

EasyCard Next Generation API v1 - *Checkout Page (Redirect Page)*

The simple way to collect payments hosted by EasyCard.

There are two options to generate Checkout Page:

- classic style with query string which contains all deal parameters (which can be used first of all to support legacy systems)
- Payment-Intent based Checkout Page

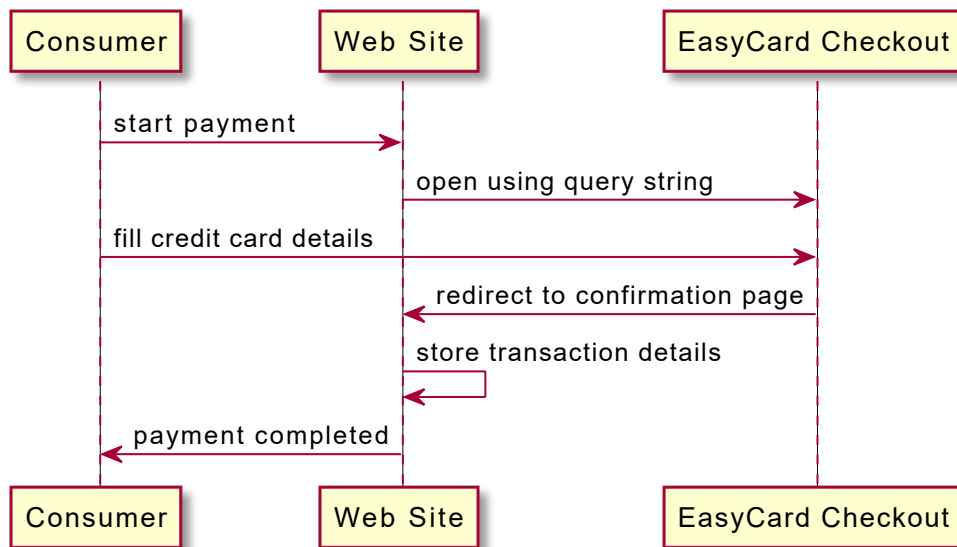
The screenshot displays the EasyCard Checkout Page. At the top, there is a blue header bar containing a logo placeholder labeled 'YOUR LOGO' and 'TAGLINE GOES HERE', the text 'Bob's Enterprise', and a language selector set to 'English'. Below the header, the page title is 'Goods and services from Test Wen Store: My Product'. The 'Transaction type' is set to 'Regular Deal' in a dropdown menu. The 'Total Sum' is displayed as '100.00₺'. The payment form includes fields for Card Number, Expiration, CVV, Card Owner Name, Email, National ID, Phone, and an optional Authorization Code. At the bottom of the form are two buttons: 'Cancel payment' and 'Pay with EasyCard'. The footer shows '© 2021 - EasyCard'.

Please note that under the hood Checkout Page generates regular payment transaction, so all documentation and settings related to transactions are relevant also for transactions generated by Checkout Page - please see corresponding documentation.

Checkout Page url for live environment is <https://checkout.e-c.co.il>

Classic Checkout Page

The general behavior is next:



Checkout page can be opened inside iframe

Query string parameters

Name	In	Type	Required	Description
Description	query	string	true	deal description
Name	query	string	true	consumer name
NationalID	query	string	false	consumer national id
Email	query	string	false	consumer's email
Phone	query	string	false	consumer's phone
Currency	query	string	false	currency in ISO format, like ILS, USD etc.
Amount	query	decimal	false	deal amount, can be omitted - see description below
UserAmount	query	boolean	false	use this flag if consumer can change payment amount
RedirectUrl	query	string	false	url on your website which will accept response from EasyCard
ApiKey	query	string	true	Shared api key - see description below
IssueInvoice	query	boolean	false	invoice will be generated by EasyCard if "true"
AllowPinPad	query	boolean	false	Pinpad device payment can be used within Checkout page

