

# REST API Payment with Recurring Mandates

Recurring mandates allow you to debit a customer's bank account repeatedly without requiring the customer to approve each payment individually. This is a common pattern for subscription models, one-click checkouts and adhoc merchant-initiated debits.

Computop Paygate supports recurring mandates for the two main scenarios:

- **Unscheduled Customer-Initiated Transactions (CIT):** One-click payments where the customer initiates payment manually but does not have to re-enter payment information.
- **Unscheduled Merchant-Initiated Transactions (MIT) & Recurring Payments:** Payments initiated by you without any customer interaction, such as in the case of subscriptions or ad-hoc charges.

## Implementation workflow

You can setup recurring mandates with Computop by following 3 simple steps:

1. Initial transaction with recurring intent
2. Store credentials securely
3. Subsequent transactions

### 1. Initial transaction

Setup intent for an recurring by passing the below information in the `credentialOnFile` object:

```
{
  ...
  "credentialOnFile": {
    "type": "RECURRING",
    "initialPayment": true
  }
}
```

### 2. Credential storage

Store the account and mandate details based on the integration type upon receiving successful payment response to the initial transaction:

Integration type	Action
Hosted forms – <a href="#">Hosted Payment Page</a> or Direct Debit <a href="#">Hosted forms</a>	Fetch the account and mandate details by calling <a href="#">Retrieve payment details by payment ID</a> and store it <pre>{   ...   "paymentMethods": {     "type": "DIRECTDEBIT",     "directDebit": {       ....       "mandate": {         "mandateId": "SEPA123456789",         "dateOfSignature": "15.02.2025"       },       "account": {         "code": "DE123456789",         "number": "DE89370400440532013000",         "accountHolderName": "John Doe",         "bankName": "National Bank"       }     }   } }</pre>

Direct integration	Store the account and mandate details that you already have during the payment process.
--------------------	---

### 3. Subsequent transactions

Submit the subsequent payment request with request parameters depending on the type of your integration

Integration type	Payment parameters
Hosted forms - Hosted Payment Page or card hosted forms	<ul style="list-style-type: none"> <li>Submit the <code>credentialOnFile</code> object as below: <pre> {   ...   "credentialOnFile": {     "type": "RECURRING",     "initialPayment": false   } } </pre> </li> <li>Pass the account details in <code>prefillInfo</code> object and mandate details in <code>mandate</code> object, returned previously in the response of initial transaction, for a faster checkout <pre> {   "paymentMethods": {     "type": "DIRECTDEBIT",     "directDebit": {       "mandate": {         "mandateId": "SEPA123456789",         "dateOfSignature": "15.02.2025"       },     },     "prefillInfo": {       "account": {         "code": "BYLADEMLKMD",         "number": "DE89370400440532013000",         "accountHolderName": "John Doe",         "bankName": "National Bank"       }     }   } } </pre> </li> </ul>

## Direct Integration

- Submit the `credentialOnFile` object as below:

```
{
  ...
  "credentialOnFile": {
    "type": "RECURRING",
    "initialPayment": false
  }
}
```

- Pass the account and mandate details as below:

```
{
  "paymentMethods": {
    "type": "DIRECTDEBIT",
    "directDebit": {
      "mandate": {
        "mandateId": "SEPA123456789",
        "dateOfSignature": "15.02.2025"
      },
      "account": {
        "code": "BYLADEM1KMD",
        "number": "DE89370400440532013000",
        "accountHolderName": "John Doe",
        "bankName": "National Bank"
      }
    }
  }
}
```