

Instrukcja Api dla klientów

Api dla klientów korzysta z Api GraphQL dostępnego w Magento 2. Ogólne informacje na temat działania oraz obsługi Api można znaleźć pod adresem:

<https://devdocs.magento.com/guides/v2.3/graphql/>

Zapytania do Api należy wysyłać metodą POST na adres sklepu dodając na końcu /graphql a w body requestu precyzujemy z jakiego zapytania bądź mutacji chcemy skorzystać.

GraphQL jest bardzo elastyczny w zakresie wysyłanych i pobieranych danych dlatego poniższe requesty należy potraktować jako przykłady, można je edytować, usunąć niektóre dane z responsu jeśli nie są potrzebne, w razie potrzeby dodać inne itd. warto zapoznać się z podlinkowaną wyżej dokumentacją aby poznać pełne możliwości.

1. Pobieranie tokenu customera.

Aby wykonywać dalsze operacje należy pobrać token customera. Poniżej przykładowe body requestu:

```
mutation {
  generateCustomerToken(
    email: "firstname.lastname@mail.net",
    password: "password123*"
  ) {
    token
  }
}
```

przykładowy response:

```
{
  "data": {
    "generateCustomerToken": {
      "token":
"eyJraWQpOilmliWiYWxnIjo1SFMyNTYifQ.eyJ1aWQiOjE1LCJ1dHlwIj0iOjMsImIhdCI6MTY5MjE4NTQxM2wiZXhwIjoxNjkyMTg5MDEwfQ.bLC24kLjuD8j0sfETi-jXQoCFRfYR_UoRUgexy
mXo8w"
    }
  }
}
```

Otrzymany token należy dodać do headera każdego kolejnego requestu jako klucz "Authorization" o wartości "Bearer *otrzymany token*".

2. Pobieranie informacji produktowej wraz z ceną indywidualną.

przykładowe body pozwalające na pobranie informacji produktowej o podanym sku wraz z ceną indywidualną:

```
{
  products(search: "Rozrusznik") {
    total_count
    items {
      name
      sku
      price_range {
        minimum_price {
          regular_price {
            value
            currency
          }
        }
        individual_price {
          net
          gross
          currency
        }
      }
    }
  }
}
page_info {
  page_size
  current_page
}
```

przykładowy response:

```
{
  "data": {
    "products": {
      "total_count": 1,
      "items": [
        {
          "name": "Rozrusznik",
```

```

    "sku": "Rozrusznik",
    "price_range": {
      "minimum_price": {
        "regular_price": {
          "value": 15.977701,
          "currency": "PLN"
        },
        "individual_price": {
          "net": 12,
          "gross": 14.76,
          "currency": "PLN"
        }
      }
    },
    "page_info": {
      "page_size": 20,
      "current_page": 1
    }
  }
}
}
}

```

3. Pobieranie informacji o zamówieniach

aby pobrać listę zamówień należy skorzystać z query getOrderList, poniżej przykładowy request i response zawierający ogólne informacje o zamówieniu. Dla zwięzłości zapytanie nie zawiera pola 'items' zwracającego pozycje zamówienia, zostanie ono przedstawione dalej w przykładzie pobrania pojedynczego zamówienia jednak w razie potrzeby można o nie zapytać również w tym zapytaniu.

```

request:
{
  getOrderList {
    order_id
    order_number
    status
    billing_address
    shipping_address
    products_net_total
  }
}

