

# [CPS] Api dla BOK w promocji "naklejkowej" dla JMP

## Historia modyfikacji:

Wersja	Data modyfikacji	Opis modyfikacji	Autor
1.0	2016-07-28	Draft	Tomasz Kruk
1.1	2016-07-29	Poprawa formatu numeru telefonu (bez +) Zwracanie aktualnego statusu klienta w odpowiedzi usługi modyfikującej dane klienta Zmiana sposobu wyznaczania znacznika danych klienta - ETag Dodanie metody do powiązania karty Single z kartą Single/Master Dodanie metody do usunięcia powiązania karty Slave z kartą Master	Romuald Bąk + Tomasz Kruk
1.2	2016-08-03	Zwracanie informacji o wersji portfeli przy pobieraniu klienta	Filip Frątczak

## 1. Uczestnicy procesu

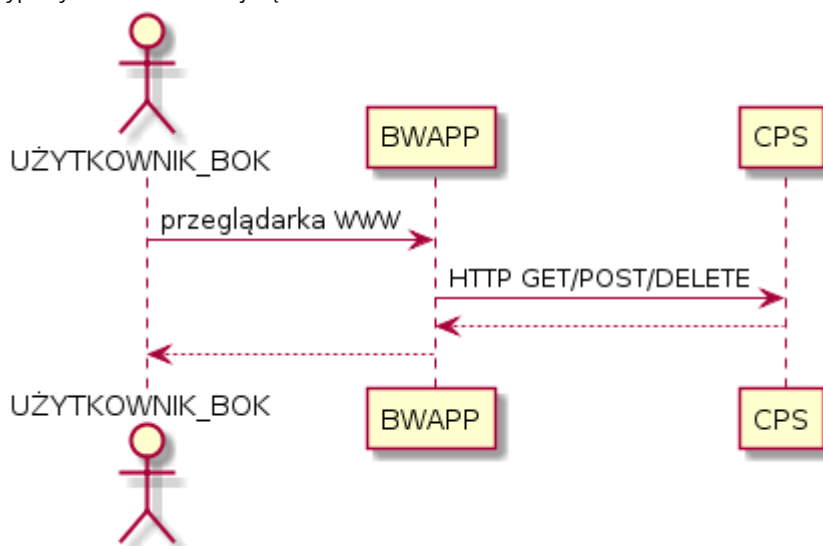
BOK WebApp (BWAPP) - aplikacja BOK, z której korzysta obsługa promocji "naklejkowej"

Customer Portal Server (CPS) - serwer aplikacji udostępniający usługi do wyświetlania i zarządzania danymi osobowymi klientów, wyświetlania stanu punktów zgromadzonych na karcie i zarządzania relacjami pomiędzy kartami.

Użytkownik - Użytkownik BOK

## 2. Komunikacja między uczestnikami procesu

Typowy schemat realizacji żądania



Komunikacja jest realizowana za pomocą usług HTTP REST. Dane użyte w żądaniach i odpowiedziach są serializowane z użyciem JSON.

Komunikacja wymaga zestawienia bezpiecznego połączenia między serwerami zabezpieczonego protokołem TLS 1.2.

Odpowiednie żądania i odpowiedzi powinny deklarować nagłówki: `Accept` i `Content-Type` z wartością: `application/json`.

Statusy błędów są zwracane za pomocą standardowych kodów HTTP Status Codes. W sytuacjach w których niezbędne jest dostarczenie dodatkowych informacji o przyczynie błędów, usługi zwracają rezultat w postaci:

```
{
  "HttpCode": "404",
  "MessageCode": "CardNotFound",
  "Message": "Nie znaleziono klienta o podanym numerze kartyu"
}
```

Pole MessageCode zawiera identyfikator, który może być wykorzystany przez konsumenta usług do wyświetlenia odpowiedniego komunikatu dla użytkownika. W przypadku nieobsłużonych błędów pole "MessageCode" może zawierać wartość "UnknownError" i wówczas pole "Message" zawiera komunikat błędu zwracany przez serwer. W pozostałych sytuacjach pole Message zawiera komunikat, który może być ignorowany, jeśli aplikacja rozpoznała MessageCode.

## 3. Udostępnione usługi

### 3.1. Pobranie danych klienta

#### Request

- GET <https://{host}/bok/loyalty/v1/loyalty-cards/{cardNumber}/customer>  
Żądanie pobiera dane klienta na podstawie numeru karty.
- GET <https://{host}/bok/loyalty/v1/customer-phones/{phoneNumber}/customer>  
Żądanie pobiera dane klienta na podstawie numeru telefonu.
- Nagówek HTTP:
  - **If-None-Match** (domylnie brak) - ETag odesany we wcześniejszej odpowiedzi usugi.  
Jeeli ETag po stronie usugi jest inny lub nie zostanie przesany to zwracane s aktualne dane klienta, w przeciwnym razie zwracany jest kod bdu 304 (Not modified).

