# UPORABA BANKARTOVEGA PAYMENT GATEWAY-A – NAVODILA ZA TRGOVCE

december 2025

# Kazalo

# 1. SLOVAR

**Avtentikacija (3DS):** Postopek preverjanja identitete uporabnika, s čimer se zagotovi, da ima uporabnik pravico za izvedbo transakcije.

**Avtorizacija:** Proces preverjanja podatkov o kartici, o limitih in če ima stranka na bančnem računu dovolj sredstev za nakup, ki ga želi opravit itd.. Ta proces se načeloma zgodi po tem, ko je bila avtentikacija uspešno opravljena, v določenih izjemah pa tudi brez uspešne avtentikacije.

**Payment Gateway**: Sistem, ki trgovcem omogoča oz. poenostavi sprejemanje plačil s plačilnimi karticami ali takojšnjimi plačili (Flik,..). Je sistem ki povezuje spletna mesta(trgovine) z izdajatelji plačilnih metod, in prevzemniki plačilnih metod. Payment Gateway tudi poskrbi za povezovanje s sistemi za preverjanje istovetnosti stranke. Payment Gateway ima tudi spletni vmesnik za lažji pregled opravljeni transakcij

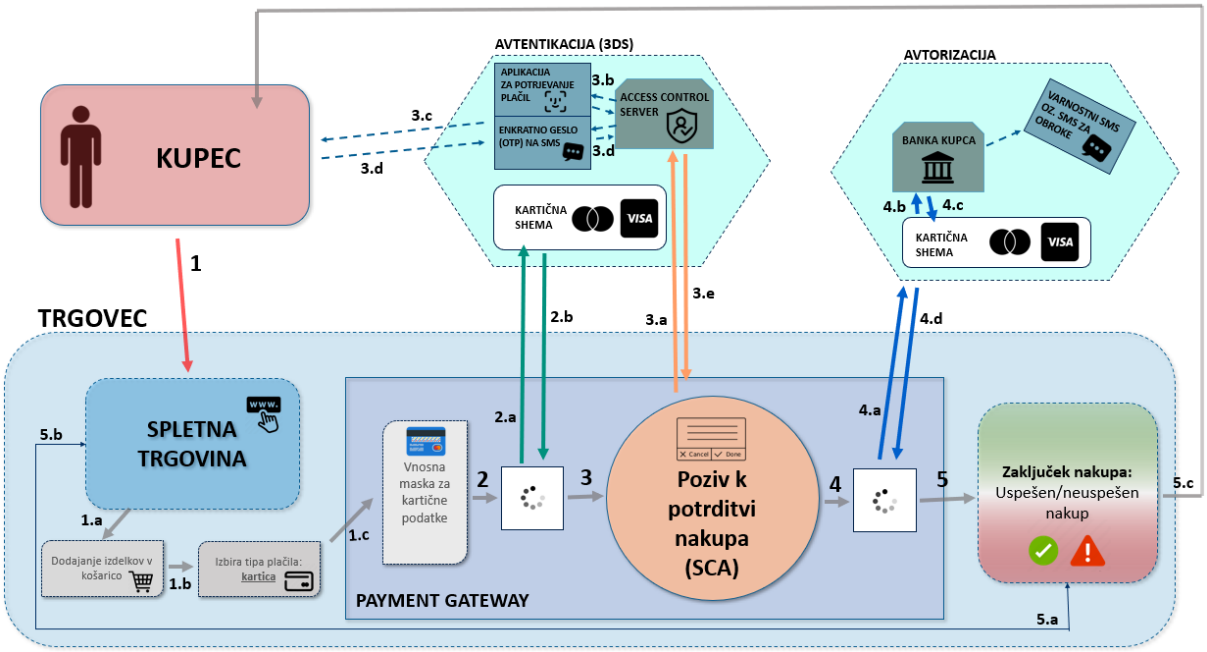
**Access Control Server (ACS):** Sistem, ki se na strani bank izdajateljic kartic uporablja za avtentikacijo imetnikov kartic z več različnimi metodami.

# 2. UVOD

Dokument je namenjen trgovcem, ki uporabljajo Bankartov Payment Gateway za sprejem kartičnih plačil. Dokument podaja informacije potrebne za integracijo Payment Gateway-a, poleg katerih podaja tudi splošne informacije o kartičnih transakcijah in navodila za testiranje.