Sample query string:

```
https://checkout.e-c.co.il/?
Description=Goods%20and%20services%20from%20Test%20Web%20Store%3A%20My%20Product&Name=John%20Smith&Email=testemail@gmail.com&Currency=ILS&Amount=99.99&RedirectUrl=https%3A%2F%2Fecng-testwebstore.azurewebsites.net%3A443%2FPaymentResult%3FInternalOrderID%3D123456&ApiKey=XXXXXXXX&IssueInvoice=True&AllowPinPad=False&UserAmount=False
```

You can use simple **Test Web Store** (demo web page) which demonstrate query string redirect: <https://ecng-testwebstore.azurewebsites.net/> If you need to specify any additional parameters, like **OrderID** or some security key in your system, you can simple add it to **RedirectUrl**

Configuration in EasyCard Merchant's Portal

On EasyCard Merchant's portal <https://merchant.e-c.co.il> there are few settings related to Checkout Page. To open the settings page you need to press on your user name link:

The screenshot shows the 'My profile' settings page in the EasyCard Merchant's Portal. The top navigation bar is blue with the 'easy/card' logo and buttons for 'Charge', 'Request', and 'Refund'. Below the navigation bar, the user's name 'Vladimir Volkov' and a clock icon are visible. The left sidebar contains a menu with options: Dashboard, Transactions, Customers, Items, Invoicing, Payment Requests, and Billing Deals. The main content area is titled 'My profile' and includes the following elements:

- App Version: 1.0.0.20211123.3
- Buttons: ACCOUNT SETTINGS and SIGN OUT
- Language: English (dropdown)
- Terminal: Vlad's terminal 29/07 (dropdown)
- Currency: ₪ (dropdown)
- Terminal Settings section with a table:

Label	Created	Status
Vlad's terminal 29/07	Friday, 3 April 2020 10:47	Pending Approval

21 / 50

Features section with toggle switches:

- Prevent Double Transactions: ☐
- Billing: ☒
- Sms Notification: ☐

Redirect urls

Then you need to find **Checkout Redirect URLs** section:

Dashboard

Transactions

Customers

MS

Routing

Payment Requests

Linking Deals

Checkout

Custom CSS

[Link to CSS](#)

Issue Invoice

Checkout Redirect URLs

Extended response for checkout transaction

ADD

Redirect URL #1

https://ecng-testwebstore.azurewebsites.net

Redirect URL #2

https://localhost:3000

Please add base address of your website here. You can use several base addresses. **RedirectUrl** in query string described above should belongs to one of the configured base urls.

Custom logo and stylesheet

You can specify custom logo image and custom css file. Please use **Checkout** section and **Payment Request** section:

4 / 13


Payment Request

Default Request Subject
Payment request from Coca Cola
30 / 128

Default Refund Request Subject
0 / 128

From Address
no-reply@cc.co.il

Merchant Logo



Billing

Notification emails
0 / 512

☒ Create Recurrent Payments Automatically

Separate Emails with Commas

Checkout


Custom CSS

[Link to CSS](#)

☒ Issue Invoice

Checkout Redirect URLs

☐ Extended response for checkout transaction



Legacy redirect url response

In **Checkout Redirect URLs** section there is a flag **Extended response for checkout transaction** enables additional query string parameters which will be added to **RedirectUrl** by EasyCard. In this mode NextGen EasyCard will mimic EasyCard classic behavior.

Checkout Redirect URLs

☐ Extended response for checkout transaction

You can use **Test Web Store** to check all parameters which will be present in this mode.

In case if **Extended response for checkout transaction** is disabled, only EasyCard **TransactionID** will be present in response, then you can get transaction details via API (see below).