```

```
products_tax_total
products_gross_total
shipping_net_total
shipping_gross_total
shipping_method
shipping_number
shipping_tracking_link
ordered_at
shipped_at
updated_at
ordered_by_email
}
}
```

response:

```
{
  "data": {
    "getOrderList": [
      {
        "order_id": 1,
        "order_number": "123",
        "status": "new",
        "billing_address": "ul. Testowa 1 12-345 Warszawa",
        "shipping_address": "ul. Testowa 1 12-345 Warszawa",
        "products_net_total": 22,
        "products_tax_total": 2,
        "products_gross_total": 22,
        "shipping_net_total": 22,
        "shipping_gross_total": 22,
        "shipping_method": "pickup",
        "shipping_number": "AZ12345",
        "shipping_tracking_link": "https://testtrack.com/AZ12345",
        "ordered_at": "2023-07-05 11:53:03",
        "shipped_at": "2023-07-05 11:54:59",
        "updated_at": "2023-07-25 06:20:52",
        "ordered_by_email": "companyuser@company.com"
      },
      {
        "order_id": 2,
        "order_number": "124",
        "status": "new",
        "billing_address": "ul. Testowa 1 12-345 Warszawa",
        "shipping_address": "ul. Testowa 1 12-345 Warszawa",
        "products_net_total": 22,
        "products_tax_total": 0,
```

```
    "products_gross_total": 22,  
    "shipping_net_total": 22,  
    "shipping_gross_total": 22,  
    "shipping_method": "carrier",  
    "shipping_number": "AZ12346",  
    "shipping_tracking_link": "https://testtrack.com/AZ12346",  
    "ordered_at": "2023-07-05 11:54:03",  
    "shipped_at": "2023-07-05 11:55:59",  
    "updated_at": "2023-07-10 10:33:01",  
    "ordered_by_email": "sdsa"  
  }  
]  
}  
}
```

Aby pobrać informacje o pojedynczym zamówieniu należy skorzystać z query getOrder i podać order_id(zwracany przez getOrderList) jako parametr.

request:

```
{  
  getOrder(order_id: 1){  
    order_id  
    order_number  
    status  
    billing_address  
    shipping_address  
    products_net_total  
    products_tax_total  
    products_gross_total  
    shipping_net_total  
    shipping_gross_total  
    shipping_method  
    shipping_number  
    shipping_tracking_link  
    ordered_at  
    shipped_at  
    updated_at  
    ordered_by_email  
    items {  
      name  
      sku  
      index  
      net_price  
      gross_price  
      tax_rate  
      quantity
```

```
    quantity_shipped
    row_total_net
    row_total_gross
    row_total_tax
    unit
  }
}
```

response:

```
{
  "data": {
    "getOrder": {
      "order_id": 1,
      "order_number": "123",
      "status": "new",
      "billing_address": "ul. Testowa 1 12-345 Warszawa",
      "shipping_address": "ul. Testowa 1 12-345 Warszawa",
      "products_net_total": 22,
      "products_tax_total": 2,
      "products_gross_total": 22,
      "shipping_net_total": 22,
      "shipping_gross_total": 22,
      "shipping_method": "pickup",
      "shipping_number": "AZ12345",
      "shipping_tracking_link": "https://testtrack.com/AZ12345",
      "ordered_at": "2023-07-05 11:53:03",
      "shipped_at": "2023-07-05 11:54:59",
      "updated_at": "2023-07-25 06:20:52",
      "ordered_by_email": "companyuser@company.com"
      "items": [
        {
          "name": "Rozrusznik",
          "sku": "roz-1",
          "index": "roz-1",
          "net_price": 10,
          "gross_price": 12,
          "tax_rate": 23,
          "quantity": 1,
          "quantity_shipped": 1,
          "row_total_net": 10,
          "row_total_gross": 12,
          "row_total_tax": 2,
          "unit": "szt."
        },
      ],
    }
  }
}
```

```

    {
      "name": "Rozrusznik 2",
      "sku": "roz-2",
      "index": "roz-2",
      "net_price": 10,
      "gross_price": 12,
      "tax_rate": 23,
      "quantity": 1,
      "quantity_shipped": 1,
      "row_total_net": 10,
      "row_total_gross": 12,
      "row_total_tax": 2,
      "unit": "szt."
    }
  ]
}
}
}

```

4. Składanie zamówień

Aby złożyć zamówienie należy wykonać kilka kroków:

Stworzenie koszyka:

request:

```

mutation {
  createEmptyCart
}

```

response:

```

{
  "data": {
    "createEmptyCart": "cYqwKNqy5iWCBEC7V42aJB8FPhvjIASq"
  }
}

