Bill Pay API Contract

Table of Contents

[1. Enrollment Flow 3](#_Toc31898465)

[1.1 Get Customer Status 3](#_Toc31898466)

[1.2 Add Customer 7](#_Toc31898467)

[1.3 Get Customer Info 9](#_Toc31898468)

[2. Manage Payees 9](#_Toc31898469)

[2.1 Get Payee List 9](#_Toc31898470)

[2.2 Get Payee by Id 12](#_Toc31898471)

[2.3 Add Unmanaged Payee 13](#_Toc31898472)

[2.4 Edit Unmanaged Payee 17](#_Toc31898473)

[2.5 Add Managed Payee 20](#_Toc31898474)

[2.6 Edit Managed Payee 23](#_Toc31898475)

[2.7 Delete Payee 25](#_Toc31898476)

[2.8 Get Merchants 26](#_Toc31898477)

[3. Discover Credit Cards 27](#_Toc31898478)

[3.1 Get Discover Cards 27](#_Toc31898479)

[3.2 Add Discover Cards 28](#_Toc31898480)

[4. Payment Reminders 30](#_Toc31898481)

[4.1 Set Reminder 30](#_Toc31898482)

[4.2 Delete Reminder 31](#_Toc31898483)

[5. Payment Operations 32](#_Toc31898484)

[5.1 Schedule a One-Time Payment 32](#_Toc31898485)

[5.2 Set a Repeating Payment **(For both Add and Edit flow)** 35](#_Toc31898486)

[5.3 Schedule an E-Bill One-Time Payment 37](#_Toc31898487)

[**5.4** Set an E-Bill Automatic Payment **(For both Add and Edit flow)** 38](#_Toc31898488)

[5.5 Edit Any One-Time Payment (Including a Single Payment from a Repeating Rule and Bill Payment) 40](#_Toc31898489)

[5.6 Delete Any One-Time Payment (Including a Single Payment from a Repeating Rule and Bill Payment) 43](#_Toc31898490)

[5.7 Delete a Series of Repeating Payments 44](#_Toc31898491)

[5.8 Delete E-Bill Automatic Payment 45](#_Toc31898492)

[6. Review Payments 45](#_Toc31898493)

[6.1 Get Payment List 45](#_Toc31898494)

[6.2 Get Scheduled Payment List for a Payee 48](#_Toc31898495)

[6.3 Search Payments 50](#_Toc31898496)

[6.4 Repeating Payment History 52](#_Toc31898497)

[7. Ebill Operations 54](#_Toc31898498)

[7.1 Get E-Bill Tokens 54](#_Toc31898499)

[7.2 E-Bill Enrollment 55](#_Toc31898500)

[7.3 E-Bill Un-enroll 59](#_Toc31898501)

[7.4 Get E-Bill Details 60](#_Toc31898502)

[7.5 File E-Bill (Mark as Paid) 61](#_Toc31898503)

[8. Bill Pay Utility Methods 62](#_Toc31898504)

[8.1 Get Bank Holiday List 62](#_Toc31898505)

[8.2 Get Payment Accounts 63](#_Toc31898506)

# Enrollment Flow

This flow checks and prompts customer’s bill pay enrollment.

## Get Customer Status

|  |  |
| --- | --- |
| **Endpoint Name –** GetCustomerStatus | |
| **Description –** Get customer’s bill pay enrollment and eligibility status. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/customer/v1/enroll | |
| **Request Type –** GET | |
| **Scenario 1** Description: User has eligible accounts and is already enrolled into bill pay | |
| Request Body: | Response Body: |
| {} | **Example 1 (no payees):**  {  "eligible": true,  "fundsAvailable": true,  "enrolledInFIS": true,  "enrolledInDFS": true,  "payees": [],  "\_links": {  "terms": {  "href": "https://asys-mapi.discoverbank.com/mobile/bank/payments/terms.html"  }  }  }  enrolledInFIS – Customer is enrolled in new service.  enrolledInDFS – Customer is enrolled in DFS database. |
| Request Body: | Response Body: |
| {} | **Example 2 (has payees):**  {  "eligible": true,  "fundsAvailable": true,  "enrolledInFIS": true,  "enrolledInDFS": true,  "payees": [  {  "id": "102276950",  "name": "Cox Communications",  "nickName": "cox com",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*7987",  "address": {  "postalCode": "850628322"  },  "verified": true,  "status": "ACTIVE",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "earliestPaymentDate": "2020-02-24",  "leadDays": 2,  "cutOffTime": "15:00:00",  "reminderEligible": false,  "ebill": {  "id": "208241150",  "type": "DIRECT",  "status": "UNPAID",  "paymentURIDetails": "https://bpm-uat.readiness.billdomain.com/csp/ShowPBillServlet  ?brid=7742&token=98DD2061409&Pg=1",  "cycleStatus": "IN\_CYCLE",  "payeeId": "102276950",  "dueDate": "2020-02-07",  "statementDate": "2020-01-08",  "receivedDate": "2020-01-28",  "totalDue": 117.95,  "minimumDue": 22  },  "ebillStatus": "ACTIVE"  },  {  "id": "102281508",  "name": "Cox Communications",  "nickName": "ebill cox",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*7911",  "address": {  "postalCode": "850628322"  },  "verified": true,  "status": "ACTIVE",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "earliestPaymentDate": "2020-02-24",  "leadDays": 2,  "cutOffTime": "15:00:00",  "reminderEligible": true,  "ebillStatus": "AVAILABLE"  },  {  "id": "102276953",  "name": "George",  "nickName": "Washington",  "accountNumber": "\*2345",  "phoneNumber": {  "number": "7738900987",  "formatted": "773-890-0987"  },  "address": {  "streetAddress": "4846 wright terrace",  "extendedAddress": "apt 12345",  "locality": "skokie",  "region": "IL",  "postalCode": "60077",  "formatted": "4846 wright terrace apt 12345 \nskokie IL 60077\nUSA"  },  "verified": false,  "status": "ACTIVE",  "deliveryMethod": "DIRECT\_CHECK",  "earliestPaymentDate": "2020-02-24",  "leadDays": 4,  "cutOffTime": "15:00:00",  "reminderEligible": false,  "repeatingPayment": {  "ruleId": "8831899",  "frequency": "ONCE\_A\_WEEK",  "amount": 10.01,  "startDate": "2020-02-11",  "nextPaymentDate": "2020-03-10"  },  "ebillStatus": "NOT\_AVAILABLE"  }  ],  "\_links": {  "terms": {  "href": "https://tsys-mapi.discoverbank.com/mobile/bank/payments/terms.html"  }  }  } |
| *Result:* User has bill pay eligible accounts and is enrolled in bill pay | |
| **Scenario 2** Description: User has eligible accounts and is not enrolled in bill pay. | |
| Request Body: | Response Body: |
| {} | {  "eligible": true,  "fundsAvailable": true,  "enrolledInFIS": false,  "enrolledInDFS": false,  "payees": [],  "\_links": {  "terms": {  "href": "https://asys-mapi.discoverbank.com/mobile/bank/payments/terms.html"  }  }  } |
| *Result:* User has bill pay eligible accounts and is NOT enrolled in bill pay | |
| **Scenario 3** Description: User does not have eligible accounts and is not enrolled in bill pay. | |
| Request Body: | Response Body: |
| {} | {  "eligible": false,  "fundsAvailable": false,  "enrolledInFIS": false,  "enrolledInDFS": false,  "payees": [],  "\_links": {  "terms": {  "href": "https://asys-mapi.discoverbank.com/mobile/bank/payments/terms.html"  }  }  } |
| *Result:* User does NOT have bill pay eligible accounts and is NOT enrolled in bill pay. Show tech-diff. | |
| **Scenario 4** Description: User does not have eligible accounts and is enrolled in bill pay. | |
| Request Body: | Response Body: |
| {} | {  "eligible": false,  "fundsAvailable": false,  "enrolledInFIS": true,  "enrolledInDFS": true,  "payees": [],  "\_links": {  "terms": {  "href": "https://asys-mapi.discoverbank.com/mobile/bank/payments/terms.html"  }  }  } |
| *Result:* User does NOT have bill pay eligible accounts and is enrolled in bill pay (i.e. eligible account was removed). | |
| **Scenario 5** Description: User has eligible accounts without positive balance and is not enrolled in bill pay. | |
| Request Body: | Response Body: |
| {} | {  "eligible": true,  "fundsAvailable": false,  "enrolledInFIS": false,  "enrolledInDFS": false,  "payees": [],  "\_links": {  "terms": {  "href": "https://asys-mapi.discoverbank.com/mobile/bank/payments/terms.html"  }  }  } |
| *Result:* User has bill pay eligible accounts with insufficient balance and is NOT enrolled in bill pay. | |
| **Scenario 6** Description: User has eligible accounts and is enrolled in bill pay but not enrolled in DFS. | |
| Request Body: | Response Body: |
| {} | {  "eligible": true,  "fundsAvailable": true,  "enrolledInFIS": true,  "enrolledInDFS": false,  "payees": [],  "\_links": {  "terms": {  "href": "https://asys-mapi.discoverbank.com/mobile/bank/payments/terms.html"  }  }  } |
| *Result:* User has bill pay eligible accounts and is enrolled in FIS and not in DFS (migrated customer). Show New T&C page. | |
| **Scenario 7** Description: General error. In case of any service error. | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "General.TechnicalDifficulties",  "httpStatus": 500,  "message": "Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information."  }  ]  } |
| *Result:* An error was encountered. (i.e. invalid user accessed a bill pay page) | |

## Add Customer

|  |  |
| --- | --- |
| **Endpoint Name –** AddCustomer | |
| **Description –** Add customer to FIS bill pay system. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/customer/v1/enroll | |
| **Request Type –** POST | |
| **Scenario 1** Description: User is successfully enrolled into bill pay system by accepting Terms and Conditions. | |
| Request Body: | Response Body: |
| {} | **Example 1 (no payees):**  {  "eligible": true,  "fundsAvailable": true,  "enrolledInFIS": true,  "enrolledInDFS": true,  "payees": []  }  enrolledInFIS – Customer is enrolled in new service.  enrolledInDFS – Customer is enrolled in DFS database. |
| Request Body: | Response Body: |
|  | **Example 2 (has payees):**  {  "eligible": true,  "fundsAvailable": true,  "enrolledInFIS": true,  "enrolledInDFS": true,  "payees": [  {  "id": "102281508",  "name": "Cox Communications",  "nickName": "ebill cox",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*7911",  "address": {  "postalCode": "850628322"  },  "verified": true,  "status": "ACTIVE",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "earliestPaymentDate": "2020-02-24",  "leadDays": 2,  "cutOffTime": "15:00:00",  "reminderEligible": true,  "ebillStatus": "AVAILABLE"  },  {  "id": "102276953",  "name": "George",  "nickName": "Washington",  "accountNumber": "\*2345",  "phoneNumber": {  "number": "7738900987",  "formatted": "773-890-0987"  },  "address": {  "streetAddress": "4846 wright terrace",  "extendedAddress": "apt 12345",  "locality": "skokie",  "region": "IL",  "postalCode": "60077",  "formatted": "4846 wright terrace apt 12345 \nskokie IL 60077\nUSA"  },  "verified": false,  "status": "ACTIVE",  "deliveryMethod": "DIRECT\_CHECK",  "earliestPaymentDate": "2020-02-24",  "leadDays": 4,  "cutOffTime": "15:00:00",  "reminderEligible": true,  "ebillStatus": "NOT\_AVAILABLE"  }  ]  } |
| *Result:* User has bill pay eligible accounts and is successfully enrolled in bill pay | |
| **Scenario 2** Description: General error. In case of any service error. | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "General.TechnicalDifficulties",  "httpStatus": 500,  "message": "Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information."  }  ]  } |
| *Result:* An error was encountered during enrollment. User was not enrolled. | |

## Get Customer Info

|  |  |
| --- | --- |
| **Endpoint Name –** GetCustomerInfo | |
| **Description –** Get customer’s personal information | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/customer/v1/profile | |
| **Request Type –** GET | |
| **Scenario 1** Description: User’s personal information is requested. | |
| Request Body: | Response Body: |
| {} | {  "firstName": "SHEILA",  "lastName": "RINGHISER",  "email": "CUSTOMER95274231@CUSTOMER.COM"  } |
| *Result:* User’s personal information is successfully returned. | |
| **Scenario 2** Description: Service Error occurred. | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "General.TechnicalDifficulties",  "httpStatus": 500,  "message": "Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information."  }  ]  } |

# 2. Manage Payees

This module is used to manage all user’s payees.