Shared API Key

You can get **ApiKey** for simple Checkout Page using **Show Shared Key** button in **Checkout** section or generate new key using **Reset Shared Key** button:

☒ Use Quick Charge by Default

Min Installments
 Max Installments

Min Credit Installments
 Max Credit Installments

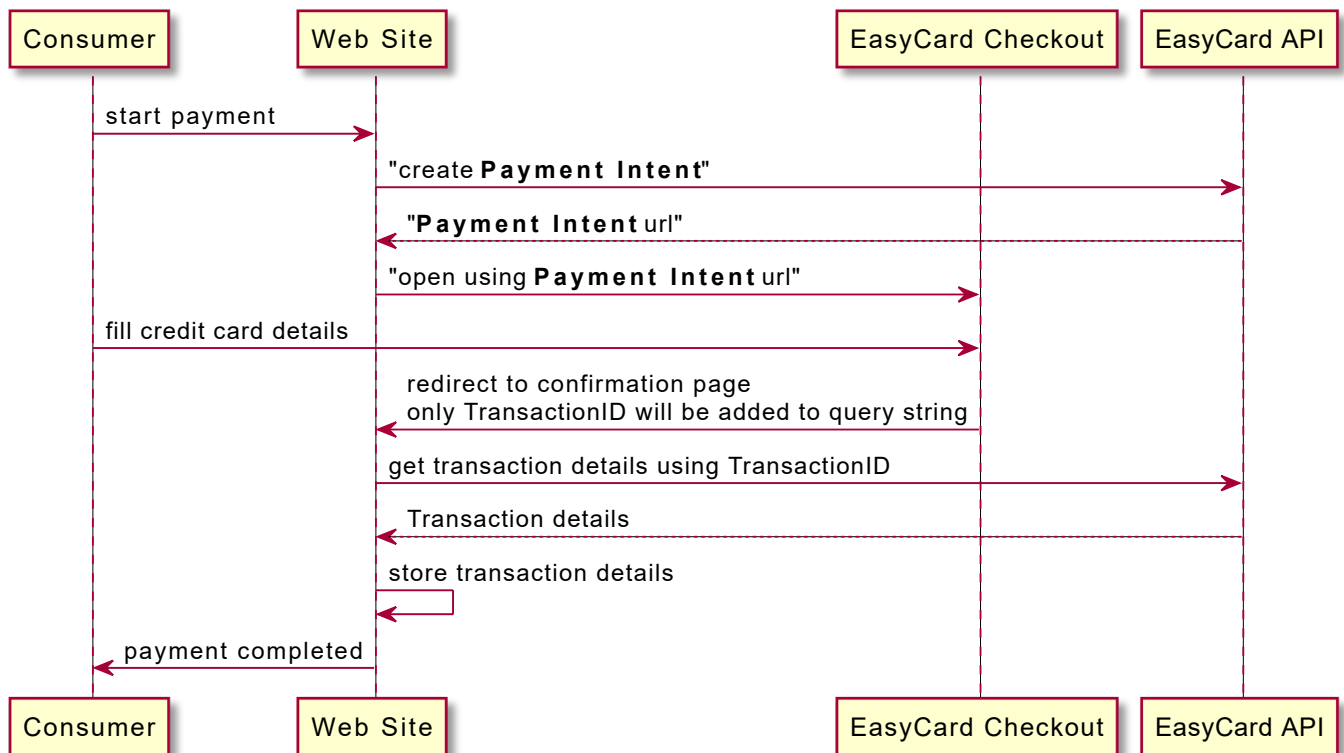
Shared Api Key

Shared Api Key

Private Api Key

Advanced Checkout Page

EasyCard Next Generation also provides more secured and solid way to collect Checkout transactions using Payment Intent API:



Please find out API description in section below.

You can simple check how this functionality works by using **Payment Link** feature on Merchant's Portal:

Vlad's terminal 29/07

Payment Request Settings

Due Date

☐ Refund

☐ Allow PinPad

☐ Installment Transaction

☐ Allow User Amount

Customer Name

Vlad Volkov

11 / 50

Customer Email

vit.muromsky@gmail.com

Customer Phone

12345678

Customer Address

Test Strasse

Deal Description

0 / 250

Invoice Type

Invoice With Payment Info >

Invoice Subject

Invoice

Additional Email to CC

Separate Emails with Commas

PAYMENT REQUEST

CREATE PAYMENT LINK

EasyCard Next Generation API v1 - *Transactions API* (sections related to Checkout Page)