```

Dodanie produktu/produktów do koszyka

w przykładzie dodanie produktu typu prostego(w przypadku innych typów produktów szczegóły w dokumentacji, w zakładce mutations można znaleźć przykłady requestów dla

każdego typu produktu

<https://devdocs.magento.com/guides/v2.3/graphql/mutations/index.html>)

request:

```
mutation {
  addSimpleProductsToCart(
    input: {
      cart_id: "cYqwKNqy5iWCBEC7V42aJB8FPhvjIASq"
      cart_items: [
        {
          data: {
            quantity: 1
            sku: "Rozrusznik"
          }
        }
      ]
    }
  ){
    cart {
      items {
        id
        product {
          name
          sku
        }
        quantity
      }
    }
  }
}
```

response:

```
{
  "data": {
    "addSimpleProductsToCart": {
      "cart": {
        "items": [
          {
            "id": "39",
            "product": {
              "name": "Rozrusznik",
              "sku": "Rozrusznik"
            },
            "quantity": 1
          }
        ]
      }
    }
  }
}
```



```
    ]
  }
}
}
```

Pobranie Id adresów firmy

następnym krokiem jest ustawienie id adresów na koszyku, aby to zrobić musimy pobrać informacje o adresach.

Request:

```
{
customer {
  addresses {
    id
    default_billing
    default_shipping
    firstname
    lastname
    street
    city
    region {
      region_code
      region
    }
    postcode
    country_code
    telephone
  }
}
}
```

response:

```
{
  "data": {
    "customer": {
      "addresses": [
        {
          "id": 11,
          "default_billing": true,
          "default_shipping": true,
          "firstname": "K",
          "lastname": "J",
          "street": [
```

```

        "sdadsa"
    ],
    "city": "dasdsadsa",
    "region": {
        "region_code": "PL-16",
        "region": "opolskie"
    },
    "postcode": "1321312",
    "country_code": "PL",
    "telephone": "123123123"
    }
]
}
}
}
}

```

w przypadku większej liczby adresów zwracamy uwagę na pola default_billing i default_shipping, jako adres rozliczeniowy ustawiamy id adresu przy którym znajdziemy wartość default_billing=true, analogicznie z adresem wysyłki(shipping).

Mając id adresów możemy je przypisać do koszyka.

Ustawienie adresu billingowego koszyka

Request:

```

mutation {
  setBillingAddressOnCart(
    input: {
      cart_id: "cYqwKNqy5iWCBEC7V42aJB8FPhvjIASq"
      billing_address: {
        customer_address_id: 11
        same_as_shipping: false
      }
    }
  ){
    cart {
      billing_address {
        firstname
        lastname
        company
        street
        city
        region{
          code

```

```
    label
  }
  postcode
  telephone
  country{
    code
    label
  }
}
}
}
```

Response:

```
{
  "data": {
    "setBillingAddressOnCart": {
      "cart": {
        "billing_address": {
          "firstname": "K",
          "lastname": "J",
          "company": "testowafirma",
          "street": [
            "sdadsa"
          ],
          "city": "dasdsadsa",
          "region": {
            "code": "PL-16",
            "label": "opolskie"
          },
          "postcode": "1321312",
          "telephone": "123123123",
          "country": {
            "code": "PL",
            "label": "PL"
          }
        }
      }
    }
  }
}
```

Jeśli najpierw adres billingowy i shippingowy są takie same możemy najpierw ustawić adres shippingowy koszyka a później wywołać powyższy request wysyłając w obiekcie `billing_address` jedynie pole `same_as_shipping`: true

Ustawienie adresu shippingowego koszyka

Request:

```
mutation {
  setShippingAddressesOnCart(
    input: {
      cart_id: "cYqwKNqy5iWCBEC7V42aJB8FPhvjIASq"
      shipping_addresses:[{
        customer_address_id: 11
      }]
    }
  ){
    cart {
      shipping_addresses {
        firstname
        lastname
        company
        street
        city
        region{
          code
          label
        }
        postcode
        telephone
        country{
          code
          label
        }
      }
    }
  }
}
```