## Get Payee List

|  |  |
| --- | --- |
| **Endpoint Name –** GetPayees | |
| **Description –** Get all customer’s payees. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payee/v1/ | |
| **Request Type –** GET | |
| **Scenario 1** Description: User has payees. A list of payee details is successfully returned. List is sorted by payee nickname. | |
| Request Body: | Response Body: |
| {} | [  {  "id": "102276950",  "name": "Cox Communications",  "nickName": "ebill cox",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*7987",  "address": {  "postalCode": "850628322"  },  "verified": true,  "status": "ACTIVE",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "earliestPaymentDate": "2020-02-24",  "leadDays": 2,  "cutOffTime": "15:00:00",  "reminderEligible": false,  "automaticPayment": {  "paymentType": "PAY\_FULL\_BALANCE",  "sendPaymentType": "ON\_DUE\_DATE"  },  "ebill": {  "id": "208241150",  "type": "DIRECT",  "status": "UNPAID",  "paymentURIDetails": "https://bpm-uat.readiness.billdomain.com/csp/ShowPBillServlet  ?brid=7742&token=98DD206140D&Pg=1",  "cycleStatus": "IN\_CYCLE",  "payeeId": "102276950",  "dueDate": "2020-02-07",  "statementDate": "2020-01-08",  "receivedDate": "2020-01-28",  "totalDue": 117.95,  "minimumDue": 22  },  "ebillStatus": "ACTIVE"  },  {  "id": "102281373",  "name": "Discover Credit Card",  "nickName": "Discover Credit Card 3339",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*3339",  "address": {  "postalCode": "601976103"  },  "verified": true,  "status": "ACTIVE",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "earliestPaymentDate": "2020-02-24",  "leadDays": 2,  "cutOffTime": "15:00:00",  "reminderEligible": true,  "reminder": {  "frequency": "ONCE\_IN\_2\_WEEKS",  "amount": 21.4,  "date": "2020-03-09"  },  "cardPaymentInfo": {  "statementBalance": 137.58,  "currentBalance": 137.58,  "minimumBalance": 0,  "dueDate": "2020-09-28"  },  "ebillStatus": "AVAILABLE"  },  {  "id": "102276953",  "name": "Alexander",  "nickName": "Hamilton",  "accountNumber": "\*2345",  "phoneNumber": {  "number": "7738900987",  "formatted": "773-890-0987"  },  "address": {  "streetAddress": "4846 wright terrace",  "extendedAddress": "apt 12345",  "locality": "skokie",  "region": "IL",  "postalCode": "60077",  "formatted": "4846 wright terrace apt 12345 \nskokie IL 60077\nUSA"  },  "verified": false,  "status": "ACTIVE",  "deliveryMethod": "DIRECT\_CHECK",  "earliestPaymentDate": "2020-02-24",  "leadDays": 4,  "cutOffTime": "15:00:00",  "reminderEligible": false,  "repeatingPayment": {  "ruleId": "8831899",  "frequency": "ONCE\_A\_WEEK",  "amount": 10.01,  "startDate": "2020-02-11",  "nextPaymentDate": "2020-03-10"  },  "ebillStatus": "NOT\_AVAILABLE"  } |
| *Result:* User has payees. Three Managed Payees and one Unmanaged Payee.  For a managed payee, verified flag is set to true and only postalCode is provided. | |
| ***Additional Information***  reminderEligible – payee is eligible for reminders when this value is true.  ebillStatus – Provides payee e-bill status information with the following status values:   * AVAILABLE – Payee eligible for e-bill enrollment. * ACTIVE – Payee is already enrolled in e-bill. * NOT\_AVAILABLE – Payee is not eligible for e-bill enrollment. * PENDING\_ACTIVATION – E-bill activation is pending. * REQUEST\_REJECTED – E-bill payee rejected a change to the e-bill status. * VALIDATION\_FAILED – E-bill enrollment validation failed. | |
| **Scenario 2** Description: User does NOT have any payees. | |
| Request Body: | Response Body: |
| {} | [] |
| *Result:* An empty list is returned. | |
| **Scenario 3** Description: Service Error occurred. | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "General.TechnicalDifficulties",  "httpStatus": 500,  "message": "Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information."  }  ]  } |

## Get Payee by Id

|  |  |
| --- | --- |
| **Endpoint Name –** GetPayeeByID | |
| **Description –** View a customer’s specific payee. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payee/v1/{payeeID} | |
| **Request Type –** GET | |
| **Scenario 1** Description: Payee details are successfully found for requested payee ID. | |
| Request Body: | Response Body: |
| {} | {  "id": "102272277",  "name": "Alexander",  "nickName": "Hamilton",  "accountNumber": "\*\*3123",  "phoneNumber": {  "number": "2244056873",  "formatted": "224-405-6873"  },  "address": {  "streetAddress": "2500 lake cook rd",  "locality": "Riverwoods",  "region": "IL",  "postalCode": "60015",  "formatted": "2500 lake cook rd \nRiverwoods IL 60015\nUSA"  },  "verified": false,  "status": "ACTIVE",  "deliveryMethod": "DIRECT\_CHECK",  "earliestPaymentDate": "2019-12-24",  "leadDays": 4,  "cutOffTime": "15:00:00",  "reminderEligible": true,  "reminder": {  "frequency": "ONCE\_IN\_2\_WEEKS",  "amount": 1,  "date": "2019-12-13"  },  "ebillStatus": "NOT\_AVAILABLE"  } |
| *Result:* Payee ID matches and respective details are returned. | |
| **Scenario 2** Description: Service Error occurred | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "General.TechnicalDifficulties",  "httpStatus": 500,  "message": "Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information."  }  ]  } |

## Add Unmanaged Payee

|  |  |
| --- | --- |
| **Endpoint Name –** AddUnmanagedPayee | |
| **Description –** Add an unmanaged payee for a customer. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payee/v1/ | |
| **Request Type –** POST | |
| **Scenario 1** Description: Unmanaged payee without account number is successfully added for a user. | |
| Request Body: | Response Body: |
| {  "name": "George",  "nickName": "Washington",  "phoneNumber":{  "number": "7731215533"  },  "address": {  "streetAddress": "4846 Wright Terr",  "locality": "Skokie",  "region": "IL",  "postalCode": "60077"  },  "verified":false  } | {  "id": "102274913",  "name": " George",  "nickName": " Washington",  "phoneNumber": {  "number": "7731215533",  "formatted": "773-121-5533"  },  "address": {  "streetAddress": "4846 Wright Terr",  "locality": "Skokie",  "region": "IL",  "postalCode": "60077",  "formatted": "4846 Wright Terr \nSkokie IL 60077\nUSA"  },  "verified": false,  "status": "ACTIVE",  "deliveryMethod": "DIRECT\_CHECK",  "earliestPaymentDate": "2019-12-24",  "leadDays": 4,  "cutOffTime": "15:00:00",  "reminderEligible": true,  "ebillStatus": "NOT\_AVAILABLE"  } |
| *Result:* Unmanaged payee details are returned. | |
| **Scenario 2** Description: Unmanaged payee with account number is successfully added for a user. | |
| Request Body: | Response Body: |
| {  "name": "Abraham",  "nickName": "Lincoln",  "accountNumber": "1234",  "phoneNumber":{  "number": "7734445522"  },  "address": {  "streetAddress": "4840 Wright Terr",  "locality": "Skokie",  "region": "IL",  "postalCode": "60077"  },  "verified":false  } | {  "id": "102274970",  "name": "Abraham",  "nickName": "Lincoln",  "accountNumber": "\*\*34",  "phoneNumber": {  "number": "7734445522",  "formatted": "773-444-5522"  },  "address": {  "streetAddress": "4840 Wright Terr",  "locality": "Skokie",  "region": "IL",  "postalCode": "60077",  "formatted": "4840 Wright Terr \nSkokie IL 60077\nUSA"  },  "verified": false,  "status": "ACTIVE",  "deliveryMethod": "DIRECT\_CHECK",  "earliestPaymentDate": "2019-12-24",  "leadDays": 4,  "cutOffTime": "15:00:00",  "reminderEligible": true,  "ebillStatus": "NOT\_AVAILABLE"  } |
| *Result:* Unmanaged payee details are returned. | |
| **Scenario 3** Description: User tries to add payee with duplicate nickname. (Duplicate Payee Nickname Found) | |
| Request Body: | Response Body: |
| {  “name”: “Me, A Name, I Call Myself”,  “nickName”: “Ray, a Drop of Golden Sun!”,  “phoneNumber”:{  “number”: “7734445522”  },  “address”: {  “streetAddress”: “4840 Wright Terr”,  “locality”: “Skokie”,  “region”: “IL”,  “postalCode”: “60077”  },  “verified”:false  } | {  "errors": [  {  "code": "Payee.Nickname.Exist",  "httpStatus": 500,  "message": "Nickname already exists. Enter a new Nickname.",  "name": "nickName"  }  ]  } |
| *Result: User tries to add payee with duplicate nickname.* | |
| **Scenario 4** Description: User tries to add an existing payee. (Duplicate Payee Found) | |
| Request Body: | Response Body: |
| {  “name”: “John Davidson”,  “nickName”: “Rockefeller”,  “accountNumber”: “555555555”,  “phoneNumber”:{  “number”: “7734445522”  },  “address”: {  “streetAddress”: “4840 Wright Terr”,  “locality”: “Skokie”,  “region”: “IL”,  “postalCode”: “60077”  },  “verified”:false  } | {  “errors”: [  {  “code”: “Payee.Duplicate”,  “httpStatus”: 500,  “message”: “Payee Already Exists.”  }  ]  } |
| *Result:* Payee Exists error sent in response. | |
| **Scenario 5** Description: Invalid Inputs. | |
| Request Body: | Response Body: |
| {  “name”: “>”,  “nickName”: “<”,  “accountNumber”: “-“,  “phoneNumber”:{  “number”: “j”  },  “address”: {  “streetAddress”: “/”,  “locality”: “?”,  “region”: “\_”,  “postalCode”: “:”  },  “verified”:false  } | {  "errors": [  {  "code": "Payee.AccountNumber.Invalid",  "httpStatus": 500,  "message": "Invalid Account Number.",  "name": "accountNumber"  },  {  "code": "Payee.PhoneNumber.Invalid",  "httpStatus": 500,  "name": "phoneNumber.number"  },  {  "code": "Payee.Nickname.Invalid",  "httpStatus": 500,  "message": "Invalid Nickname",  "name": "nickName"  },  {  "code": "Payee.Name.Invalid",  "httpStatus": 500,  "message": "Invalid Name",  "name": "name"  },  {  "code": "Payee.Address.StreetAddress.Invalid",  "httpStatus": 500,  "message": "Invalid Street Address.",  "name": "address.streetAddress"  },  {  "code": "Payee.Address.Locality.Invalid",  "httpStatus": 500,  "message": "Invalid City",  "name": "address.locality"  },  {  "code": "Payee.Address.PostalCode.Invalid",  "httpStatus": 500,  "message": "Invalid Zip Code",  "name": "address.postalCode"  }  ]  } |
| *Result:* Invalid field input errors are sent. | |
| **Scenario 6** Description: Missing Input fields. | |
| Request Body: | Response Body: |
| {      “accountNumber”: “PT007”,    “address”: {  “streetAddress”: “5001 Wright Terr”,  “region”: “IL”,  “postalCode”: “60077”  },  “verified”:false  } | {  "errors": [  {  "code": "Payee.PhoneNumber.NotEmpty",  "httpStatus": 500,  "message": "Phone Number cannot be empty.",  "name": "phoneNumber"  },  {  "code": "Payee.Address.Locality.NotEmpty",  "httpStatus": 500,  "message": "City cannot be blank.",  "name": "address.locality"  },  {  "code": "Payee.Nickname.NotEmpty",  "httpStatus": 500,  "message": "Nickname cannot be blank.",  "name": "nickName"  },  {  "code": "Payee.Name.NotEmpty",  "httpStatus": 500,  "message": "Payee Name is empty.",  "name": "name"  }  ]  } |
| *Result:* Missing input errors are sent. | |
| **Scenario 7** Description: Service error occurred. | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "General.TechnicalDifficulties",  "httpStatus": 500,  "message": "Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information."  }  ]  } |