Please check details in full API description document

Live transactions api address is <https://api.e-c.co.il>

Authentication

- API Key (oauth2)
 - Parameter Name: **Authorization**, in: header. Standard Authorization header using the Bearer scheme. Example: "bearer {token}"

PaymentIntent

- (Auth policies: terminal_or_merchant_frontend_or_admin)

POST /api/paymentIntent

Body parameter

```
{
  "terminalID": "57d5dd68-4280-44f6-a8f7-547180e3a1d6",
  "dealDetails": {
    "dealReference": "string",
    "dealDescription": "string",
    "consumerEmail": "user@example.com",
    "consumerName": "string",
    "consumerNationalID": "string",
    "consumerPhone": "string",
    "consumerID": "9e541e9d-045d-45e2-8cf0-ead10a4f23cf",
    "items": [
      {
        "itemID": "f1f85a48-b9b1-447d-a06c-c1acf57ed3a8",
        "externalReference": "string",
        "itemName": "string",
        "sku": "string",
        "price": 0,
        "quantity": 0.01,
        "amount": 0,
        "vatRate": 1,
        "vat": 0,
        "netAmount": 0,
        "discount": 0,
        "extension": {
          "property1": [
            []
          ],
          "property2": [
            []
          ]
        }
      }
    ],
    "consumerAddress": {
      "countryCode": "string",
      "city": "string",
      "zip": "string",
      "street": "string",
      "house": "string",
      "apartment": "string"
    }
  },
}
```



```

    "consumerExternalReference": "string"
  },
  "currency": "ILS",
  "paymentRequestAmount": 0,
  "dueDate": "2019-08-24T14:15:22Z",
  "installmentDetails": {
    "numberOfPayments": 1,
    "initialPaymentAmount": 0.01,
    "totalAmount": 0.01,
    "installmentPaymentAmount": 0.01
  },
  "invoiceDetails": {
    "invoiceType": "invoice",
    "invoiceSubject": "string",
    "sendCCTo": [
      "string"
    ]
  },
  "pinPadDetails": {
    "terminalID": "string"
  },
  "issueInvoice": true,
  "allowPinPad": true,
  "vatRate": 1,
  "vatTotal": 0,
  "netTotal": 0,
  "requestSubject": "string",
  "fromAddress": "string",
  "isRefund": true,
  "redirectUrl": "string",
  "userAmount": true,
  "cardOwnerNationalID": "string",
  "extension": {
    "property1": [
      []
    ],
    "property2": [
      []
    ]
  }
}

```

Parameters

Name	In	Type	Required	Description
body	body	Transactions.Api.Models.PaymentRequests.PaymentRequestCreate	false	none

Example responses

201 Response

```

{
  "message": "string",
  "status": "success",
  "entityID": 0,
  "entityUID": "1a366e78-0ad8-4c6d-9367-977d46d0652e",
  "entityReference": "string",
  "correlationId": "string",
  "entityType": "string",
  "errors": [
    {
      "code": "string",
      "description": "string"
    }
  ],
  "concurrencyToken": "string",
  "innerResponse": {
    "message": "string",
    "status": "success",
    "entityID": 0,
    "entityUID": "1a366e78-0ad8-4c6d-9367-977d46d0652e",
    "entityReference": "string",
    "correlationId": "string",
    "entityType": "string",
    "errors": [
      {
        "code": "string",
        "description": "string"
      }
    ]
  },
  "concurrencyToken": "string",
  "innerResponse": {},
  "additionalData": {
    "property1": [
      []
    ],
    "property2": [
      []
    ]
  }
},
"additionalData": {
  "property1": [
    []
  ],
  "property2": [
    []
  ]
}
}

```

Responses

Status	Meaning	Description	Schema
201	Created	Success	Shared.Api.Models.OperationResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Forbidden	None

To perform this operation, you must be authenticated by means of one of the following methods: oauth2 (Scopes: terminal_or_merchant_frontend_or_admin)