## 2.1. Potek spletne transakcije

Spodaj je slika sheme okvirnega poteka spletne transakcije, za lažje razumevanje pa je pod sliko opis posameznega koraka spletne transakcije.




*Slika 1: Potek spletne transakcije*

### Koraki poteka spletne transakcije:

1. **Kupec** brska v spletni trgovini in išče izdelke:
    a. doda izdelke v košarico;
    b. izbere način plačila, ki je v tem primeru plačilna kartica in potrdi nakup;
    c. trgovec s klicem API-ja pokliče **Bankartov Payment Gateway** in prikaže se vnosna maska za vnos kartičnih podatkov kupca.
2. Po vnosu kartičnih podatkov se preveri istovetnost kartičnih podatkov (*Card Enrollment*):
    a. Payment Gateway se poveže s kartično shemo, kjer pridobi podatke o kartici, o verziji 3DS protokola in podatek, ali je sploh vključena v 3DS sistem;
    b. kartična shema podatke posreduje nazaj na Payment Gateway.
3. Payment Gateway se poveže s sistemom za *avtentikacijo plačila:*
    a. če gre za transakcijo, za katero ni zahtevana močna avtentikacija kupca, se transakcija izvede brez močne avtentikacije kupca;

   b. če je potrebna močna avtentikacija, ACS pošlje zahtevo po avtentikaciji v kupčevo aplikacijo za potrjevanje plačil (običajno v obliki potisnega obvestila) ali pa se na tel. številko pošlje edinstveno geslo (OTP);
   c. kupec opravi avtentikacijo;
   d. sistem za avtentikacijo informacijo o rezultatu avtentikacije posreduje ACS sistemu;
   e. ACS posreduje informacijo o rezultatu avtentikacije na Payment Gateway.
4. V naslednjem koraku se začne *avtorizacija:*
   a. Payment Gateway pošlje zahtevo po avtorizaciji na kartično shemo;
   b. ta posreduje zahtevo na banko kupca, kjer se nakup avtorizira;
   c. banka kupca vrne odgovor na kartično shemo, hkrati pa se sproži proces za pošiljanje varnostnega SMS-a (če je storitev vklopljena) in obročnega SMS-a (če gre za obročni nakup) kupcu;
   d. kartična shema vrne odgovor Payment Gateway-u.
5. Payment Gateway pokliče URL za prikaz zaslona o uspešnosti transakcije:
   a. Payment Gateway vrne rezultat transakcije trgovcu;
   b. trgovec potrdi prejem rezultata;
   c. kupcu se prikaže zaslon o uspehu/neuspehu transakcije.

# 3. TEHNIČNA INTEGRACIJA KARTIČNE PLAČILNE METODE (NAVODILA ZA RAZVIJALCE)

## 3.1. Glavni koraki uvedbe podpore

1. Podpis pogodbe med trgovcem in banko;
2. Posredovanje podatkov s strani banke v Bankart, kjer je izveden vnos trgovca v sistem podpore spletnemu poslovanju;
3. Posredovanje podatkov za potrebe uvedbe podpore na kontaktno mobilno telefonsko številko in e-mail naslov trgovca (gesla za dostop do Payment Gateway portala in povezavo API/WEB uporabnika);
4. Trgovčevi razvijalci izvedejo integracijo s Payment Gateway-om;
5. Izvedba testov, ki pokrivajo dogovorjen obseg poslovanja (avtorizacija, zajem, storno, obroki, pravilen prikaz obvestil ob odobreni in zavrnjeni transakciji …);
6. Banka registrira trgovca preko MasterCard Connecta v sistem EMV 3D Secure. V primeru Vise registracija ni potrebna, ker se ob prijavi v Bankart sisteme podatki že avtomatsko pošljejo tja;
7. Trgovec obvesti Bankart o uspešno izvedenih testih, Bankart preveri rezultate še z vidika podpore procesnega centra;
8. Posredovanje podatkov za dostop do produkcijskega okolja trgovcu.