## Edit Unmanaged Payee

|  |  |
| --- | --- |
| **Endpoint Name –** EditUnmanagedPayee | |
| **Description –** Edit an unmanaged payee for a customer. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payee/v1/update/{payeeID} | |
| **Request Type –** POST | |
| **Scenario 1** Description: Name, account number, phone number, extended address and city fields of an unmanaged payee are successfully edited. | |
| Request Body: | Response Body: |
| {  “name”:”PostTest01”,  “accountNumber”: “12345”,  “phoneNumber”: {  “number”: “8471234567”  },  “address”: {  “extendedAddress”: “APT 201”,  “locality”: “Niles”  },  “verified”: false  }  Note: All fields are not required in edit.  Only send fields that were updated by the user. | {  "id": "102275195",  "name": "PostTest01",  "nickName": "PT00",  "accountNumber": "\*2345",  "phoneNumber": {  "number": "8471234567",  "formatted": "847-123-4567"  },  "address": {  "streetAddress": "4840 Wright Terr",  "extendedAddress": "APT 201",  "locality": "Niles",  "region": "IL",  "postalCode": "60077",  "formatted": "4840 Wright Terr APT 201 \nNiles IL 60077\nUSA"  },  "verified": false,  "status": "ACTIVE",  "deliveryMethod": "DIRECT\_CHECK",  "earliestPaymentDate": "2019-12-24",  "leadDays": 4,  "cutOffTime": "15:00:00",  "reminderEligible": true,  "ebillStatus": "NOT\_AVAILABLE"  } |
| *Result:* Updated unmanaged payee details are returned. | |
| **Scenario 2** Description: Extended address and account number fields of an unmanaged payee are successfully removed. | |
| Request Body: | Response Body: |
| {  “accountNumber”: “”,  “address”: {  “extendedAddress”: “”  },  “verified”: false  }  Note: An empty String is used to remove these field values. This is applicable for optional fields only.  For unmanaged payee account number and address line 2 (extended address) is optional. | {  "id": "102275195",  "name": "PostTest01",  "nickName": "PT00",  "phoneNumber": {  "number": "8471234567",  "formatted": "847-123-4567"  },  "address": {  "streetAddress": "4840 Wright Terr",  "locality": "Niles",  "region": "IL",  "postalCode": "60077",  "formatted": "4840 Wright Terr \nNiles IL 60077\nUSA"  },  "verified": false,  "status": "ACTIVE",  "deliveryMethod": "DIRECT\_CHECK",  "earliestPaymentDate": "2019-12-24",  "leadDays": 4,  "cutOffTime": "15:00:00",  "reminderEligible": true,  "ebillStatus": "NOT\_AVAILABLE"  } |
| *Result:* Updated unmanaged payee details are returned. | |
| **Scenario 3** Description: Invalid Inputs. | |
| Request Body: | Response Body: |
| {    “nickname”: “I”,  “accountNumber”:”0”,  “phoneNumber”:{  “number”: “2”  },  “address”: {  “streetAddress”: “2”,  “locality”: “2”,    “postalCode”: “5”  },  “verified”:false  } | {  "errors": [  {  "code": "Payee.Address.PostalCode.Invalid",  "httpStatus": 500,  "message": "Invalid Zip Code",  "name": "address.postalCode"  },  {  "code": "Payee.Address.StreetAddress.Invalid",  "httpStatus": 500,  "message": "Invalid Street Address.",  "name": "address.streetAddress"  },  {  "code": "Payee.AccountNumber.Invalid",  "httpStatus": 500,  "message": "Invalid Account Number.",  "name": "accountNumber"  },  {  "code": "Payee.Address.Locality.Invalid",  "httpStatus": 500,  "message": "Invalid City",  "name": "address.locality"  },  {  "code": "Payee.PhoneNumber.Invalid",  "httpStatus": 500,  "name": "phoneNumber.number"  }  ]  } |
| *Result:* Invalid input errors are sent. | |
| **Scenario 4** Description: User tries to add payee with duplicate nickname. (Duplicate Payee Nickname Found) | |
| Request Body: | Response Body: |
| {  “nickname”: “ PT11”,  “verified”:false  } | {  "errors": [  {  "code": "Payee.Nickname.Exist",  "httpStatus": 500,  "message": "Nickname already exists. Enter a new Nickname.",  "name": "nickName"  }  ]  } |
| *Result:* User tries to add payee with duplicate nickname. | |
| **Scenario 5** Description: User tries to add an existing payee. (Duplicate Payee Found) | |
| Request Body: | Response Body: |
| {  "name": "Langston Hughes ",  "accountNumber": "12",  "address": {  "streetAddress": "4846 Wright Terr",  "extendedAddress":"English B"  },  "verified":false  } | {  “errors”: [  {  “code”: “Payee.Duplicate”,  “httpStatus”: 500,  “message”: “Payee Already Exists.”  }  ]  } |
| *Result:* Payee Exists error sent in response. | |
| **Scenario 6** Description: Service error occurred. | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "General.TechnicalDifficulties",  "httpStatus": 500,  "message": "Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information."  }  ]  } |

## Add Managed Payee

|  |  |
| --- | --- |
| **Endpoint Name –** AddManagedPayee | |
| **Description –** Add a managed payee for a customer. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payee/v1/ | |
| **Request Type –** POST | |
| **Scenario 1** Description: Managed payee without zip-code is successfully added for a user. | |
| Request Body: | Response Body: |
| {  "name": "Discover Credit Cards",  "nickName": "disco card!",  "accountNumber": "6011208974009999",  "verified":true  } | {  "id": "102275228",  "name": "Discover Credit Cards",  "nickName": "disco card!",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*9999",  "verified": true,  "status": "ACTIVE",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "earliestPaymentDate": "2019-12-24",  "leadDays": 2,  "cutOffTime": "15:00:00",  "reminderEligible": true,  "ebillStatus": "AVAILABLE"  } |
| *Result:* Managed payee details are returned. | |
| **Scenario 2** Description: Managed payee with zip-code is successfully added for a user. | |
| Request Body: | Response Body: |
| {  "name": "Salem Five Bank",  "nickName": "salem biller",  "accountNumber": "0000000005500456",  "address": {  "postalCode": "019700940"  },  "verified":true  } | {  "id": "102275196",  "name": "Salem Five Bank",  "nickName": "salem biller",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*0456",  "address": {  "postalCode": "019700940"  },  "verified": true,  "status": "ACTIVE",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "earliestPaymentDate": "2019-12-24",  "leadDays": 2,  "cutOffTime": "15:00:00",  "reminderEligible": true,  "ebillStatus": "AVAILABLE"  } |
| *Result:* Managed payee details are returned. | |
| **Scenario 3** Description: User tries to add an existing payee. (Duplicate Payee Found) | |
| Request Body: | Response Body: |
| {  “name”: “Salem Five Bank”,  “nickname”: “salem payee”,  “accountNumber”: “0000000005500001”,  “address”: {  “postalCode”: “019700940”  },  “verified”:true  } | {  “errors”: [  {  “code”: “Payee.Duplicate”,  “httpStatus”: 500,  “message”: “Payee Already Exists.”  }  ]  } |
| *Result:* User tries to add duplicate payee. | |
| **Scenario 4** Description: User tries to add payee with duplicate nickname. (Duplicate Payee Nickname Found) | |
| Request Body: | Response Body: |
| {  “name”: “Salem Five Bank”,  “nickname”: “salem payee”,  “accountNumber”: “0000000005500001”,  “address”: {  “postalCode”: “019700940”  },  “verified”:true  } | {  "errors": [  {  "code": "Payee.Nickname.Exist",  "httpStatus": 500,  "message": "Nickname already exists. Enter a new Nickname.",  "name": "nickName"  }  ]  } |
| *Result:* User tries to add payee with duplicate nickname. | |
| **Scenario 5** Description: Missing and invalid inputs sent in the request. | |
| Request Body: | Response Body: |
| {    “nickName”: “$”,  “accountNumber”: “-“,  “address”: {  “postalCode”: “1234”  },  “verified”:true  } | {  "errors": [  {  "code": "Payee.Name.NotEmpty",  "httpStatus": 500,  "message": "Payee Name is empty.",  "name": "name"  },  {  "code": "Payee.Nickname.Invalid",  "httpStatus": 500,  "message": "Invalid Nickname",  "name": "nickName"  },  {  "code": "Payee.AccountNumber.Invalid",  "httpStatus": 500,  "message": "Invalid Account Number.",  "name": "accountNumber"  },  {  "code": "Payee.Address.PostalCode.Invalid",  "httpStatus": 500,  "message": "Invalid Zip Code",  "name": "address.postalCode"  }  ]  } |
| *Result:* Missing and Invalid input errors are sent. | |
| **Scenario 6** Description: Account Number is invalid while adding managed payee. | |
| Request Body: | Response Body: |
| {  “name”: “Discover Credit Cards”,  “nickname”: “Discover Credit Cards”,  “accountNumber”: “1234”,  “verified”: true  } | {  "errors": [  {  "code": "Payee.AccountNumber.Mismatch",  "httpStatus": 500,  "message": "Given account number does not match records. ",  "name": "accountNumber"  }  ]  } |
| *Result:* Zip Code required error was sent from service. | |
| **Scenario 7** Description: invalid zip code received in request for merchant with zip required. | |
| Request Body: | Response Body: |
| {  “name”: “Salem Five Bank”,  “nickname”: “salem five bank payee”,  “accountNumber”: “0000000005500001”,  “address”: {  “postalCode”: “123456789”  },  “verified”:true  } | {  “errors”: [  {  “code”: “Payee.ZipCode.Mismatch”,  “httpStatus”: 500,  “message”: “Given Zip Code does not match records.” ,  "name": "address.postalCode"  }  ]  } |
| *Result:* Zip Code mismatch error was sent from service. | |
| **Scenario 8** Description: Managed payee not found for given information. | |
| Request Body: | Response Body: |
| {  “name”: “Salem Five Bank”,  “nickname”: “salem five bank payee”,  “accountNumber”: “0000000005500001”,  “address”: {  “postalCode”: “123456789”  },  “verified”:true  } | {  “errors”: [  {  “code”: “Payee.NotFound”,  “httpStatus”: 500,  “message”: “Selected Payee not found.”  }  ]  } |
| *Result:* Managed Payee Not found. | |
| **Scenario 9** Description: Service error occurred. | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "General.TechnicalDifficulties",  "httpStatus": 500,  "message": "Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information."  }  ]  } |