Response:

```
{
  "data": {
    "setShippingAddressesOnCart": {
      "cart": {
        "shipping_addresses": [
          {
            "firstname": "K",
            "lastname": "J",
            "company": "testowafirma",
            "street": [
```

```
    "sdadsa"
  ],
  "city": "dasdsadsa",
  "region": {
    "code": "PL-16",
    "label": "opolskie"
  },
  "postcode": "1321312",
  "telephone": "123123123",
  "country": {
    "code": "PL",
    "label": "PL"
  }
}
]
}
}
}
```

Sprawdzenie dostępnych metod płatności i wysyłki

ogólne zapytanie pobierające informacje o koszyku

Request:

```
{
  cart(cart_id: "cYqwKNqy5iWCBEC7V42aJB8FPhvjIASq") {
    email
    billing_address {
      city
      country {
        code
        label
      }
      firstname
      lastname
      postcode
      region {
        code
        label
      }
      street
      telephone
    }
  }
}
```

```
shipping_addresses {
  firstname
  lastname
  street
  city
  region {
    code
    label
  }
  country {
    code
    label
  }
  telephone
  available_shipping_methods {
    amount {
      currency
      value
    }
    available
    carrier_code
    carrier_title
    error_message
    method_code
    method_title
    price_excl_tax {
      value
      currency
    }
    price_incl_tax {
      value
      currency
    }
  }
  selected_shipping_method {
    amount {
      value
      currency
    }
    carrier_code
    carrier_title
    method_code
    method_title
  }
}
```

```
items {
  id
  product {
    name
    sku
  }
  quantity
}
available_payment_methods {
  code
  title
}
selected_payment_method {
  code
  title
}
applied_coupons {
  code
}
prices {
  grand_total {
    value
    currency
  }
}
}
```

Response:

```
{
  "data": {
    "cart": {
      "email": "krzysztof.jasniok@strix.net",
      "billing_address": {
        "city": "dasdsadsa",
        "country": {
          "code": "PL",
          "label": "PL"
        },
        "firstname": "K",
        "lastname": "J",
        "postcode": "1321312",
        "region": {
          "code": "PL-16",
          "label": "opolskie"
        }
      }
    }
  }
}
```

```
},
"street": [
  "sdadsa"
],
"telephone": "123123123"
},
"shipping_addresses": [
  {
    "firstname": "K",
    "lastname": "J",
    "street": [
      "sdadsa"
    ],
    "city": "dasdsadsa",
    "region": {
      "code": "PL-16",
      "label": "opolskie"
    },
    "country": {
      "code": "PL",
      "label": "PL"
    },
    "telephone": "123123123",
    "available_shipping_methods": [
      {
        "amount": {
          "currency": "PLN",
          "value": 5
        },
        "available": true,
        "carrier_code": "flatrate",
        "carrier_title": "Flat Rate",
        "error_message": "",
        "method_code": "flatrate",
        "method_title": "Fixed",
        "price_excl_tax": {
          "value": 5,
          "currency": "PLN"
        },
        "price_incl_tax": {
          "value": 6.15,
          "currency": "PLN"
        }
      }
    ]
  }
],
```



```
payment_method: {
  code: "checkmo"
}
){
cart {
  selected_payment_method {
    code
    title
  }
}
}
```

Response:

```
{
  "data": {
    "setPaymentMethodOnCart": {
      "cart": {
        "selected_payment_method": {
          "code": "checkmo",
          "title": "Check / Money order"
        }
      }
    }
  }
}
```

