

# API Ostsee-Guide

- [Authentifizierung Via Buchungscode](#)

# Authentifizierung Via Buchungscode

Diese Funktion ist für die "Karte hinzufügen" Funktionalität gedacht.

## Allgemein

- Jeder Gast hat einen eindeutigen Auth-Code, da dieser sich aus der Meldeschein-ID sowie der Gast-ID zusammensetzt
  - Der Auth-Code muss zur Gemeinde ID passen, somit ist es möglich, weitere Gemeinden einfach anzubinden
  - Somit können auch 3 Gäste eines einzelnen Meldescheins eindeutig identifiziert werden
- Auth Codes sind immer 8-stellig
- Auth Codes können nur folgende Zeichen enthalten: `abcdefghijklmnopqrstuvwxyz1234567890`

## Authentifizierung

Die API Endpunkte sind nur authentifiziert und mit gültigem Header: `Authorization: JWT <Wert>` erreichbar.

Mehr zur Authentifizierung finden Sie [hier](#)

## API Endpunkte

Für das Entwicklungs-System `https://dev-api.meldescheine.de/api/buchtguide/by_auth_code/`

Für das Produktiv-System `https://api.meldescheine.de/api/buchtguide/by_auth_code/`

## Request

Das Payload kann als JSON oder Form Encodes übergeben werden.

```
{
  "auth_code": "<string: AuthentifizierungsCode>",
  "gemeinde_id": "<string: GemeindeID>"
}
```

## Beispiel

```
{
  "auth_code": "zkeyfmjz",
  "gemeinde_id": 2
}
```

# Response

## Erfolg

Die Antwort enthält alle relevanten Daten über den Gast und den Meldeschein.

- Meldescheinnummer
- Gastdaten
  - Tarif Bezeichnung sowie interne ID
  - Gastdaten (beschränkt auf die notwendigsten)
- QR Code daten (siehe Abschnitt "QR Code")

Hinweis: Es wird Standardmäßig nur der Gast angefordert, dem der AuthCode zugewiesen wurde.

HTTP Response : 200

```
{
  "auth_code": "19pxsjyx",
  "meldeschein_id": 1133,
  "gemeinde_id": 1,
  "meldeschein": {
```

```
"meldescheinnummer": "1133",
"travel_span": {
  "arrival_date": "2023-07-24T00:00:00",
  "departure_date": "2023-08-06T00:00:00",
  "travel_days": 13,
  "uebernachtungen_sum": 26
},
"gast_data": [
  {
    "id": 2165,
    "tarif_id": 1,
    "tarif_alias": "Erwachsener",
    "Gast_Vorname": "Max",
    "Gast_Name": "Mustermann",
    "Gast_Email": "max@mustermann.de"
  }
],
"qr_code_data": "/9j/4AAQSkZJRgABA...<abgeschnitten>",
"qr_code_encoding": "base64",
"qr_code_mime": "image/jpeg"
}
```

# Alle Gäste des Meldescheins anfordern

Um alle Meldescheine des Meldescheins über einen AuthCode anzufordern muss dem JSON Payload der Parameter `get_all` mit dem boolschen Wert `true` angefügt werden.

```
{
  "auth_code": "19pxsjyx",
  "gemeinde_id": 1,
  "get_all": true  //<---
}
```

Die Rückgabe des Servers entspricht dem oben gezeigten Format, mit dem Unterschied dass das `gast_data` Array **alle** Gäste des Meldescheins enthält.

## QR Code Bilddatei erzeugen

- In den Feldern `qr_code_data`, `qr_code_encoding` sowie `qr_code_mime` sind alle relevanten Informationen enthalten, eine Bilddatei zu erzeugen.
- `qr_code_data` enthält ein `base64` kodierte Bild im JPEG Format (wie im Feld `qr_code_mime` angegeben).

Über das Dekodieren des Wertes `qr_code` Data kann ein JPG Bild generiert werden, welches in der Endanwendung angezeigt werden kann

## Inhalt des QR Codes

Der Inhalt des QR Codes ist der AuthCode als `String` im Klartext.

{QR-Code} -> Inhalt: `eyJhdXRoX2NvZGUiOiAiMTIweHNqeX...` -> Wert des QR Codes: `19pxsjyx`

der QR Code enthält zu Sicherheitszwecken keine Personendaten.

## Fehlerfall

### Auth Code nicht gefunden

HTTP Response : `404`

```
{
  "detail": "Unbekannter Auth Code [<übergebener Auth Code>]",
  "status_code": 404
}
```