## Edit Managed Payee

|  |  |
| --- | --- |
| **Endpoint Name –** EditManagedPayee | |
| **Description –** Edit a managed payee for a customer. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payee/v1/update/{payeeID} | |
| **Request Type –** POST | |
| **Scenario 1** Description: Nicknames, and account number fields of a managed payee are successfully edited. | |
| Request Body: | Response Body: |
| {  "nickName":"happy nickname!",  "accountNumber": "0000000005501111",  "verified":true  }  Note: All fields are not required in edit.  Only send fields that were updated by the user. | {  "id": "102275196",  "name": "Salem Five Bank",  "nickName": "happy nickname!",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*1111",  "address": {  "postalCode": "019700940"  },  "verified": true,  "status": "ACTIVE",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "earliestPaymentDate": "2019-12-24",  "leadDays": 2,  "cutOffTime": "15:00:00",  "reminderEligible": true,  "ebillStatus": "AVAILABLE"  } |
| *Result:* Updated managed payee details are returned. | |
| **Scenario 2** Description: Invalid input received in request. | |
| Request Body: | Response Body: |
| {  "nickName":"",  "accountNumber": "",  "verified":true  } | {  "errors": [  {  "code": "Payee.AccountNumber.Invalid",  "httpStatus": 500,  "message": "Invalid Account Number.",  "name": "accountNumber"  },  {  "code": "Payee.Nickname.Invalid",  "httpStatus": 500,  "message": "Invalid Nickname",  "name": "nickName"  }  ]  } |
| *Result:* Invalid input errors sent. | |
| **Scenario 3** Description: User attempted to edit account number for Payee with E-Bill status enrolled or pending enrollment. | |
| Request Body: | Response Body: |
| {  “accountNumber”: “234567897”,  “verified”:true  } | {  “errors”: [  {  “code”: “Payee.AccountNumber.Edit.NotAllowed”,  “httpStatus”: 500,  “message”: “Payee Account Number cannot be updated.” ,  "name": "accountNumber"  }  ]  } |
| *Result:* Account number not editable error sent. | |
| **Scenario 4** Description: User tries to edit an existing payee. (Duplicate Payee Found) | |
| Request Body: | Response Body: |
| {  “accountNumber”: “0000000005500001”,  “verified”:true  } | {  “errors”: [  {  “code”: “Payee.Duplicate”,  “httpStatus”: 500,  “message”: “Payee Already Exists.”  }  ]  } |
| *Result:* User tries to add duplicate payee. | |
| **Scenario 5** Description: User tries to add payee with duplicate nickname. (Duplicate Payee Nickname Found) | |
| Request Body: | Response Body: |
| {  “nickname”: “salem payee”,  “verified”:true  } | {  "errors": [  {  "code": "Payee.Nickname.Exist",  "httpStatus": 500,  "message": "Nickname already exists. Enter a new Nickname.",  "name": "nickName"  }  ]  } |
| *Result:* User tries to add payee with duplicate nickname. | |
| **Scenario 6** Description: Service error occurred. | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "General.TechnicalDifficulties",  "httpStatus": 500,  "message": "Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information."  }  ]  } |

## Delete Payee

|  |  |
| --- | --- |
| **Endpoint Name –** DeletePayee | |
| **Description –** Delete a payee for a customer. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payee/v1/{payeeID} | |
| **Request Type –** POST | |
| **Scenario 1** Description: Unmanaged payee is successfully deleted. | |
| Request Body: | Response Body: |
| {} | {  "id": "102272277",  "name": "Helen",  "nickName": "Keller",  "accountNumber": "\*\*3123",  "phoneNumber": {  "number": "2244056873",  "formatted": "224-405-6873"  },  "address": {  "streetAddress": "2500 lake cook rd",  "locality": "Riverwoods",  "region": "IL",  "postalCode": "60015",  "formatted": "2500 lake cook rd \nRiverwoods IL 60015\nUSA"  },  "verified": false,  "status": "DELETED",  "deliveryMethod": "DIRECT\_CHECK",  "leadDays": 4,  "cutOffTime": "15:00:00",  "reminderEligible": true,  "ebillStatus": "NOT\_AVAILABLE"  } |
| *Result:* Deleted unmanaged payee details are returned. | |
| **Scenario 2** Description: Managed payee is successfully deleted. | |
| Request Body: | Response Body: |
| {} | {  "id": "102275196",  "name": "Salem Five Bank",  "nickName": "happy nickname!",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*1111",  "address": {  "postalCode": "019700940"  },  "verified": true,  "status": "DELETED",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "leadDays": 2,  "cutOffTime": "15:00:00",  "reminderEligible": true,  "ebillStatus": "AVAILABLE"  } |
| *Result:* Deleted managed payee details are returned. | |
| **Scenario 3** Description: User attempted to delete a Payee that is E-Bill enrolled or pending enrollment. | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “Payee.Delete.NotAllowed”,  “httpStatus”: 500,  “message”: “This Payee cannot be deleted.”  }  ]  } |
| *Result:* Delete Not Allowed error is returned. | |
| **Scenario 4** Description: Service error occurred. | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "General.TechnicalDifficulties",  "httpStatus": 500,  "message": "Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information."  }  ]  } |

## Get Merchants

|  |  |
| --- | --- |
| **Endpoint Name –** GetMerchants | |
| **Description –** Get list of merchants. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payee/v1/merchants | |
| **Request Type –** POST | |
| **Scenario 1** Description: Retrieve a list of available merchants. If zipRequired is true, then the zip code input field should be shown when adding payee. | |
| Request Body: | Response Body: |
| {} | [  {  “names”: [  “Liberty Mutual Insurance Group”  ],  “id”: “400737”,  “zipRequired”: true  },  {  “names”: [  “Pacific Power”  ],  “id”: “300226”,  “zipRequired”: false  }  ] |
| *Result:* Get merchants list | |

# 3. Discover Credit Cards

This operation is used for fetching the user Discover credit cards

## Get Discover Cards

|  |  |
| --- | --- |
| **Endpoint Name –** Get Discover Cards | |
| **Description –** Get a list of Discover Credit Cards that are not yet added as a user’s payees | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payee/v1/cards | |
| **Request Type –** GET | |
| **Scenario 1** Description: Customer has only one eligible Discover Credit Card that is not added as a payee. | |
| Request Body: | Response Body: |
|  | [  {  "name": "Discover Credit Card",  "nickName": "Discover Credit Card 9171",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*9171",  "verified": true,  "leadDays": 0,  "reminderEligible": true  }  ] |
| *Result:* A list of Discover Credit Card payee info is successfully returned. | |
| **Scenario 2** Description: Customer has multiple eligible Discover Credit Cards that are not yet added as payees. | |
| Request Body: | Response Body: |
|  | [  {  "name": "Discover Credit Card",  "nickName": "Discover Credit Card 9171",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*9171",  "verified": true,  "leadDays": 0,  "reminderEligible": true  },  {  "name": "Discover Credit Card",  "nickName": "Discover Credit Card 1590",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*1590",  "verified": true,  "leadDays": 0,  "reminderEligible": true  },  {  "name": "Discover Credit Card",  "nickName": "Discover Credit Card 8941",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*8941",  "verified": true,  "leadDays": 0,  "reminderEligible": true  }  ] |
| *Result:* All Discover Credit Card payee info is successfully returned in a list. | |
| **Scenario 3** Description: User does not have any eligible Discover Credit Cards that are not yet payees. | |
| Request Body: | Response Body: |
|  | [] |
| *Result:* An empty list is returned. | |
| **Scenario 4** Description: Service Error occurred. | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

## Add Discover Cards

|  |  |
| --- | --- |
| **Endpoint Name –** Add Discover Cards | |
| **Description –** Add multiple Discover Credit Cards as payees | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payee/v1/cards/add | |
| **Request Type –** POST | |
| **Scenario 1** Description: Add multiple Discover Credit Cards as payees. | |
| Request Body: | Response Body: |
| [  {  "name": "Discover Credit Card",  "nickName": "Discover Credit Card 9171",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*9171",  "verified": true,  },  {  "name": "Discover Credit Card",  "nickName": "Discover Credit Card 1590",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*1590",  "verified": true,  },  {  "name": "Discover Credit Card",  "nickName": "Discover Credit Card 8941",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*8941",  "verified": true,  }  ] | [  {  "id": "102268551",  "name": "Discover Credit Card",  "nickName": "Discover Credit Card 9171",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*9171",  "verified": true,  "status": "ACTIVE",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "earliestPaymentDate": "2019-11-01",  "leadDays": 2,  "cutOffTime": "13:00:00",  "reminderEligible": true,  "billerID": "300175",  "ebillStatus": "AVAILABLE",  "ebillCycle": "ONCE\_A\_MONTH"  },  {  "id": "102268552",  "name": "Discover Credit Card",  "nickName": "Discover Credit Card 1590",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*1590",  "verified": true,  "status": "ACTIVE",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "earliestPaymentDate": "2019-11-01",  "leadDays": 2,  "cutOffTime": "13:00:00",  "reminderEligible": true,  "billerID": "300175",  "ebillStatus": "AVAILABLE",  "ebillCycle": "ONCE\_A\_MONTH"  },  {  "id": "102268550",  "name": "Discover Credit Card",  "nickName": "Discover Credit Card 8941",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*8941",  "verified": true,  "status": "ACTIVE",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "earliestPaymentDate": "2019-11-01",  "leadDays": 2,  "cutOffTime": "13:00:00",  "reminderEligible": true,  "billerID": "300175",  "ebillStatus": "AVAILABLE",  "ebillCycle": "ONCE\_A\_MONTH"  }  ] |
| *Result:* Multiple Discover Credit Cards successfully added as payees and their respective payee details are returned. | |
| **Scenario 2** Description: Error occurred for one Discover Credit Card payee while adding multiple Discover Credit Cards as payees. | |
| Request Body: | Response Body: |
| [  {  "name": "Discover Credit Card",  "nickName": "Discover Credit Card 9171",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*9171",  "verified": true,  },  {  "name": "Discover Credit Card",  "nickName": "Discover Credit Card 1590",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*1590",  "verified": true,  }  ] | [  {  "name": "Discover Credit Card",  "nickName": "Discover Credit Card 9171",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*9171",  "verified": true,  "status": "Failure",  "leadDays": 0,  "reminderEligible": true  },  {  "id": "102268554",  "name": "Discover Credit Card",  "nickName": "Discover Credit Card 1590",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*1590",  "verified": true,  "status": "ACTIVE",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "earliestPaymentDate": "2019-11-01",  "leadDays": 2,  "cutOffTime": "13:00:00",  "reminderEligible": true,  "billerID": "300175",  "ebillStatus": "AVAILABLE",  "ebillCycle": "ONCE\_A\_MONTH"  }  ] |
| *Result:* Only one Discover Credit Card was not added. The other cards were successfully added as payees.  **Note:** There will be no error in case of failures. The payee status will be returned as “Failure”. | |

# 4. Payment Reminders

This module is used to manage all Payment Reminders for a user.

## Set Reminder

|  |  |
| --- | --- |
| **Endpoint Name –** SetPaymentReminder | |
| **Description –** Set a payment reminder for a payee. This is common for both Add / Edit reminder flow. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payment/v1/reminder | |
| **Request Type –** POST | |
| **Scenario 1** Description: Payee is eligible for payment reminders and payment reminder is successfully set for a payee. | |
| Request Body: | Response Body: |
| {  “id”: “102261207”,  “reminder”: {  “frequency”: “ONCE\_IN\_6\_MONTHS”,  “amount”: 42.51,  “date”:”2019-10-15”  }  } | Status: 201 Created |
| *Result:* Payment reminder successfully set for payee with response status: 201 Created. | |
| **Scenario 2** Description: Invalid frequency, amount, and date received in request. | |
| Request Body: | Response Body: |
| {  "id": "102274970",  "reminder": {  "frequency": "NEVER",  "amount": 2000000000.301,  "date":"2017-12-31"  }  } | {  "errors": [  {  "code": "Payment.Amount.Invalid.Max",  "httpStatus": 500,  "message": "Your bill payment exceeds the maximum allowable amount of $25,000.",  "name": "reminder.amount"  },  {  "code": "Payment.DeliverBy.Invalid.Start",  "httpStatus": 500,  "message": "Please select a valid delivery date.",  "name": "reminder.date"  },  {  "code": "Payment.Frequency.Invalid",  "httpStatus": 500,  "message": "Please select a valid frequency",  "name": "reminder.frequency"  },  {  "code": "Payment.Amount.Invalid",  "httpStatus": 500,  "message": "The entered amount is invalid. Please enter a valid amount.",  "name": "reminder.amount"  }  ]  } |
| *Result:* Invalid amount error is returned. | |
| **Scenario 2** Description: Service Error occurred | |
| Request Body: | Response Body: |
| {  “id”: “102261207”,  “reminder”: {  “frequency”: “ONCE\_IN\_6\_MONTHS”,  “amount”: 42.63,  “date”:”2019-10-15”  }  } | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