Ustawienie metody wysyłki w koszyku

Request:

```
mutation {
  setShippingMethodsOnCart(
    input: {
      cart_id: "cYqwKNqy5iWCBEC7V42aJB8FPhvjIASq",
      shipping_methods: [
        {
          carrier_code: "flatrate"
          method_code: "flatrate"
        }
      ]
    }
  ){
    cart {
```

```
shipping_addresses {
  selected_shipping_method {
    carrier_code
    carrier_title
    method_code
    method_title
    amount {
      value
      currency
    }
  }
}
}
```

Response:

```
{
  "data": {
    "setShippingMethodsOnCart": {
      "cart": {
        "shipping_addresses": [
          {
            "selected_shipping_method": {
              "carrier_code": "flatrate",
              "carrier_title": "Flat Rate",
              "method_code": "flatrate",
              "method_title": "Fixed",
              "amount": {
                "value": 5,
                "currency": "PLN"
              }
            }
          }
        ]
      }
    }
  }
}
```

Złożenie zamówienia

Po wykonaniu wszystkich wcześniejszych kroków koszyk jest gotowy by stać się zamówieniem.

Request:

```
mutation {
  placeOrder(
    input: {
      cart_id: "grfLPCjH6Oeqpq0q13Ssrx762gmAt5EI"
    }
  ){
    order {
      order_number
    }
  }
}
```

Response:

```
{
  "data": {
    "placeOrder": {
      "order": {
        "order_number": "000000032"
      }
    }
  }
}
```

5. Historia rozliczeń

Request:

```
{
  getSettlementHistory {
    merchant_credit {
      label
      value
    }
    sales_documents {
      document_number
      type
      status
      file_download_link_pdf
      file_download_link_edi
      file_download_link_xml
    }
  }
}
```

```
    billing_address
  settlements
}
}
}
```

Response:

```
{
  "data": {
    "getSettlementHistory": {
      "merchant_credit": [
        {
          "label": "Przyznana kwota kredytu",
          "value": "100 000.00PLN"
        },
        {
          "label": "Aktualne zobowiązania",
          "value": "30 000.00PLN"
        },
        {
          "label": "Dostępna kwota kredytu",
          "value": "70 000.00PLN"
        }
      ],
      "sales_documents": [
        {
          "document_number": "1234",
          "type": "Faktura VAT",
          "status": "W realizacji",
          "file_download_link_pdf":
"https://agrorami.test/pl/settlement/document/download/id/314/fileFormat/pdf/",
          "file_download_link_edi":
"https://agrorami.test/pl/settlement/document/download/id/314/fileFormat/edi/",
          "file_download_link_xml":
"https://agrorami.test/pl/settlement/document/download/id/314/fileFormat/xml/",

          "billing_address": "test street 4/33 43-100 Tychy",
          "settlements": "{\\"Kwota\\": \\"1000.00 PLN\\", \\"Zapłacono\\": \\"100.00 PLN\\", \\"Pozostało\\": \\"900.00
          PLN\\}"
        },
        {
          "document_number": "12345",
          "type": "Faktura VAT",
          "status": "W realizacji",
```

```

        "file_download_link_pdf":
"https://agrorami.test/pl/settlement/document/download/id/306/fileFormat/pdf",
        "file_download_link_edi": null,
        "file_download_link_xml": null,

        "billing_address": "test street 4/33 43-100 Tychy",
        "settlements": "{\"Kwota\": \"1000.00 PLN\", \"Zapłacono\": \"100.00 PLN\", \"Pozostało\": \"900.00
PLN\"}"
    },
    {
        "document_number": "13455",
        "type": "Faktura VAT",
        "status": "W realizacji",
        "file_download_link_pdf":
"https://agrorami.test/pl/settlement/document/download/id/313/fileFormat/pdf",
        "file_download_link_edi": null,
        "file_download_link_xml": null,

        "billing_address": "Testowy Adres Bill",
        "settlements": "{\"Kwota\": \"1000.00 PLN\", \"Zapłacono\": \"100.00 PLN\", \"Pozostało\": \"900.00
PLN\"}"
    }
]
}
}
}
}
}

```

Aby pobrać dokumenty rozliczeniowe należy użyć linków z pól 'file_download_link_*format*'.