |
| art | Art des Formulars Beispiel: BESTAETIGUNG |
| name | Name des Formulars Beispiel: webformular |
| format | Wir empfehlen die Verwendung des PDF-Formats. Hier ist sichergestellt, dass alle Informationen korrekt dargestellt werden. Mögliche Werte: pdf html jpg |
| id | Übergeben Sie hier die Legacy-ID, die Ihnen beim Abschluss der Buchung in der Response zurückgeliefert wurde. Beispiel: JDNGJDAwJDU2JDAwJDK5JDAwJDFEJDAwJDQxJDAwJDQwJDAw |

Beispiel:

```
/services/api/v1/ratio/reports/generate?bereich=T-TOURISTIK&art=BESTAETIGUNG&name=webformular&format=pdf&id=JDNGJDAwJDU2JDAwJDK5JDAwJDFEJDAwJDQxJDAwJDQwJDAw
```

Das Dokument wird im Response-Body im gewünschten Format zurückgeliefert. Die Daten können Sie direkt per Mail verschicken oder als Datei abspeichern.

Gutscheine

Diese Funktionen dienen zur Entwicklung einer individuellen Gutscheinbestellmaske für Kunden. Um den vollen Funktionsumfang auszuschöpfen, können jedoch weitere Funktionen anderer Bereiche eingesetzt werden, z.B. Registrierung, Finanzbuchhaltung, Reports.

Zahlung über Dienstleister in Standardbuchungsmaske

Wird die Standardbuchungsmaske von Ratio per iFrame in eine Website eingebettet, kann die Zahlung trotzdem über einen beliebigen Zahlungsanbieter abgewickelt werden. Hierzu ist es notwendig in der Grundkonfiguration die Option **gutschein.zahlung_custom_aktiv** auf den Wert **JA** zu setzen.

In diesem Fall hat der Kunde die Möglichkeit eine weitere Zahlungsart auszuwählen. Die Bezeichnung sowie der beschreibende Text können ebenfalls in der Grundkonfiguration bei den Übersetzungen angepasst werden:

- **GUTSCHEIN.VIEW.ZAHLUNG.CUSTOM.LABEL**
- **GUTSCHEIN.VIEW.ZAHLUNG.CUSTOM.TEXT**

Die Buchungsstrecke wird regulär durchlaufen und nach Klick auf *Zahlungspflichtig buchen* ein neuer Gutschein im RatioWw Gutscheinmodul angelegt.

Anschließend werden die relevanten Zahlungsdaten per **postMessage** an das umgebende Parent-Window geschickt. Das MessageEvent enthält in der data-Eigenschaft folgende Werte:

| Name | Typ | Referenz | Beschreibung |
|--------------------|---------------|---------------------|---|
| messageId | STRING | | Identifiziert zur Erkennung des Nachrichtentyps für das Parent-Window. Enthält immer den Wert: ratio-gutschein-custom-zahlung |
| betrag | NUMERIC(15,2) | | Zahlungsbetrag |
| waehrung | VARCHAR(255) | | Währung laut Grundkonfiguration |
| vorgangsnummer | INTEGER | T_VORGANG.NR | Vorgangsnummer |
| successUrl | STRING | | URL zur Fortsetzung der Buchungsstrecke nach einer erfolgreichen Zahlung |
| failureUrl | STRING | | URL zur Fortsetzung der Buchungsstrecke nach einer fehlgeschlagenen Zahlung |
| kundeKennziffer | INTEGER | ADRESSEN.KENNZIFFER | Kennziffer des Kunden |
| kundeAnrede | CHAR(20) | | Anrede |
| kundeTitel | CHAR(20) | | Titel |
| kundeName1 | CHAR(30) | | Vorname |
| kundeName2 | CHAR(30) | | Nachname |
| kundeName3 | CHAR(30) | | Zusatz |
| kundeNamenszusatz | CHAR(60) | | Namenszusatz |
| kundeNamenszusatz2 | CHAR(120) | | Namenszusatz (Verlängerung) |
| kundeStrasse | CHAR(30) | | Straße |
| kundeStrasse2 | VARCHAR(80) | | Straße (Verlängerung) |
| kundePlz | CHAR(15) | | Postleitzahl |
| kundeOrt | CHAR(30) | | Ort |
| kundeOrt2 | CHAR(80) | | Ort (Verlängerung) |
| kundeLand | CHAR(15) | | Ländercode |
| kundeRegion | CHAR(20) | | Region |
| kundeTelefon1 | CHAR(25) | | Telefonnummer 1 |
| kundeTelefon2 | CHAR(25) | | Telefonnummer 2 |
| kundeFax | CHAR(25) | | Faxnummer |
| kundeEmail | CHAR(60) | | E-Mail-Adresse |

Diese Informationen können je nach Bedarf an den angebotenen Zahlungsdienstleister übergeben werden. Im Anschluss an die Zahlungsabwicklung kann entweder die **successUrl** oder **failureUrl** aufgerufen werden. In beiden Fällen wird die Buchung abgeschlossen und die Bestätigungsmail verschickt (sofern aktiv). Bei Aufruf der *failureUrl* wird ein Hinweis über die fehlgeschlagene Zahlung auf der Bestätigungsseite und in der Bestätigungsmail (sofern aktiv) angezeigt.

Es ist nicht zwingend erforderlich eine der beiden URLs aufzurufen. Stattdessen kann auch eine eigene Bestätigungsseite angezeigt und eine eigene Mail versandt werden.

Achtung: Die Zahlung muss in jedem Fall selbst erfasst werden. Dazu kann die API-Funktion **save-zahlung** aus dem Bereich *Finanzbuchhaltung* verwendet werden.