

REST API Wiederkehrende Zahlungen

Wiederkehrende Zahlungen, auch Abonnementzahlungen genannt, werden einem Kunden in regelmäßigen Abständen automatisch in Rechnung gestellt.

Implementierungsablauf

Mit Computop können Sie wiederkehrende Zahlungen in nur drei einfachen Schritten einrichten und abrechnen:

1. Erste Transaktion mit wiederkehrender Absicht
2. Anmeldeinformationen sicher speichern
3. Folgende wiederkehrende Transaktionen

1. Erste Transaktion

Richten Sie die Absicht für eine wiederkehrende Zahlung ein, indem Sie die folgenden Informationen im Objekt `credentialOnFile` übergeben:

```
{
  ...
  "credentialOnFile": {
    "type": "RECURRING",
    "initialPayment": true,
    "recurring": {
      "frequency": "DAILY", // "WEEKLY"/"MONTHLY"/"YEARLY"
      "startDate": "2025-01-01",
      "expiryDate": "2025-12-01",
      "useCase": "FIXED", // "FLEXIBLE_AMOUNT"/"FLEXIBLE_FREQUENCY"
    }
  }
}
```

Parameterübersicht

Parameter	Werte	Beschreibung
frequency	DAILY/MONTHLY/WEEKLY /YEARLY	Abrechnungsintervall
startDate	YYYY-MMM-DD	Datum der ersten Abbuchung
expiryDate	YYYY-MMM-DD	Kündigungsdatum
useCase	FIXED/FLEXIBLE_AMOUNT /FLEXIBLE_FREQUENCY	<ul style="list-style-type: none">• FIXED: Gleicher Betrag/gleiches Intervall. Z. B. Abonnements für SaaS-Produkte• FLEXIBLE_AMOUNT: Unterschiedliche Gebühren. Z. B. verbrauchsabhängige Nebenkostenabrechnung• FLEXIBLE_FREQUENCY: Anpassbare Intervalle. Z. B. Abonnements mit der Möglichkeit, die Abrechnungsintervalle im Laufe der Zeit zu ändern.

2. Speicherung der Anmeldeinformationen

Nach Erhalt der erfolgreichen Zahlungsbestätigung für die erste Transaktion werden die Anmeldeinformationen je nach Integrationsart sicher gespeichert:

Integrationsart	Aktion
-----------------	--------

<p>Gehostete Formulare – Gehostete Zahlungsseite or Kreditkarten Gehostete Formulare</p>	<p>Rufen Sie die <code>pseudoCardNumber</code> ab, indem Sie Zahlungsdetails per Payment ID abrufen aufrufen und speichern sie.</p> <pre> { ... "paymentMethods": { "type": "CARD", "card": { "cardholderName": "John Doe", "pseudoCardNumber": "01234567890124444", "first6Digits": 555555, "last4Digits": 4444, "expiryDate": "01.01.2028", "schemeReferenceId": "4234234234", } } } </pre>
<p>Direkte Integration</p>	<p>Speichern Sie entweder:</p> <ul style="list-style-type: none"> • sicher eindeutige Kartendaten, die Sie bereits während des Zahlungsvorgangs haben. • die in der Zahlungsantwort erhaltene <code>pseudoCardNumber</code>, falls Sie den Dienst aktiviert haben.

3. Nachfolgende Transaktionen

Nachfolgende Transaktionen für einen ungeplanten MIT sollten immer über die direkte Integration gesendet werden.

- Übergeben Sie die folgenden Daten im Objekt `credentialOnFile` für nachfolgende wiederkehrende Transaktionen:

```

{
  ...
  "credentialOnFile": {
    "type": "RECURRING",
    "initialPayment": false,
    "recurring": {
      "frequency": "daily", // "weekly"/"monthly"/"yearly"
      "startDate": "2025-01-01",
      "expiryDate": "2025-12-01",
      "useCase": "FIXED", // "FLEXIBLE_AMOUNT"/"FLEXIBLE_FREQUENCY"
    }
  }
}

```

- Übergeben Sie zusätzlich zu den obigen Daten die gespeicherten Anmeldeinformationen und die `schemeReferenceId`, die zuvor in der Antwort der ersten Transaktion zurückgegeben wurden:

```
{
  ...
  "paymentMethods": {
    "type": "CARD",
    "card": {
      "expiryDate": "202506",
      "cardHolderName": "John Doe",
      "securityCode": "123",
      "number": "5555555555554444", // clear card number or pseudo card number
      "schemeReferenceId": "4234234234"
      .....
      .....
    }
  }
}
```