## 3.2. Tehnična dokumentacija in zahteve

Splošna dokumentacija Payment Gateway-a je dostopna na spletni strani: https://gateway.bankart.si/documentation/gateway
API dokumentacija za povezovanje in integracijo je dostopna na spletni strani: https://gateway.bankart.si/documentation/apiv3?php#transaction-api-v3-0 (JSON)

Vsi API klici morajo vsebovati podpis, ki se generira na podlagi poslane vsebine in deljene skrivnosti. Povratni (callback) klici so prav tako podpisani, tako lahko sami preverite verodostojnost prejete vsebine. API klici se na strani Payment gateway-a lahko v delu neobveznih polj spreminjajo. Podpora na strani trgovca mora biti narejena na način, da sprememba ne bo imela vpliva na delovanje. Tu je nujno poudarit poudariti, da trgovec pri iniciaciji transakcije z API klicem v JSON-u, ne sme pozabit poslati polja callbackURL, saj na ta URL naslov dobi rezultat transakcije, ko je le ta zaključena.

Vzorčna koda v programskih jezikih PHP, Java, C# (.NET Framework) in vtičniki za zadnje verzije platform WooCommerce oziroma Wordpress (v4 in v5), Prestashop (v1.7 – prejšnje verzije niso podprte), Magento (v2) in Opencart (v3 – v2 ni podprta) je dostopna na spletnem mestu: povezava.

## 3.3. Podpora procesiranju transakcij na strani trgovca

Za komunikacijo med trgovčevo spletno stranjo in Payment Gateway-em se uporablja SSL komunikacija. Trgovec si mora za te potrebe na svoji strani zagotoviti ustrezen strežniški certifikat, kateri je splošno podprt.

**POMEMBNO**: trgovec oziroma njegova informacijska podpora mora poskrbeti za pravočasno zamenjavo strežniškega certifikata še pred potekom veljavnosti, saj ustrezna podpora brez veljavnega

certifikata ni mogoča. Zahtevana je spremljava produkcije po narejeni spremembi na strani trgovca. V kolikor trgovec ob spremljavi ne dobi enakih rezultatov procesiranja, kot so bili pred izvedeno spremembo, naj svoj sistem vrne v prejšnje stanje in o tem obvesti Bankart (customer.support@bankart.si).

## 3.4. Pridobivanje kartičnih podatkov od kupca

Payment Gateway podpira različne načine za pridobivanje podatkov o kartici s strani kupca. Trgovec lahko v ta namen uporabi predpripravljeno plačilno stran z vnosno masko ali pa izvede integracijo na svojo spletno stran. Možno je tudi shranjevanje podatkov o plačilni kartici, pri čemer se občutljivi podatki o kartici shranijo na strani Payment Gateway-a, trgovec pa rokuje le s token-om posamezne kartice. Bolj podroben opis je podan v nadaljevanju.

Za sam postopek avtentikacije in avtorizacije transakcije so pomembni spodaj navedeni podatki kupca, ki jih pridobi trgovec iz uproabniškega računa ali ob vpisu kontaktnih podatkov, pred vnosom kartičnih podatkov Priporočljivo je, da trgovec preverja, ali jih je kupec zapisal v pravilni obliki, saj je tako transakcija bolj verjetno uspešno izvedena.

| Ime polja | Opis |
|---|---|
| **billAddrCity** | Mesto prebivališča.<br>Format: spremenljiva dolžina, maximum 50 znakov |
| **billAddrCountry** | Država prebivališča.<br>Format: max. 3 znake, JSON Data Type: String, potrebno napolnit s številčno 3-mestno kodo pod standardu ISO 3166-1 (izjeme so v tabeli Table A.5 v EMV 3DS specifikaciji) |
| **billAddrLine1** | Naslov prebivališča.<br>Format: spremenljiva dolžina, max. 50 znakov, Data Type: String |
| **billAddrPostCode** | Poštna številka prebivališča.<br>Format: spremenljiva dolžina, max. 16 znakov, Data Type: String |
| **email** | E-mail naslov plačnika.<br>Format: spremenljiva dolžina, max. 254 znakov, potrebno napolniti s podatki, ki izpolnjujejo zahteve v tabeli v poglavju 3.4 znotraj dokumenta IETF RFC 5322. |
| **cardholderName** | Ime lastnika kartice.<br>Format: spremenljiva dolžina, 2–45 znakov (preverja Bankart) |

### 3.4.1. Oblikovanje plačilne strani z vnosno masko

Na plačilno stran (HPP – Hosted Payment Page), ki prikaže vnosno masko za kartične podatke in se prikaže v primeru preusmeritve iz trgovčeve strani, lahko dodamo trgovčev logotip. Tega naj trgovec v *.png* formatu posreduje na customer.support@bankart.si pred vključitvijo v produkcijo. Največja dovoljena velikost slike je *500px* x *130px*. Za logotipe vključenih plačilnih shem se obrnite na vašo banko.

