

[CPS] Api do zarządzania klientami dla ALCS'a

Historia modyfikacji:

Wersja	Data modyfikacji	Opis modyfikacji	Autor
1.0	2016-07-19	Draft	Filip Frątczak
1.1	2016-07-20	Dostosowanie api do standardów używanych w ALCS. Rozszerzenie body żądania "Rejestracja karty klienta" o dane karty lojalnościowej. Dodanie numeru telefonu do body żądania aktywacji karty.	Romuald Bąk
1.2	2016-08-03	Dodane wersjonowanie portfeli	Romuald Bąk

1. Uczestnicy procesu

Customer Portal Server (CPS) - serwer aplikacji udostępniający usługi do zarządzania danymi osobowymi klientów

Ares Loyalty Customer Solution (ALCS) - serwer autoryzacyjny obsługujący procesy związane z obsługą kart klienta

2. Komunikacja między uczestnikami procesu

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.

Nie jest wymagana autoryzacja w usłudze aby rozpocząć przetwarzanie.

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": "403",
  "MessageCode": "NoAuthothorization",
  "Message": "Błędny numer karty lub pin"
}
```

Pole `MessageCode` zawiera identyfikator, który może być wykorzystany przez konsumenta usług do wyświetlenia odpowiedniego komunikatu dla klienta. 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. Rejestracja karty klienta

Usługa będzie wywoływana przez ALCS w momencie wydawania karty klientowi.

Usługa będzie wykonywała następujące operacje:

- założenie klienta w bazie Perspectiv'a
- aktualizacja danych klienta w Redis zgodnie z [Redis jako cache klientów](#). Klient będzie zakładany jako nieaktywny, nie muszą być

zatem wypełnione wszystkie dane obowiązkowe

- aktualizacja karty klienta w Redis zgodnie z [Redis jako cache klientów](#). Karta będzie zakładana jako nieaktywna.

Request

- PUT `http://{host}/alcs/loyalty/v1/loyalty-cards/{card-number}`
- Body

```
{
  "Identity":
  {
    "PosId": "09fae9a5d4ea40e79dd7c1096c3694e0", // opcjonalny dodatkowy identyfikator kasy (np. GUID)
    "PosNumber": "5", // numer kasy - int
    "PosOperatorId": "b614f1dbda36454b970b4e1d409f4b01", // opcjonalny dodatkowy identyfikator
operatora (np. GUID); pusta wartość lub brak, gdy kasjer nie jest zalogowanyzalogowany
    "PosOperatorNumber": 23, // numer operatora (kasjera); 0 lub brak, gdy kasjer nie jest
    "SiteId": "2345" //identyfikator sklepu (domyślnie: numer sklepu - wartość alfanumeryczna)
  },
  "ResponseLanguage":
  {
    "ClientLanguage": "pl_PL",
    "CashierLanguage": "en_US"
  },
  "LoyaltyCard":
  {
    "CardNumber": "99419283746389"
    "PhoneNumber": "+48123456789", // może być null
    "Gender": "{M/F/U}",
    "CardPinHMac": "3f729ea140a769400985ade930f695452cf622fd0dd3ca69218319333ca2386c",
    "Wallets": [
      {
        "Id": "Strickers2016",
        "Balance": 10.0,
        "Stat": {
          "Lost": 0.22,
          "TransferredOut": 132.00,
          "TransferredIn": 9.12,
          "Issued": 199.22
        },
        "Version": "abc",
      },
      {
        "Id": "SomethingElse",
        "Balance": 0.0,
        "Stat": {
          "Lost": 0.0,
          "TransferredOut": 0.0,
        },
        "Version": "xyz",
      }
    ],
  }
}
```

Response

- Status = 200, Body:

```
{
  "CustomerId": "df90170f3eb44124a22fa12031d25ecb",
}
```

Rejestracja przebiegła poprawnie. Rezultat zawiera id klienta, który do którego została przypięta karta.

- Status = 4??, Body:

```
{
  "HttpCode": "4??",
  "MessageCode": "CardAlreadyRegistered",
  "Message": "Karta jest już zarejestrowana"
}
```

Możliwe wartości "MessageCode":

CardAlreadyRegistered

3.2. Aktywacja karty

Usługa będzie wywoływana przez ALCS w momencie potwierdzenia dostarczenia sms'a pod zadany numer telefonu.

Usługa będzie wykonywała następujące operacje:

- aktualizacja statusu klienta w bazie Perspectiv na Active
- aktualizacja statusu klienta w Redis (patrz [Redis jako cache klientów](#))
- aktualizacja statusu danych karty klienta zgodnie z [Redis jako cache klientów](#).
- aktywacja karty w ALCS'ie ([\[CPS\] Api do zarządzania klientami dla ALCS'a](#))

Request

- POST `http://{host}/alcs/loyalty/v1/loyalty-cards/{card-number}/activate`
- Body

```
{
  "PhoneNumber": "+48123456789"
}
```

Response

- Status = 204
Aktywacja przebiegła pomyślnie (Aktywowano kartę lub karta była już wcześniej aktywowana)
- Status = 400, Body:

```
{
  "HttpCode": "400",
  "MessageCode": "CustomerWithGivenCardNumberNotFound",
  "Message": "Nie odnaleziono klienta o podanym numerze karty"
}
```

Podano błędny numer karty.

```
{
  "HttpCode": "400",
  "MessageCode": "IncorrectPhoneNumber",
  "Message": "Błędny numer telefonu"
}
```

Numer telefonu nie zgadza się z numerem podanym wcześniej przez klienta

- Status = 422, Body:

```
{
  "HttpCode": "422",
  "MessageCode": "CardWasNotWaitingForActivation",
  "Message": "Karta nie oczekiwała na aktywację"
}
```

Karta o podanym numerze nie czekała na aktywację (jest nieaktywna lub zablokowana)

- Status = 500

```
{
```

```
"HttpCode": "500",
"MessageCode": "CustomerActivationError",
"Message": "Błąd podczas aktywacji klienta"
}
```

Inny, niezidentyfikowany błąd przy aktywacji klienta

3.3. Modyfikacja numeru telefonu klienta posługującego się kartą

Zgodnie z wymaganiem nr 23 z dokumentu: ... Loyalty- process description v20, jeśli klient na stronie www wpisał niepoprawny numer telefonu, to kasjer powinien przy następnej transakcji klienta otrzymać informację "Pobierz numer telefonu od klienta". Po wprowadzeniu numeru telefonu przez kasjera, ALCS powinien zaktualizować numer telefonu klienta wykorzystując niniejszą usługę.

Request

- PATCH `http://{host}/alcs/loyalty/v1/loyalty-cards/{card-number}`
- Body

```
{
  "PhoneNumber": "+48123456789"
}
```

Response

- Status = 204
Zmiana numeru telefonu przebiegła pomyślnie
- Status = 400, Body:

```
{
  "HttpCode": "400",
  "MessageCode": "CustomerWithGivenCardNumberNotFound",
  "Message": "Nie odnaleziono klienta o podanym numerze karty"
}
```

Podano błędny numer karty.

3.4. Zablokowanie karty klienta

Request

- POST `http://{host}/alcs/loyalty/v1/loyalty-cards/{card-number}/lock`

Response

- Status = 204
Deaktywacja przebiegła pomyślnie