## Delete Reminder

|  |  |
| --- | --- |
| **Endpoint Name –** DeletePaymentReminder | |
| **Description –** Delete the payment reminder for a payee. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payment/v1/reminder/{payeeID} | |
| **Request Type –** POST | |
| **Scenario 1** Description: Payment reminder for a payee is successfully deleted. | |
| Request Body: | Response Body: |
| {} | Status: 204 No Content |
| *Result:* Payment reminder successfully deleted for payee with response status: 204 No Content. | |
| **Scenario 2** Description: Service Error occurred | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

# 5. Payment Operations

This module is used to add, edit and delete payments for a user

## Schedule a One-Time Payment

|  |  |
| --- | --- |
| **Endpoint Name –** AddPayment | |
| **Description –** Schedule a one-time payment for a payee. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payment/v1/add | |
| **Request Type –** POST | |
| **Scenario 1** Description: One-time payment for a payee is successfully scheduled. | |
| Request Body: | Response Body: |
| {  "payee" : {  "id" : "102274501"  },  "amount": 1.84,  "paymentMethod" : {  "id": "SM1179491"  },  "memo": "wow",  "note": "wee",  "deliverByDate" : "2020-01-09",  "frequency":"ONE\_TIME"  }  Note : Need frequency in request for both one-time and repeating payments | {  "paymentId": "876112718",  "payee": {  "id": "102274501",  "name": "Discover Credit Cards",  "nickName": "card payee",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*2089"  },  "paymentMethod": {  "id": "SM1179491",  "accountName": " CASHBACK DEBIT ",  "accountNumber": "\*\*\*\*\*\*1122"  },  "amount": 1.84,  "deliverByDate": "2020-01-09",  "memo": "wow",  "note": "wee",  "type": "ONE\_TIME",  "status": "SCHEDULED",  "scheduledOn": "2019-12-24",  "confirmationNumber": "RBSBLZK4",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "\_links": {  "Edit": {  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/edit/876112718  ?envc=aer&X-DFS-ENV=asys"  },  "Delete": {  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/delete/876112718  ?envc=aer&X-DFS-ENV=asys"  }  }  } |
| *Result:* Scheduled payment confirmation details displayed in response | |
| **Scenario 2** Description: Invalid input given for one-time payment. | |
| Request Body: | Response Body: |
| {  "payee" : {  "id" : "102271072"  },  "amount": 123456.090,  "paymentMethod" : {  "id": "SM1326367"  },  "memo": "<>",  "note": "{}",  "deliverByDate" : "2015-10-25",  "frequency":"NeVeR"  } | {  "errors": [  {  "code": "Payment.Amount.Invalid.Max",  "httpStatus": 500,  "message": "Your bill payment exceeds the maximum allowable amount of $25,000.",  "name": "amount"  },  {  "code": "Payment.Amount.Invalid",  "httpStatus": 500,  "message": "The entered amount is invalid. Please enter a valid amount.",  "name": "amount"  },  {  "code": "Payment.DeliverBy.Invalid.Start",  "httpStatus": 500,  "message": "Please select a valid delivery date.",  "name": "deliverByDate"  },  {  "code": "Payment.Frequency.Invalid",  "httpStatus": 500,  "message": "Please select a valid frequency",  "name": "frequency"  },  {  "code": "Payment.Memo.Invalid",  "httpStatus": 500,  "message": "The following characters are not allowed: <>()&;\"[]{}",  "name": "memo"  },  {  "code": "Payment.Note.Invalid",  "httpStatus": 500,  "message": "The following characters are not allowed: <>()&;\"[]{}",  "name": "note"  }  ]  } |
| *Result:* invalid input error list is returned in the response. | |
| **Scenario 3** Description: Duplicate one-time payment requested. | |
| Request Body: | Response Body: |
| {  "payee" : {  "id" : "102271072"  },  "amount": 1.84,  "paymentMethod" : {  "id": "SM1179491"  },  "memo": "wow",  "note": "wee",  "deliverByDate" : "2020-01-09",  "frequency":"ONE\_TIME"  } | {  "errors": [  {  "code": "Payment.Duplicate",  "httpStatus": 500,  "message": "We''re sorry. You cannot schedule a duplicate payment."  }  ]  } |
| *Result:* duplicate payment error is returned in the response. | |
| **Scenario 4** Description: One-time payment scheduled on a bank holiday or weekend (invalid date). | |
| Request Body: | Response Body: |
| {  “payee” : {  “id” : “102260013”  },  “amount”: 3.12,  “paymentMethod” : {  “id”: “SM1326367”  },  “deliverByDate” : “2019-12-25”,  “frequency”:”ONE\_TIME”  }  Note: Weekends and holidays should be blocked in the UI calendar. Incase it comes to the backend we throw this error. | {  "errors": [  {  "code": "Payment.DeliverBy.Invalid.BankHoliday",  "httpStatus": 500,  "message": "Please select a valid delivery date." ,  "name": "deliverByDate"  }  ]  } |
| *Result:* Invalid payment date error is returned in the response. | |
| **Scenario 5** Description: Service Error occurred | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

## Set a Repeating Payment **(For both Add and Edit flow)**

|  |  |
| --- | --- |
| **Endpoint Name –** AddPayment | |
| **Description –** Schedule a repeating payment rule for a payee. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payment/v1/add | |
| **Request Type –** POST | |
| **Scenario 1** Description: Repeating payment for a payee (Never Ends Scenario) is successfully scheduled. | |
| Request Body: | Response Body: |
| {  "payee" : {  "id" : "102270923"  },  "amount": 1.24,  "paymentMethod" : {  "id": "SM1031548"  },  "memo": "wow",  "deliverByDate" : "2020-01-31",  "frequency":"ONCE\_A\_WEEK"  }  Note :  Need frequency in request for both one-time and repeating payments.  For Repeating Payment setup only memo is available.  For “Never Ends” Scenario, no need to send any end date in the request. | {  "payee": {  "id": "102270923",  "name": "hero123",  "nickName": "hero123"  },  "paymentMethod": {  "id": "SM1031548",  "accountName": " CASHBACK DEBIT ",  "accountNumber": "\*\*\*\*\*\*1681"  },  "amount": 1.24,  "deliverByDate": "2020-01-31",  "frequency": "ONCE\_A\_WEEK",  "memo": "wow",  "ruleId": "8829454"  } |
| *Result:* Scheduled payment confirmation details displayed in response | |
| **Scenario 2** Description: Repeating payment for a payee (No Of Payments) is successfully scheduled. | |
| Request Body: | Response Body: |
| {  "payee" : {  "id" : "102270923"  },  "amount": 1.51,  "paymentMethod" : {  "id": "SM1031548"  },  "memo": "This is repeating",  "deliverByDate" : "2020-01-31",  "frequency":"ONCE\_A\_WEEK",  "noOfPayments" : 10  }  Note :  Need frequency in request for both one-time and repeating payments.  For Repeating Payment setup only memo is available. | {  "payee": {  "id": "102270923",  "name": "hero123",  "nickName": "hero123"  },  "paymentMethod": {  "id": "SM1031548",  "accountName": " CASHBACK DEBIT ",  "accountNumber": "\*\*\*\*\*\*1681"  },  "amount": 1.51,  "deliverByDate": "2020-01-31",  "frequency": "ONCE\_A\_WEEK",  "memo": "This is repeating",  "noOfPayments": 10,  "ruleId": "8829428"  } |
| *Result:* Scheduled payment confirmation details displayed in response | |
| **Scenario 3** Description: Invalid input given for one-time payment. | |
| Request Body: | Response Body: |
| {  “payee” : {  “id” : “102260013”  },  “amount”: 0.102,  “paymentMethod” : {  “id”: “SM1326367”  },  “memo”: “[]”,  “note”: “’;”,  “deliverByDate” : “2011-11-08”,  “frequency”:”FOREVER”  } | {  "errors": [  {  "code": "Payment.Frequency.Invalid",  "httpStatus": 500,  "message": "Please select a valid frequency",  "name": "frequency"  },  {  "code": "Payment.Amount.Invalid",  "httpStatus": 500,  "message": "The entered amount is invalid. Please enter a valid amount.",  "name": "amount"  },  {  "code": "Payment.Memo.Invalid",  "httpStatus": 500,  "message": "The following characters are not allowed: <>()&;\"[]{}",  "name": "memo"  },  {  "code": "Payment.Note.Invalid",  "httpStatus": 500,  "message": "The following characters are not allowed: <>()&;\"[]{}",  "name": "note"  },  {  "code": "Payment.Amount.Invalid.Min",  "httpStatus": 500,  "message": "Your bill payment is less than the minimum allowable amount of $1.",  "name": "amount"  },  {  "code": "Payment.DeliverBy.Invalid.Start",  "httpStatus": 500,  "message": "Please select a valid delivery date.",  "name": "deliverByDate"  }  ]  } |
| *Result:* invalid input error list is returned in the response. | |
| **Scenario 4** Description: Service Error occurred | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

## Schedule an E-Bill One-Time Payment

|  |  |
| --- | --- |
| **Endpoint Name –** Add Payment | |
| **Description –** Schedule a one-time payment for a payee. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payment/v1/add | |
| **Request Type –** POST | |
| **Scenario 1** Description: A one-time payment for an e-bill enrolled payee is successfully scheduled. | |
| Request Body: | Response Body: |
| {  "payee" : {  "id" : "102282877"  },  "amount": 27.20,  "paymentMethod" : {  "id": "SM1019776"  },  "note":"one-time ebill",  "deliverByDate" : "2020-02-20",  "frequency":"ONE\_TIME",  "billId":"208241336"  }  **Note** – “billId” is needed for an ebill one-time payment.  billId is returned in the GetBillDetails endpoint | {  "paymentId": "876137557",  "payee": {  "id": "102282877",  "name": "Cox Communications",  "nickName": "another cox",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*7123"  },  "paymentMethod": {  "id": "SM1019776",  "accountName": " CASHBACK DEBIT ",  "accountNumber": "\*\*\*\*\*\*8858"  },  "amount": 27.2,  "deliverByDate": "2020-02-20",  "note": "one-time ebill",  "type": "ONE\_TIME",  "status": "SCHEDULED",  "scheduledOn": "2020-02-05",  "confirmationNumber": "XB7BFZY4",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "\_links": {  "Edit": {  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/edit/876137557  ?envc=aer&X-DFS-ENV=mst0"  },  "Delete": {  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/delete/876137557  ?envc=aer&X-DFS-ENV=mst0"  }  }  } |
| *Result:* Scheduled payment confirmation details are displayed in response | |
| **Scenario 2** Description: Service Error occurred. | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