Transactions

GET /api/transactions/{transactionID}

- (Auth policies: terminal_or_merchant_frontend_or_admin)

Parameters

Name	In	Type	Required	Description
transactionID	path	string(uuid)	true	none

Example responses

200 Response

```
{
  "paymentTransactionID": "19fb5b03-41a8-4687-a896-97851484218b",
  "transactionDate": "2019-08-24T14:15:22Z",
  "transactionTimestamp": "2019-08-24T14:15:22Z",
  "initialTransactionID": "7423475c-5ccd-47b4-826d-813710b50fda",
  "currentDeal": 0,
  "terminalID": "57d5dd68-4280-44f6-a8f7-547180e3a1d6",
  "terminalName": "string",
  "merchantID": "c30c8496-c2db-425f-8614-0e5f16767e31",
  "status": "initial",
  "paymentTypeEnum": "card",
  "quickStatus": "Pending",
  "finalizationStatus": "initial",
  "transactionType": "regularDeal",
  "specialTransactionType": "regularDeal",
  "jDealType": "J4",
  "transactionJ5ExpiredDate": "2019-08-24T14:15:22Z",
  "rejectionReason": "unknown",
  "currency": "ILS",
  "cardPresence": "cardNotPresent",
  "numberOfPayments": 0,
  "transactionAmount": 0,
  "amount": 0,
  "initialPaymentAmount": 0,
  "totalAmount": 0,
  "installmentPaymentAmount": 0,
  "creditCardDetails": {
```

```
"cardNumber": "string",
"cardExpiration": {
  "year": 20,
  "month": 1,
  "expired": true
},
"cardVendor": "string",
"cardBrand": "string",
"solek": "string",
"cardOwnerName": "string",
"cardOwnerNationalID": "string",
"cardReaderInput": "string"
},
"creditCardToken": "string",
"dealDetails": {
  "dealReference": "string",
  "dealDescription": "string",
  "consumerEmail": "user@example.com",
  "consumerName": "string",
  "consumerNationalID": "string",
  "consumerPhone": "string",
  "consumerID": "9e541e9d-045d-45e2-8cf0-ead10a4f23cf",
  "items": [
    {
      "itemID": "f1f85a48-b9b1-447d-a06c-c1acf57ed3a8",
      "externalReference": "string",
      "itemName": "string",
      "sku": "string",
      "price": 0,
      "quantity": 0.01,
      "amount": 0,
      "vatRate": 1,
      "vat": 0,
      "netAmount": 0,
      "discount": 0,
      "extension": {
        "property1": [
          []
        ],
        "property2": [
          []
        ]
      }
    }
  ]
},
"consumerAddress": {
  "countryCode": "string",
  "city": "string",
  "zip": "string",
  "street": "string",
  "house": "string",
  "apartment": "string"
},
"consumerExternalReference": "string"
},
```

```
"shvaTransactionDetails": {},
"clearingHouseTransactionDetails": {},
"upayTransactionDetails": {},
"updatedAt": "2019-08-24T14:15:22Z",
"updateTimestamp": "string",
"allowTransmission": true,
"allowTransmissionCancellation": true,
"billingDealID": "4cd118ba-cace-48f5-9541-35d080140955",
"rejectionMessage": "string",
"vatRate": 0,
"vatTotal": 0,
"netTotal": 0,
"consumerIP": "string",
"merchantIP": "string",
"correlationId": "string",
"invoiceID": "4f03a727-9c4b-43a5-b699-271a3e6fdc9c",
"issueInvoice": true,
"documentOrigin": "UI",
"paymentRequestID": "e1c25505-9cc4-4f14-8180-c89257d3d43a",
"processorResultCode": 0
}
```

Responses

Status	Meaning	Description	Schema
200	OK	Success	Transactions.Api.Models.Transactions.TransactionResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Forbidden	None

To perform this operation, you must be authenticated by means of one of the following methods: oauth2 (Scopes: terminal_or_merchant_frontend_or_admin)