Trenutno je HPP stran lahko prikazana v 19 različnih jezikih. To so: slovenščina, angleščina, nemščina, italijanščina, bolgarščina, bosanščina, češčina, slovaščina, španščina, hrvaščina, madžarščina, črnogorščina, makedonščina, poljščina, romunščina, ruščina, albanščina, srbščina in turščina. Za možnost dodatnih jezikov se lahko obrnete na vašo banko.

### 3.4.2. Plačilna stran po meri

Trgovcem, ki želijo plačilno stran po meri, predlagamo **payment.js integracijo**, o kateri je več napisano v API dokumentaciji, ki je razvijalcem prosto dostopna na spletni strani: https://gateway.bankart.si/documentation/gateway#paymentjs-javascript-integration

Če trgovec prehaja iz HPP maske na payment.js integracijo, ali se za to odloči že na začetku integracije, je potrebno izvesti dodatne teste API klicev, ki omogočajo to implementacijo. Za ureditev tega postopka se trgovec obrne na Bankartovo podporo za stranke (customer support)

## 3.5. Tipi podprtih transakcij na Payment Gateway-u

Trgovec se z banko dogovori in v pogodbi določi, katere vrste transakcij želi uporabljati. Nato implementira in testira le tiste transakcije, ki so mu jih banka odobrila v skladu s predhodnim dogovorom.

### 3.5.1. Osnovni tipi podprtih transakcij

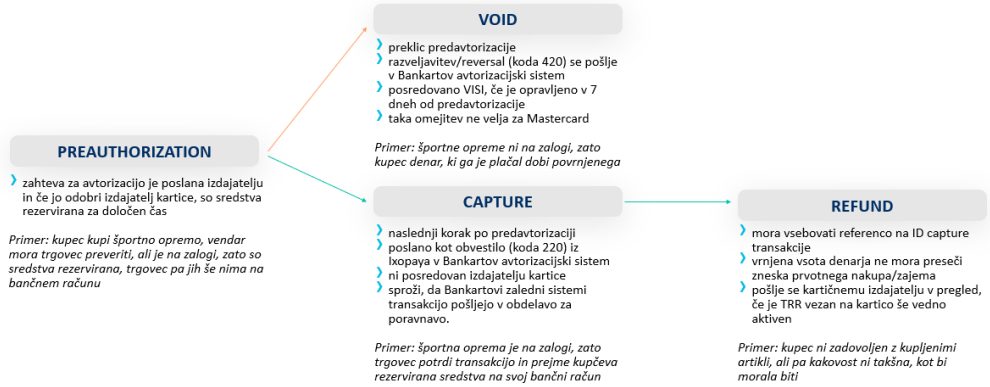
- **DEBIT (normalni spletni nakup):**
  Ta tip eno-fazne transakcije običajno uporabljajo trgovci, ki prodajajo digitalno blago, vstopnice in druge predmete, ki so strankam na voljo takoj po zaključku plačila. Trgovec ne more razveljaviti teh (končnih) avtorizacij zneskov, ki so označene, da se na koncu dneva vključijo v obdelavo obračuna in poravnave.
- **REFUND (vračilo denarja za spletni nakup):**
  Če trgovec želi narediti vračilo plačanega zneska, zaradi nezadovoljstva stranke ali drugih težav, mora sprožiti ta tip transakcije. Refund transakcije morajo vedno imeti referenco na finančno transakcijo (nakup oz. debit, zajem oz. capture) in ne morajo presegati zneska prvotne transakcije.