## Set an E-Bill Automatic Payment **(For both Add and Edit flow)**

|  |  |
| --- | --- |
| **Endpoint Name –** SetAutomaticPayments | |
| **Description –** Set e-bill automatic payments | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payment/v1/autopay/add | |
| **Request Type –** POST | |
| **Scenario 1** Description: Add or Edit an e-bill automatic payment for an e-bill enabled payee’s bills. | |
| Request Body: | Response Body: |
| {  "payee" : {  "id" : "102283322"  },  "paymentMethod" : {  "id":"SM1357625"  },  "automaticPayment": {  "amount":"PAY\_FULL\_BALANCE",  "paymentDate":"ON\_DUE\_DATE"  }  } | {  "payee": {  "id": "102283322",  "name": "Verizon Wireless",  "nickName": "Verizon Wireless",  "accountNumber": "\*\*\*\*\*\*7898"  },  "paymentMethod": {  "id": "SM1357625",  "accountName": " CASHBACK DEBIT ",  "accountNumber": "\*\*\*\*\*\*5548"  },  "automaticPayment": {  " amount": "PAY\_FULL\_BALANCE",  " paymentDate": "ON\_DUE\_DATE"  }  } |
| *Result:* Automatic e-bill payments successfully set. | |
| ***Additional Information:***  Automatic Payments are only allowed for payees with ebillStatus as **ACTIVE**.  automaticPayment – The following values can be used for automatic payments:   * amount:   + PAY\_FULL\_BALANCE – pay full bill balance.   + PAY\_MINIMUM\_BALANCE – pay minimum bill balance. * paymentDate:   + ON\_DUE\_DATE – pay on the bill due date.   + EBILL\_ARRIVES – pay on next available date when bill arrives. | |
| **Scenario 2** Description: Invalid input given for automatic payment. | |
| Request Body: | Response Body: |
| {  "payee" : {  "id" : "102283297"  },  "paymentMethod" : {  "id":"SM1357625"  },  "automaticPayment": {  "amount":"PALL\_BALANCE",  "paymentDate":"ON\_DAE"  }  } | {  "errors": [  {  "code": "AutoPay.PaymentDate.Invalid",  "httpStatus": 500,  "message": "AutoPay paymentDate is invalid.",  "name": "automaticPayment.paymentDate"  },  {  "code": "AutoPay.Amount.Invalid",  "httpStatus": 500,  "message": "AutoPay amount is invalid.",  "name": "automaticPayment.amount"  }  ]  } |
| *Result:* Set automatic payment failed. | |
| **Scenario 3** Description: Input missing for automatic payment. | |
| Request Body: | Response Body: |
| {  "payee" : {  "id" : "102283297"  },  "paymentMethod" : {  "id":"SM1357625"  },  "automaticPayment": {  }  } | {  "errors": [  {  "code": "AutoPay.Amount.NotEmpty",  "httpStatus": 500,  "message": "AutoPay amount can not be blank.",  "name": "automaticPayment.amount"  },  {  "code": "AutoPay.PaymentDate.NotEmpty",  "httpStatus": 500,  "message": "AutoPay paymentDate can not be blank.",  "name": "automaticPayment.paymentDate"  }  ]  } |
| *Result:* Set automatic payment failed. | |
| **Scenario 4** Description: Set automatic payment was attempted for a payee with a non-eligible e-bill status. | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "Autopay.Not.Eligible",  "httpStatus": 500,  "message": "Payee doesn't have Active E-Bills"  }  ]  } |
| *Result:* Set automatic payment failed. | |
| **Scenario 5** Description: Service Error occurred. | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

## Edit Any One-Time Payment (Including a Single Payment from a Repeating Rule and Bill Payment)

|  |  |
| --- | --- |
| **Endpoint Name –** EditPayment | |
| **Description –** Edit a one-time payment or a single payment from a repeating payment rule for a payee. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payment/v1/edit/{paymentID} | |
| **Request Type –** POST | |
| **Scenario 1** Description: Payment is successfully edited. | |
| Request Body: | Response Body: |
| {  "amount": 1.21,  "paymentMethod" : {  "id": "SM7624202"  },  "memo": "edited",  "note":"changed",  "deliverByDate" : "2020-01-14"  } | {  "paymentId": "876111124",  "payee": {  "id": "102271072",  "name": "left",  "nickName": "right"  },  "paymentMethod": {  "id": "SM7624202",  "accountName": " CASHBACK DEBIT ",  "accountNumber": "\*\*\*\*\*\*9073"  },  "amount": 1.21,  "deliverByDate": "2020-01-14",  "memo": "edited",  "note": "changed",  "paymentAddress": {  "streetAddress": "4850 wright terr",  "extendedAddress": "test",  "locality": "skokie",  "region": "IL",  "postalCode": "60077"  },  "type": "ONE\_TIME",  "status": "SCHEDULED",  "scheduledOn": "2019-12-18",  "confirmationNumber": "EBKBYZK4",  "deliveryMethod": "DIRECT\_CHECK" ,  "\_links": {  "Edit": {  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/edit/876111124  ?envc=aer&X-DFS-ENV=asys"  },  "Delete": {  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/delete/876111124  ?envc=aer&X-DFS-ENV=asys"  }  }  } |
| *Result:* Updated payment confirmation details displayed in response | |
| **Scenario 2** Description: Invalid input given for edit payment. | |
| Request Body: | Response Body: |
| {  “amount”: 0.102,  “paymentMethod” : {  “id”: “SM1326367”  },  “deliverByDate” : “2010-10-08”,  “memo”:”<>”,  “note”:”;;;;”  } | {  "errors": [  {  "code": "Payment.DeliverBy.Invalid.Start",  "httpStatus": 500,  "message": "Please select a valid delivery date.",  "name": "deliverByDate"  },  {  "code": "Payment.Amount.Invalid.Min",  "httpStatus": 500,  "message": "Your bill payment is less than the minimum allowable amount of $1.",  "name": "amount"  },  {  "code": "Payment.Note.Invalid",  "httpStatus": 500,  "message": "The following characters are not allowed: <>()&;\"[]{}",  "name": "note"  },  {  "code": "Payment.Memo.Invalid",  "httpStatus": 500,  "message": "The following characters are not allowed: <>()&;\"[]{}",  "name": "memo"  },  {  "code": "Payment.Amount.Invalid",  "httpStatus": 500,  "message": "The entered amount is invalid. Please enter a valid amount.",  "name": "amount"  }  ]  } |
| *Result:* invalid input error list is returned in the response. | |
| **Scenario 3** Description: Edit is attempted for non-editable payment. | |
| Request Body: | Response Body: |
| {  “amount”: 1.12,  “paymentMethod” : {  “id”: “SM1326367”  },  “deliverByDate” : “2019-10-08”,  “memo”:”this is a canceled payment”,  “note”:”should not be edited!”  } | {  "errors": [  {  "code": "Payment.Edit.NotAllowed",  "httpStatus": 500,  "message": "This payment is not editable."  }  ]  } |
| *Result:* Not editable error is displayed in response | |
| **Scenario 4** Description: Edit is attempted for non-existing payment. | |
| Request Body: | Response Body: |
| {  “amount”: 1.12,  “paymentMethod” : {  “id”: “SM1326367”  },  “deliverByDate” : “2019-10-08”,  “memo”:”no such payment exists!”,  “note”:”how can we edit!?”  } | {  "errors": [  {  "code": "Payment.NotFound",  "httpStatus": 500,  "message": "The requested payment could not be found."  }  ]  } |
| *Result:* Payment not found error is displayed in response | |
| **Scenario 5** Description: Service Error occurred | |
| Request Body: | Response Body: |
| {  “amount”: 3.12,  “paymentMethod” : {  “id”: “SM1326367”  },  “deliverByDate” : “2019-10-08”  } | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

## Delete Any One-Time Payment (Including a Single Payment from a Repeating Rule and Bill Payment)

|  |  |
| --- | --- |
| **Endpoint Name –** DeletePayment | |
| **Description –** Delete a one-time payment or a single payment from a repeating payment rule for a payee. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payment/v1/delete/{paymentId} | |
| **Request Type –** POST | |
| **Scenario 1** Description: Payment is successfully deleted / canceled. | |
| Request Body: | Response Body: |
| {} | Status: 204 No Content |
| *Result:* Status code 204 returned for successful deletion. | |
| **Scenario 2** Description: Delete is attempted on non-cancelable payment. | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "Payment.Delete.NotAllowed",  "httpStatus": 500,  "message": "This payment cannot be deleted."  }  ]  } |
| *Result:* Not cancelable error is returned in the response. | |
| **Scenario 3** Description: Delete is attempted for a non-existing payment. | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "Payment.NotFound",  "httpStatus": 500,  "message": "The requested payment could not be found."  }  ]  } |
| *Result:* Payment not found error is displayed in response | |
| **Scenario 4** Description: Service Error occurred | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

## Delete a Series of Repeating Payments

|  |  |
| --- | --- |
| **Endpoint Name –** DeletePayment | |
| **Description –** Delete a series of repeating payments rule for a payee. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payment/v1/delete/{paymentId}?type=all | |
| **Request Type –** POST | |
| **Scenario 1** Description: Payment is successfully deleted / canceled. | |
| Request Body: | Response Body: |
| {} | Status: 204 No Content |
| *Result:* Status code 204 returned for successful repeating series deletion. | |
| **Scenario 2** Description: Delete is attempted on non-cancelable payment. | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "Payment.Delete.NotAllowed",  "httpStatus": 500,  "message": "This payment cannot be deleted."  }  ]  } |
| *Result:* Not cancelable error is returned in the response. | |
| **Scenario 3** Description: Delete is attempted for a non-existing payment. | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “Payment.NotFound”,  “httpStatus”: 500,  “message”: “The requested payment could not be found.”  }  ]  } |
| *Result:* Payment not found error is displayed in response | |
| **Scenario 4** Description: Service Error occurred | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

## Delete E-Bill Automatic Payment

|  |  |
| --- | --- |
| **Endpoint Name –** DeletePaymentRule | |
| **Description –** Delete e-bill automatic payment rule | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payment/v1/rule/delete/{payeeId} | |
| **Request Type –** POST | |
| **Scenario 1** Description: Automatic payment rule is successfully deleted / canceled. | |
| Request Body: | Response Body: |
| {} | Status: 204 No Content |
| *Result:* Status code 204 returned for successful repeating series deletion. | |
| **Scenario 2** Description: Service Error occurred | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

# 6. Review Payments

This module is used to view a user’s payment activity.

## Get Payment List

|  |  |
| --- | --- |
| **Endpoint Name –** GetPaymentList | |
| **Description –** Get list of all payments for a user. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payment/v1/ | |
| **Request Type –** GET | |
| **Scenario 1** Description: List of all payments is returned in ascending date order. | |
| Request Body: | Response Body: |
| {} | [  {  "paymentId": "876082580",  "payee": {  "id": "102271063",  "name": "Midcontinent Communications",  "nickName": "Midcontinent Communications Co",  "accountNumber": "\*\*\*\*\*7888"  },  "paymentMethod": {  "id": "SM7624202",  "accountName": " CASHBACK DEBIT ",  "accountNumber": "\*\*\*\*\*\*9073"  },  "amount": 1.14,  "deliverByDate": "2019-11-18",  "memo": "a memo",  "note": "a note",  "type": "ONE\_TIME",  "status": "COMPLETED",  "scheduledOn": "2019-11-14",  "confirmationNumber": "MBPBPZZ4",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "links": []  },  {  "paymentId": "876082602",  "payee": {  "id": "102271072",  "name": "left",  "nickName": "right"  },  "paymentMethod": {  "id": "SM1179491",  "accountName": " CASHBACK DEBIT ",  "accountNumber": "\*\*\*\*\*\*1122"  },  "amount": 1.23,  "deliverByDate": "2019-11-19",  "memo": "a memo",  "paymentAddress": {  "streetAddress": "4850 wright terr",  "extendedAddress": "test",  "locality": "skokie",  "region": "IL",  "postalCode": "60077"  },  "ruleId": "8826162",  "type": "REPEATING",  "status": "SCHEDULED",  "scheduledOn": "2019-11-14",  "confirmationNumber": "GBTBPZZ4",  "deliveryMethod": "DIRECT\_CHECK",  "links": [  {  "rel": "Edit",  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/edit/876082602  ?envc=aer&X-DFS-ENV=asys"  },  {  "rel": "Delete",  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/delete/876082602  ?envc=aer&X-DFS-ENV=asys"  }  ]  },  {  "paymentId": "876136182",  "payee": {  "id": "102283292",  "name": "Cox Communications",  "nickName": "coxx",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*7141"  },  "paymentMethod": {  "id": "SM1357625",  "accountName": " CASHBACK DEBIT ",  "accountNumber": "\*\*\*\*\*\*5548"  },  "amount": 22,  "deliverByDate": "2020-02-05",  "type": "EBILL\_AUTOPAY",  "status": "SCHEDULED",  "scheduledOn": "2020-02-03",  "confirmationNumber": "EB4B4ZY4",  "deliveryMethod": "STANDARD\_ELECTRONIC",  "links": [  {  "rel": "Edit",  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/edit/876136182  ?envc=aer&X-DFS-ENV=tsys"  },  {  "rel": "Delete",  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/delete/876136182  ?envc=aer&X-DFS-ENV=tsys"  }  ]  }  ] |
| *Result:* Payment details for all scheduled payments and past payments from the last 18 months are returned as a list. | |
| **Scenario 2** Description: Service Error occurred | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