#### Response

- Status = 200,  
Nagówek HTTP: **ETag** - znacznik danych klienta. Znacznik zmienia si wraz z modyfikacj danych klienta.  
Body:

```
{
  "FirstName":"Adam",
  "LastName":"Bialecki",
  "PhoneNumber":"48602123789",
  "Email":"adam234.bialecki223@bialecki.com",
  "Street":"Gajowa",
  "HouseNumber":"6",
  "ApartmentNumber":"8",
  "PostalCode":"00-950",
  "City":"Poznań",
  "Status":"Active",
  "PersonalDataProcessingConsent":true,
  "EmailCommunicationConsent":true,
  "SmsCommunicationConsent":false,
  "PreferredLanguage":"pl-PL",
  "PreferredStore":"3456",
  "LoyaltyCard":{
    "CardNumber":"9949837485921",
    "Wallets":[
      {
        "Id":"Stickers2016",
        "Balance":10.0,
        "Stat":{
          "Lost":0.22,
          "TransferredOut":132.00,
          "TransferredIn":9.12,
          "Issued":199.22
        }
      }
    ]
  }
}
```

```

"Version": 1
  },
  {
    "Id": "SomeOtherWallet",
    "Balance": 0.0,
    "Stat": {
      "Lost": 0.22,
      "TransferredOut": 132.00,
      "TransferredIn": 9.12,
      "Issued": 199.22
    }
  },
"Version": 2
  }
],
"MasterCardReference": {
  "CardNumber": "9949837485333",
  "OwnerFirstName": "Maria",
  "OwnerLastName": "Błaszczuk"
},
"SlaveCardReferences": [
  {
    "CardNumber": "9949837485333",
    "OwnerFirstName": "Maria",
    "OwnerLastName": "Błaszczuk",
    "Wallets": [
      {
        "Id": "Stickers016",
        "Balance": 10.0,
        "Stat": {
          "TransferredOut": 132.00
        }
      }
    ]
  }
],
{
  "CardNumber": "9949837485332",
  "OwnerFirstName": "Jan",
  "OwnerLastName": "Pachlowski",
  "Wallets": [
    {
      "Id": "Stickers2016",
      "Balance": 10.0,
      "Stat": {
        "TransferredOut": 132.00
      }
    }
  ]
}
],
},
"CustomerProperties": [
  {
    "Name": "Property1",
    "Value": "Value1",
    "ReadOnly": false,
    "ValidationRegex": "^a.*z$",
    "Required": true
  },
  {
    "Name": "Zgoda na otrzymanie nagrody w postaci bonu wysłanego pocztą elektroniczną",
    "Value": false,
    "ReadOnly": true
  }
],
"Messages": [

```

```
{
  "Id": "d939ae873ada4d0db805953b96299016",
  "Message": "Twój numer telefonu nie mógł być poprawnie zweryfikowany. Sprawdź czy podałeś poprawny numer telefonu lub skontaktuj się z BOK."
},
{
  "Id": "362ac541b22b4eeba0e7105fbdf5fe9d",
  "Message": "Zgromadziłeś wystarczającą liczbę punktów aby uzyskać nagrodę. Zapraszamy po odbiór do dowolnego sklepu Biedronka."
}
}
```

- "CustomerProperties" zawiera właściwości "dynamiczne" klienta, które użytkownik systemu Perspectiv może dodać z poziomu aplikacji. Ten mechanizm będzie używany do zarządzania dodatkowymi zgodami klienta na specyficzne procesy związane z obsługą programu lojalnościowego.

Możliwe wartości "Status":

- Active - aktywny,
- WaitingForActivationSmsConfirmation - czeka na aktywację,
- NotActivated - nieaktywowany,
- Disabled - zablokowany.

Możliwe wartości "Accounts.Name":

- AccumulatedStickers - obecna liczba naklejek do odebrania,
  - LostStickers - liczba naklejek jakie utraci klient (nie chciał ich odebrać w POS),
  - IssuedStickers - liczba naklejek wydanych klientowi
  - TransferredStickers - liczba punktów przeniesionych z karty *Slave* do karty *Master*.
- Status = 304, Dane klienta nie zostały zmodyfikowane,
- Body:

```
{
  "HttpCode": "304",
  "MessageCode": "CustomerNotModified",
  "Message": "Dane klienta nie zostały zmodyfikowane"
}
```