  DEBIT + REFUND shema:


*Slika 2: Prikaz poteka DEBIT+REFUND flowa*

- **PREAUTHORIZE + CAPTURE in VOID + REFUND (predavtorizacija + zajem zneska in preklic nakupa):**
  Trgovec, ki prodaja fizično blago, bo morda želel najprej preveriti zalogo, preden potrdi plačilo. Če trgovec ugotovi, da ne more izpolniti naročila, lahko razveljavi avtorizacijo in sprosti rezervirana sredstva na računu imetnika kartice (izvede VOID). Zato temu pravimo dvo-fazna transakcija.
  Transakcija je tako razdeljena na dva dela, (pred)avtorizacijo in kasnejši ukaz CAPTURE (zajem), ki pokaže, da je treba transakcijo vključiti v obdelavo obračuna in poravnave na koncu dneva.
  Zajem je treba izvesti v enem tednu (odvisno od kartične sheme), da se izognete večjemu tveganju povratnih bremenitev. Ko je transakcija zajeta, je ni mogoče razveljaviti, vendar je mogoče izvesti REFUND (vračilo denarja). Če se naredi delni zajem, je potrebno za preostali del zneska naredit VOID, da se prvotna predavtorizacija zapre in kupec dobi preostanek zneska, ki ga je plačal.

PREAUTHORIZE + CAPTURE/VOID + REFUND shema:

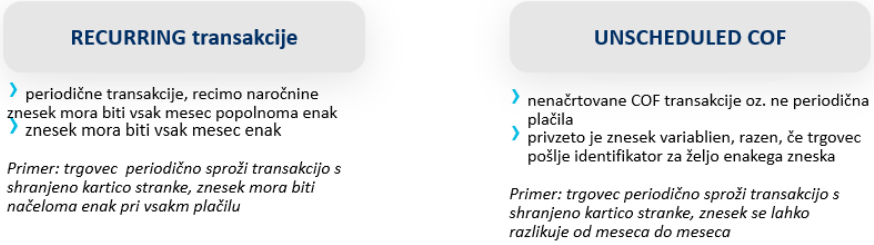



*Slika 3: Prikaz poteka PREAUTH+VOID/CAPTURE flowa*

### 3.5.2. Tipi podprtih transakcij z možnostjo hranjenja kartice

Ti tipi transakcij trgovcem omogočajo, da ponovno uporabijo podatke o kartici za prihodnje transakcije (tako imenovani COF oz. Card on File), ne da bi od imetnika kartice zahtevali, da ponovno vnese številko kartice in datum poteka veljavnosti. Obe transakciji se obvezno začneta s strani imetnika kartice, pri čemer imetnik kartice ob vpisu podatkov izvede močno avtentikacijo. Trgovec mora z banko najprej podpisati pogodbo, s katero mu banka sploh omogoča možnost izvedbe transakcij s shranjeno kartico. Šele ko je urejen pravno formalni del, trgovec lahko označuje, testira ter ponuja transakcije s shranjeno kartico.

- **IMETNIK KARTICE SPROŽI TRANSAKCIJO (*CARDHOLDER INITIATED TRANSACTIONS oz. CIT*):**
  Trgovec imetniku kartice ponuja plačilo s shranjeno kartico. Trgovec prejme maskiran PAN v prvotni transakciji, ko je bila kartica shranjena, in jo lahko uporabi za prikaz ter ponudi shranjene plačilne poverilnice imetniku kartice.
  Te transakcije so enake kot običajne transakcije, le da imetnik kartice pri nadaljnjih transakcijah ne vnese številke kartice ročno, ampak jo pridobi iz sistema varnega shranjevanja. Imetnik kartice je pri prvotni transakciji pozvan k preverjanju pristnosti (avtentikaciji) preko EMV 3DS, kot pri vsakem drugem spletnem nakupu.
- **TRGOVEC SPROŽI TRANSAKCIJO (*MERCHANT INITIATED TRANSACTIONS oz. MIT*):**
  V kontekstu e-trgovine so tukaj mišljene ponovljene (RECURRING) transakcije, npr. periodično zaračunavanje naročnine za revije. Pri reccuring transakcijah je znesek vsakokrat enak.
  Druga vrsta MIT transakcij so nenačrtovane Card on File transakcije (UNSCHEDULED COF), kar pomeni neperiodična plačila, ki se privzeto izvedejo z variabilnim zneskom, razen če trgovec pošlje v sporočilu dodaten indikator za zahtevo enakega zneska.