## Get Scheduled Payment List for a Payee

|  |  |
| --- | --- |
| **Endpoint Name –** GetPaymentList | |
| **Description –** Get list of all scheduled payments for a specific payee. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payment/v1?payees={payeeID}&statuses=SCHEDULED | |
| **Request Type –** GET | |
| **Scenario 1** Description: List of all scheduled payments for a specific payee is successfully returned. | |
| Request Body: | Response Body: |
| {} | [  {  “paymentId”: “876061028”,  “payee”: {  “id”: “102262071”,  “name”: “Discover Credit Cards”,  “nickname”: “card payee”,  “accountNumber”: “\*\*\*\*\*\*\*\*\*\*\*\*2090”  },  “paymentMethod”: {  “id”: “SM1326173”,  “accountName”: “RETAIL PERSONAL CHECKING “,  “accountNumber”: “\*\*\*\*\*\*9815”  },  “amount”: 2.61,  “deliverByDate”: “2020-01-07”,  “memo”: “this is one-time”,  “note”: “notes”,  “type”: “ONE\_TIME”,  “status”: “SCHEDULED”,  “scheduledOn”: “2019-09-27”,  “confirmationNumber”: “OBEBGZZ4”,  “deliveryMethod”: “STANDARD\_ELECTRONIC” ,  "links": {  "Edit": {  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/edit/876061028  ?envc=aer&X-DFS-ENV=asys"  },  "Delete": {  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/delete/876061028  ?envc=aer&X-DFS-ENV=asys"  }  }  },  {  “paymentId”: “876061027”,  “payee”: {  “id”: “102262071”,  “name”: “Discover Credit Cards”,  “nickname”: “card payee”,  “accountNumber”: “\*\*\*\*\*\*\*\*\*\*\*\*2090”  },  “paymentMethod”: {  “id”: “SM1326173”,  “accountName”: “RETAIL PERSONAL CHECKING “,  “accountNumber”: “\*\*\*\*\*\*9815”  },  “amount”: 2.51,  “deliverByDate”: “2020-04-01”,  “memo”: “this is one-time”,  “type”: “REPEATING”,  “ruleId”: “12345678”,  “status”: “SCHEDULED”,  “scheduledOn”: “2019-09-27”,  “confirmationNumber”: “VBEBGZZ4”,  “deliveryMethod”: “STANDARD\_ELECTRONIC”,  "links": {  "Edit": {  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/edit/876061027  ?envc=aer&X-DFS-ENV=asys"  },  "Delete": {  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/delete/876061027  ?envc=aer&X-DFS-ENV=asys"  }  }  }  ] |
| *Result:* Payment details for all scheduled payments in the coming year for a specific payee are returned as a list. | |
| **Scenario 2** Description: No payments found for the given search criteria. | |
| Request Body: | Response Body: |
| {} | [] |
| *Result:* Empty list is returned. | |
| **Scenario 3** Description: Service Error occurred | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

## Search Payments

|  |  |
| --- | --- |
| **Endpoint Name –** GetPaymentList | |
| **Description –** Get list of all scheduled payments for a specific payee. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payment/v1?accounts={accountIDs}payees={payeeIDs}  &min={minimumAmount}&max={maximumAmount}  &from={fromDate}&to={toDate}&statuses={statuses} | |
| **Request Type –** GET | |
| **Scenario 1** Description: List of payments for specific search parameters is successfully returned in ascending date order. | |
| Request Body: | Response Body: |
| {} | [  {  “paymentId”: “876061028”,  “payee”: {  “id”: “102262071”,  “name”: “Discover Credit Cards”,  “nickname”: “card payee”,  “accountNumber”: “\*\*\*\*\*\*\*\*\*\*\*\*2090”  },  “paymentMethod”: {  “id”: “SM1326173”,  “accountName”: “RETAIL PERSONAL CHECKING “,  “accountNumber”: “\*\*\*\*\*\*9815”  },  “amount”: 2.61,  “deliverByDate”: “2020-01-07”,  “memo”: “this is one-time”,  “note”: “notes”,  “type”: “ONE\_TIME”,  “status”: “SCHEDULED”,  “scheduledOn”: “2019-09-27”,  “confirmationNumber”: “OBEBGZZ4”,  “deliveryMethod”: “STANDARD\_ELECTRONIC” ,  "links": [  {  "rel": "Edit",  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/edit/876061028  ?envc=aer&X-DFS-ENV=asys"  },  {  "rel": "Delete",  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/delete/876061028  ?envc=aer&X-DFS-ENV=asys"  }  ]  },  {  “paymentId”: “876061027”,  “payee”: {  “id”: “102262071”,  “name”: “Discover Credit Cards”,  “nickname”: “card payee”,  “accountNumber”: “\*\*\*\*\*\*\*\*\*\*\*\*2090”  },  “paymentMethod”: {  “id”: “SM1326173”,  “accountName”: “RETAIL PERSONAL CHECKING “,  “accountNumber”: “\*\*\*\*\*\*9815”  },  “amount”: 2.51,  “deliverByDate”: “2020-04-01”,  “memo”: “this is one-time”,  “type”: “REPEATING”,  “ruleId”: “8826167”,  “status”: “SCHEDULED”,  “scheduledOn”: “2019-09-27”,  “confirmationNumber”: “VBEBGZZ4”,  “deliveryMethod”: “STANDARD\_ELECTRONIC” ,  "links": [  {  "rel": "Edit",  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/edit/876061027  ?envc=aer&X-DFS-ENV=asys"  },  {  "rel": "Delete",  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/delete/876061027  ?envc=aer&X-DFS-ENV=asys"  }  ]  }  ] |
| *Result:* Payment details for all scheduled payments in the coming year for a specific payee are returned as a list. | |
| *Additional Information:*  Search Parameters:  **accounts** – list of account IDs (ex: accounts=12300,45600,78900)  **payees** – list of payee IDs (ex: payees=12300,45600,78900)  **min** – minimum search amount in double format (ex: min=1.02) note: minimum amount is $1.00  **max** – maximum search amount in double format (ex: max=1000.55) note: maximum amount is $25000.00  **from** – beginning search date in YYYY-mm-dd format (ex: from=2019-11-25)  **to** – ending search date in YYYY-mm-dd format (ex: to=2019-11-25)  **statuses** – list of payment statuses (ex: statuses=SCHEDULED,PROCESSING)  Payment Status Options:  SCHEDULED, PROCESSING, COMPLETED, CANCELED, OTHER | |
| **Scenario 2** Description: No payments found for the given search criteria. | |
| Request Body: | Response Body: |
| {} | [] |
| *Result:* Empty list is returned. | |
| **Scenario 3** Description: Service Error occurred | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

## Repeating Payment History

|  |  |
| --- | --- |
| **Endpoint Name –** GetPaymentList | |
| **Description –** Get list of all scheduled payments for a specific payee. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payment/v1? payees={payeeID}&ruleId={ruleId} | |
| **Request Type –** GET | |
| **Scenario 1** Description: List of payments for specific search parameters is successfully returned in ascending date order. | |
| Request Body: | Response Body: |
| {} | [  {  “paymentId”: “876061027”,  “payee”: {  “id”: “102262071”,  “name”: “Discover Credit Cards”,  “nickname”: “card payee”,  “accountNumber”: “\*\*\*\*\*\*\*\*\*\*\*\*2090”  },  “paymentMethod”: {  “id”: “SM1326173”,  “accountName”: “RETAIL PERSONAL CHECKING “,  “accountNumber”: “\*\*\*\*\*\*9815”  },  “amount”: 2.51,  “deliverByDate”: “2020-04-01”,  “memo”: “this is one-time”,  “type”: “REPEATING”,  “ruleId”: “8826167”,  “status”: “SCHEDULED”,  “scheduledOn”: “2019-09-27”,  “confirmationNumber”: “VBEBGZZ4”,  “deliveryMethod”: “STANDARD\_ELECTRONIC” ,  "links": [  {  "rel": "Edit",  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/edit/876061027  ?envc=aer&X-DFS-ENV=asys"  },  {  "rel": "Delete",  "href": "https://messaging.discover.com  /bank/deposits/pay/billpay/payment/v1/delete/876061027  ?envc=aer&X-DFS-ENV=asys"  }  ]  }  ] |
| *Result:* Payment details for all scheduled payments in the coming year for a specific payee are returned as a list. | |
| *Additional Information:*  **ruleId** is provided for a repeating payment in GetPaymentList default response.  **ruleId** is null for non-repeating payments.  Payee ID and ruleId are both required for this request. | |
| **Scenario 2** Description: No payments found for the given search criteria. | |
| Request Body: | Response Body: |
| {} | [] |
| *Result:* Empty list is returned. | |
| **Scenario 3** Description: Service Error occurred | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

# 7. Ebill Operations

This module is used for payee ebill operations

## Get E-Bill Tokens

|  |  |
| --- | --- |
| **Endpoint Name –** GetEBillTokens | |
| **Description –** Get a biller payee’s token values for E-Bill enrollment | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/ebill/v1/token/{payeeId} | |
| **Request Type –** GET | |
| **Scenario 1** Description: Token details are requested for a specific payee that is eligible for e-bill enrollment. | |
| Request Body: | Response Body: |
| {} | **Example 1:**  {  "id": "102265895",  "name": "Salem Five Bank",  "nickName": "salem ebill",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*0022",  "ebillsTokenList": [  {  "tokenID": "54529",  "label": "Up to 11 Digit Loan Account Number",  "name": "Account Number",  "format": "0",  "formatType": "First 11 Digits of the Pay FI Account Number",  "delimiter": "|",  "valueRequired": true  },  {  "tokenID": "54530",  "label": "Up to 5 Digit Loan Note Number",  "name": "Note Number",  "format": "0",  "formatType": "Digits 12 to 16 of the Pay FI Account Number",  "delimiter": "|",  "valueRequired": true  }  ],  "ebillStatus": "AVAILABLE"  } |
| Request Body: | Response Body: |
| {} | **Example 2: with session key**  {  "id": "102265896",  "name": "Discover Credit Cards",  "nickName": "disco ebill",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*2045",  "sessionKey": "1020827",  "ebillsTokenList": [  {  "tokenID": "Q1",  "label": "User ID or Acct Number",  "name": "User ID or Acct Number",  "formatType": "TextBox"  },  {  "tokenID": "Q2",  "label": "Password",  "name": "Password",  "formatType": "PassWord"  }  ],  "ebillStatus": "AVAILABLE"  } |
| *Result:* Payee details and E-Bill enrollment token details are returned. | |
| *Additional Information*  ebillsTokenList – Returns the list of tokens / fields to be shown in UI. This list is dynamic and will vary for different payees.  Token list contains the label name and format type to be displayed in UI. Some payees have a session key. | |
| **Scenario 2** Description: Get E-Bill Token List attempted on Payee with E-Bill already Enrolled | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "Ebill.Enroll.NotEligible",  "httpStatus": 500,  "message": "This payee is not eligible for E-Bill enrollment."  }  ]  } |
| *Result:* E-Bill enrollment not eligible error is returned. | |
| **Scenario 3** Description: Service Error occurred. | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