## 3.2. Modyfikacja danych klienta

### Request

- Nagówek HTTP:
    - **If-Match** - powinien zawierać **ETag** odesany podczas pobrania danych klienta, patrz punkt 3.3. Jeeli **ETag** po stronie usługi jest taki sam to dane klienta zostaną zmodyfikowane, w przeciwnym razie zwracany jest kod bdu 412 (Precondition failed).
    - **ETag** określa kolejną wersję danych klienta i jest numerowany od 0.
  - POST <https://{host}/bok/loyalty/v1/loyalty-cards/{cardNumber}/customer>  
Kompletna modyfikacja danych klienta,
- Body:

```
{
  "FirstName": "Adam",
  "LastName": "Bialecki",
  "PhoneNumber": "48602123789",
  "Email": "adam234.bialecki223@bialecki.com",
  "Street": "Gajowa",
  "HouseNumber": "6",
  "ApartmentNumber": "8",
  "PostalCode": "00-950",
  "City": "Poznań",
  "PersonalDataProcessingConsent": true,
  "EmailCommunicationConsent": true,
  "SmsCommunicationConsent": false,
  "PreferredLanguage": "pl-PL",
  "PreferredStore": "4532",
}
```

```

"CustomerProperties":{
  "Property1":"Value1",
  "Property2":true
}
}

```

- PATCH <https://{host}/bok/loyalty/v1/loyalty-cards/{cardNumber}/customer>  
Czciowa modyfikacja danych klienta,  
Body:

```

{
  "Email":"adam.bialecki@bialecki.com",
  "PreferredStore":""
}

```

## Response

- Status = 200  
Zmiana pól w kliencie (imi oraz numeru telefonu) moe mie wpyw na zmian statusu, dlatego w odpowiedzi zwracany jest aktualny status klienta.  
Body:

```

{
  "Status":"Active"
}

```

Możliwe wartości "Status":

- Active - aktywny,
  - WaitingForActivationSmsConfirmation - czeka na aktywację,
  - NotActivated - nieaktywowany,
  - Disabled - zablokowany.
- Status = 400, bdné dane - w danu istnieją pola, które nie zostały wyspecyfikowane w dokumentacji

```

{
  "HttpCode": "400",
  "MessageCode": "NotSupportedProperties",
  "Message": "Podane pola nie odpowiadają specyfikacji"
}

```

- Status = 412 - dane zostały zmodyfikowane przez innego użytkownika lub proces,  
Body:

```

{
  "HttpCode": "412",
  "MessageCode": "CustomerModifiedByAnotherUserOrProcess",
  "Message": "Dane klienta zostały zmodyfikowane przez innego użytkownika lub proces"
}

```

- Status = 422 - błąd walidacji. Body:

```

{
  "ValidationErrorCodes": [
    {"Property": "PhoneNumber", "Error": "Invalid", "Value": "AAAA"},
    {"Property": "FirstName", "Error": "Missing"},
    {"Property": "CustomerProperties.Property1", "Error": "Invalid", "Value": "Value1"}],
}

```

Możliwe wartości "ValidationErrorCodes.Error":

Missing - brak wymaganej danej,

Invalid - błędny format,

InvalidReadOnlyProperty - pole z CustomerProperties jest tylko do odczytu,

OverridingReadOnlyProperty - pole z CustomerProperties jest niezgodne z oczekiwanym typem

"Value" zawiera wartość powodującą błąd walidacji.

Powiązanie karty Single z kartą Single/Master

### 3.3. Powiązanie karty Single z kartą Single/Master

#### Request

- POST `https://{host}/bok/loyalty/v1/loyalty-cards/{cardNumber}/master-card/{MasterCardNumber}`
- Body - brak

#### Response

- Status = 204
- Status = 404 - klient identyfikowany po MasterCardNumber nie istnieje. Body:

```
{
  "HttpCode": "404",
  "MessageCode": "CustomerNotFoundByMasterCardNumber",
  "Message": "Klient o zadanym numerze karty nie istnieje"
}
```

- Status = 422 - błąd walidacji. Body:

```
{
  "HttpCode": "422",
  "MessageCode": "SlaveCustomerCannotBecomeMaster",
  "Message": "Karta o zadanym numerze karty jest kartą Slave"
}
```

Możliwe wartości "MessageCode":

SlaveCustomerCannotBecomeMaster

SlaveCountLimitExceeded (przekroczona konfigurowalna ilość (domyślnie 10) kart Slave powiązanych z zadaną kartą Master)

### 3.4. Usunięcie powiązania karty Slave z kartą Master

#### Request

- DELETE `https://{host}/bok/loyalty/v1/loyalty-cards/{cardNumber}/master-card/`

#### Response

- Status = 204

### 3.5. Zmiana hasła klienta

#### Request

- POST `https://{host}/bok/loyalty/v1/loyalty-cards/{cardNumber}/customer/password`

Body:

```
{
  "Password": "Qwerty1@"
}
```

#### Response

- Status = 204