## E-Bill Enrollment

|  |  |
| --- | --- |
| **Endpoint Name –** EBillEnrollment | |
| **Description –** Enroll a payee for E-Bill | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/ebill/v1/enroll | |
| **Request Type –** POST | |
| **Scenario 1** Description: Payee E-Bill enrollment without session ID is successful. | |
| Request Body: | Response Body: |
| Example 1: Biller’s Account Number and Zip Code (of max size 10) are Required {  “payeeID”: “102263701”,  "ebillsTokenList": [  {  "tokenID": "54504",  "label": "Your Account Number as listed on bill",  "name": "Account Number",  "format": "0",  "formatType": "OTHER",  "delimiter": "|",  "valueRequired": true,  “value”: “234567898”  },  {  "tokenID": "54505",  "label": "Enter Your Zip Code as it appears on your bill",  "name": "Zip Code",  "format": "0",  "formatType": "ZIP10RESTRICT TEXT",  "delimiter": "|",  "valueRequired": true,  “value”: “60015”  }  ]  } | Response 1:  {  “id”: “102263701”,  “name”: “Mid Continent Communications”,  “nickname”: “Mid Cont”,  “accountNumber”: “\*\*\*\*\*7898”,  “ebillStatus”: “PENDING\_ACTIVATION”  } |
| Request Body: | Response Body: |
| Example 2: Biller’s 11 Digit Loan Account Number and 5 Digit Loan Note Number are Required  {  "id": "102265895",  "ebillsTokenList": [  {  "tokenID": "54529",  "label": "Up to 11 Digit Loan Account Number",  "name": "Account Number",  "format": "0",  "formatType": "First 11 Digits of the Pay FI Account Number",  "delimiter": "|",  "value": "00000000055"  },  {  "tokenID": "54530",  "label": "Up to 5 Digit Loan Note Number",  "name": "Note Number",  "format": "0",  "formatType": "Digits 12 to 16 of the Pay FI Account Number",  "delimiter": "|",  "value": "00022"  }  ]  } | Response 2:  {  “id”: “102263755”,  “name”: “Salem Five Bank”,  “nickname”: “salem mid Test”,  “accountNumber”: “\*\*\*\*\*\*\*\*\*\*\*\*0022”,  “ebillStatus”: “PENDING\_ACTIVATION”  } |
| Request Body: | Response Body: |
| Example 3: Biller Requires value of Terms Check-Box  {  “payeeID”: “102263668”,  "ebillsTokenList": [  {  "tokenID": "54078",  "label": " <http://www.cox.com/policy/papersuppression.asp>",  "name": " <http://www.cox.com/policy/papersuppression.asp>",  "format": "0",  "formatType": "CHECKBOX",  "delimiter": "|",  "valueRequired": true,  “value”: “Accepted”  }  ]  } | Response 3:  {  “id”: “102263668”,  “name”: “Cox Communications”,  “nickname”: “Cox COM”,  “accountNumber”: “\*\*\*\*\*\*\*\*\*\*\*\*7987”,  “ebillStatus”: “PENDING\_ACTIVATION”  } |
| *Result:* Payee’s E-Bill status details are returned. | |
| ***Additional Information:***  ebillStatus – Provides payee e-bill status information with the following status values:   * AVAILABLE – Payee eligible for e-bill enrollment. * ACTIVE – Payee is already enrolled in e-bill. * NOT\_AVAILABLE – Payee is not eligible for e-bill enrollment. * PENDING\_ACTIVATION – E-bill activation is pending. * REQUEST\_REJECTED – E-bill payee rejected a change to the e-bill status. * VALIDATION\_FAILED – E-bill enrollment validation failed. | |
| **Scenario 2** Description: Payee E-Bill enrollment flow with session ID is successful. | |
| Request Body: | Response Body: |
| **First Step:** First token response list is submitted  {  "id": "102265896",  "sessionKey": "1020827",  "ebillsTokenList": [  {  "tokenID": "Q1",  "label": "User ID or Acct Number",  "name": "User ID or Acct Number",  "formatType": "TextBox",  "value":"6011208974002045"  },  {  "tokenID": "Q2",  "label": "Password",  "name": "Password",  "formatType": "PassWord",  "value":"ccccc"  }  ]  } | **First Response:** Next token list is returned  {  "id": "102265896",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*2045",  "sessionKey": "1020827",  "ebillsTokenList": [  {  "tokenID": "Q1",  "label": " In what city was your high school? (full name of city only)",  "name": " In what city was your high school? (full name of city only)",  "formatType": "TextBox"  }  ],  "ebillStatus": "AVAILABLE"  } |
| Request Body: | Response Body: |
| **Next Step:** Next token response list is submitted  {  "id": "102265896",  "sessionKey": "1020827",  "ebillsTokenList": [  {  "tokenID": "Q1",  "label": " In what city was your high school? (full name of city only)",  "name": " In what city was your high school? (full name of city only)",  "formatType": "TextBox",  "value":"Chicago"  }  ]  } | **Final Response:** Validation Success  {  "id": "102265896",  "name": "Discover Credit Cards",  "nickName": "disco ebill",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*2045",  "ebillStatus": "PENDING\_ACTIVATION"  } |
| *Result:* After multiple token submissions the biller is successfully enrolled in e-bills | |
| ***Additional Information:***For billers with an e-bill enroll session key, there will be multiple token list submissions.  Each token submission must call the e-bill enroll endpoint and must include the session key. | |
| **Scenario 3** Description: Enroll E-Bill attempted on Payee with E-Bill already Enrolled | |
| Request Body: | Response Body: |
| {  “payeeID”: “102263668”,  "ebillsTokenList": [  {  "tokenID": "54078",  "label": " <http://www.cox.com/policy/papersuppression.asp>",  "name": " <http://www.cox.com/policy/papersuppression.asp>",  "format": "0",  "formatType": "CHECKBOX",  "delimiter": "|",  "valueRequired": true,  “value”: “Accepted”  }  ]  } | {  "errors": [  {  "code": "Ebill.Enroll.NotEligible",  "httpStatus": 500,  "message": "This payee is not eligible for E-Bill enrollment."  }  ]  } |
| *Result:* E-Bill enrollment not eligible error is returned. | |
| **Scenario 4** Description: Service Error occurred. | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

## E-Bill Un-enroll

|  |  |
| --- | --- |
| **Endpoint Name –** EBillUnenroll | |
| **Description –** Un-enroll a payee from e-bills | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/ebill/v1/unenroll/{payeeID} | |
| **Request Type –** POST | |
| **Scenario 1** Description: Payee E-Bill unenrollment is successful | |
| Request Body: | Response Body: |
| {} | {  “id”: “102263759”  “name”: “Salem Five Bank”,  “nickname”: “Salem Five Bank”,  “accountNumber”: “\*\*\*\*\*\*\*\*\*\*\*\*0001”,  “ebillStatus”: “AVAILABLE”  } |
| *Result:* E-Bill successfully unenrolled for payee. Payee is eligible for e-bill enrollment again with “ebillStatus”: “AVAILABLE” | |
| ***Additional Information:***  ebillStatus – Provides payee e-bill status information with the following status values:   * AVAILABLE – Payee eligible for e-bill enrollment. * ACTIVE – Payee is already enrolled in e-bill. * NOT\_AVAILABLE – Payee is not eligible for e-bill enrollment. * PENDING\_ACTIVATION – E-bill activation is pending. * REQUEST\_REJECTED – E-bill payee rejected a change to the e-bill status. * VALIDATION\_FAILED – E-bill enrollment validation failed. | |
| **Scenario 2** Description: Payee is not enrolled in E-Bills | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "Ebill.UnEnroll.NotEligible",  "httpStatus": 500,  "message": "This payee is not eligible for E-Bill un-enrollment."  }  ]  } |
| *Result:* E-Bill un-enroll NotEligible error is sent. | |
| **Scenario 3** Description: Service Error occurred. | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

## Get E-Bill Details

|  |  |
| --- | --- |
| **Endpoint Name –** GetBillDetails | |
| **Description –** Get e-bill details for all e-bill enrolled payees | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/ebill/v1/ | |
| **Request Type –** GET | |
| **Scenario 1** Description: Customer has e-bill enrolled payees with bill information. | |
| Request Body: | Response Body: |
| {} | [  {  "id": "102281809",  "name": "Cox Communications",  "nickName": "Cox Communication",  "accountNumber": "\*\*\*\*\*\*\*\*\*\*\*\*7980",  "ebillList": [  {  "id": "208241288",  "type": "DIRECT",  "status": "UNPAID",  "paymentURIDetails": "https://bpm-uat.readiness.billdomain.com  /csp/ShowPBillServlet?brid=7742&token=  CA9E8FD38DB51281D676B7D513723AED9A8  C51016B6B0C8CF5CBA52AE36A3EAA4725087E  6B9D92CAE9569E947575CE7E&Pg=1",  "cycleStatus": "IN\_CYCLE",  "payeeId": "102281809",  "dueDate": "2020-02-08",  "statementDate": "2020-01-09",  "receivedDate": "2020-01-29",  "totalDue": 117.95,  "minimumDue": 22  },  {  "id": "208240961",  "type": "DIRECT",  "status": "MARKED\_PAID",  "paymentURIDetails": "https://bpm-uat.readiness.billdomain.com  /csp/ShowPBillServlet?brid=7742&token=  CA9E8FD38DB51281D676B7D513723AEDED891  229179E3E9D249C340971D412B9A176A05C2FE9  90263D5CFD8672A8A7E9&Pg=1",  "cycleStatus": "IN\_CYCLE",  "payeeId": "102281809",  "dueDate": "2020-02-05",  "statementDate": "2020-01-06",  "receivedDate": "2020-01-26",  "totalDue": 117.95,  "minimumDue": 22  }  ],  "ebillStatus": "ACTIVE"  }  ] |
| *Result:* List of E-bill details for all e-bill enrolled payees are returned.  **Note:** The ebillList “id” is the bill-ID used to perform operations on an e-bill. | |
| **Scenario 2** Description: Customer does not have any e-bill enrolled payees with bill information. | |
| Request Body: | Response Body: |
| {} | [] |
| *Result:* An empty list is returned. | |
| **Scenario 3** Description: Service Error occurred. | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

## File E-Bill (Mark as Paid)

|  |  |
| --- | --- |
| **Endpoint Name –** FileEBill | |
| **Description –** File Ebill | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/ebill/v1/filebill/{billId} | |
| **Request Type –** POST | |
| **Scenario 1** Description: An e-bill enabled payee’s bill is marked as paid. | |
| Request Body: | Response Body: |
| {} | {  "id": "208235103",  "type": "DIRECT",  "status": "MARKED\_PAID",  "cycleStatus": "IN\_CYCLE",  "payeeId": "102271549",  "dueDate": "2019-10-30",  "statementDate": "2019-09-30",  "receivedDate": "2019-10-20",  "totalDue": 107.95,  "minimumDue": 21  } |
| *Result:* File e-bill is successful, confirmed by the status which is returned as: “MARKED\_PAID”. | |
| **Scenario 2** Description: Service Error occurred. | |
| Request Body: | Response Body: |
| {} | {  “errors”: [  {  “code”: “General.TechnicalDifficulties”,  “httpStatus”: 500,  “message”: “Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information.”  }  ]  } |

# 8. Bill Pay Utility Methods

This section has common bill pay methods

## Get Bank Holiday List

|  |  |
| --- | --- |
| **Endpoint Name –** Get Bank Holidays | |
| **Description –** Get a list of bank holidays for a specific year. | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payee/v1/holidays/{YYYY} | |
| **Request Type –** GET | |
| **Scenario 1** Description: Retrieve bank holiday list for a specific year | |
| Request Body: | Response Body: |
| {} | [  "2020-01-01",  "2020-01-20",  "2020-02-17",  "2020-05-25",  "2020-07-03",  "2020-09-07",  "2020-10-12",  "2020-11-11",  "2020-11-26",  "2020-12-25"  ] |
| *Result:* List of bank holidays are returned. | |
| **Scenario 2** Description: Service Error occurred. | |
| Request Body: | Response Body: |
| {} | {  "errors": [  {  "code": "General.TechnicalDifficulties",  "httpStatus": 500,  "message": "Some of our web site features are unavailable at this time. We sincerely apologize for this temporary inconvenience and are working to resolve this issue. Please contact us for more information."  }  ]  } |

## Get Payment Accounts

|  |  |
| --- | --- |
| **Endpoint Name –** GetPaymentAccounts | |
| **Description –** Get customer payment eligible accounts | |
| **Endpoint URL –** {hostname}/bank/deposits/pay/billpay/payment/v1/account | |
| **Request Type –** GET | |
| Request Body: | Response Body: |
| {} | [  {  "id": "SM34098",  "fundingId": "18805316",  "type": "MONEYMARKET",  "accountName": "MONEY MARKET",  "accountNumber": "\*\*\*\*\*\*2943",  "preferred": true,  "availableBalance": 30778.85  },  {  "id": "SM7624202",  "fundingId": "18802641",  "type": "CHECKING",  "accountName": "CASHBACK DEBIT",  "accountNumber": "\*\*\*\*\*\*9073",  "preferred": false,  "availableBalance": 874.27  }  ] |