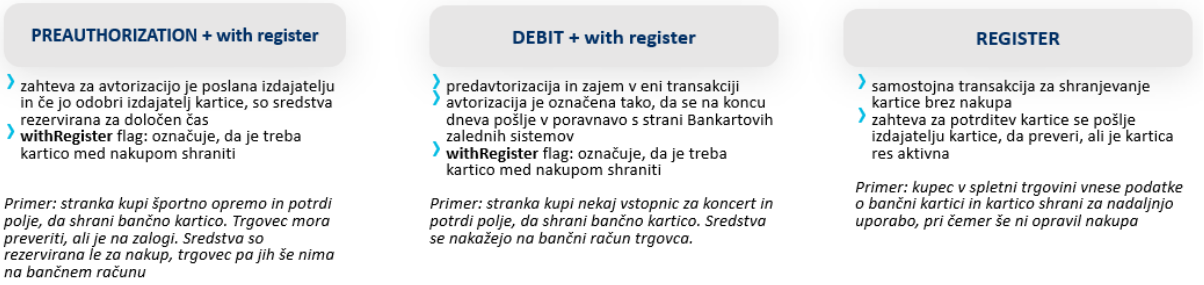



*Slika 4: Shema delitve ponavlajočih se plačil*

### 3.5.3. Shranjevanje kartice

Kartica se lahko shrani na več načinov:

- **REGISTER (shranjevanje kartice brez nakupa):**
  Kupec vnese podatke o kartici na spletno trgovino, ki ponuja možnost shranjevanja kartice. Najprej se izvede avtentikacija kupca, nato pa se pošlje zahteva za potrditev aktivnosti kartice izdajatelju kartice. Po potrditvi se shrani referenca te transakcije za nadaljnjo uporabo kartice.
- **DEBIT z »withRegister« flagom (normalni spletni nakup z dodatnim indikatorjem):**
  Klasični spletni nakup, kjer stranka izbere, da bi ob plačilu rada še shranila kartico za nadaljnje nakupe. Stranka se mora avtenticirati, podatki o kartici pa se shranijo na Bankartov Payment Gateway, kot se shrani tudi referenca na to prvo transakcijo.
- **PREAUTHORIZE z »with Register« flagom (predavtorizacija z dodatnim indikatorjem):**
  Trgovec, ki prodaja fizično blago, bo morda želel najprej preveriti zalogo, preden potrdi plačilo. Stranka ob vnosu podatkov o kartici izbere še opcijo, da bi rada shranila izbrano kartico za nadaljnje nakupe. Stranka se mora avtenticirati, podatki o kartici in referenca na prvo transakcijo pa se shranijo na Bankartov Payment Gateway.
- **DEREGISTER (odstranjevanje shranjene kartice)**
  To transakcijo lahko izvede ali trgovec ali stranka. S tem tipom transakcije se odstrani že shranjeno kartico pri nekem trgovcu. Stranka lahko odstrani kartico, če kartice pri trgovcu ne želi več uporabljati, trgovec pa jo odstrani v primeru, da stranka shrani novo kartico brez da bi staro, ki je ne uporablja več, izbrisala. Trgovec je odgovoren za izbris vseh podatkov o karticah, ki jih ne potrebuje več (tudi v primeru prenehanja poslovanja). Za izvedbo te transakcije, mora biti znotraj API klica poslana referenca na prvo transakcijo v seriji (polje reference UUID, ki je omenjeno v naslednjem podpoglavju).




V nadaljevanju je prikazano, kako morajo biti označeni posamezni tipi transakcij, ki imajo možnost hranjenja kartice (CoF , MiT in pa recurring). Priložene so tako tabele, ki označujejo glavne indikatorje, kot tudi primeri JSON klicev.

#### 3.5.3.1. Označevanje CoF (Card on File) transakcij:

| JSON | INITIAL - AMT 0 | | INITIAL - AMT <>0 | SUBSEQUENT |
|---|---|---|---|---|
| Type | COF | | COF | CIT-COF |
| Card number | enter | | enter | stored |
| transaction Type | REGISTER | PREAUTHORIZE | PREAUTHORIZE/DEBIT | PREAUT/DEBIT |
| transactionIndicator | \ | SINGLE | SINGLE | CARDONFILE |
| referenceTransactionId (referenceUuid) | \ | \ | \ | UUID(register) |
| withRegister | \ | TRUE | TRUE | \ |
| authenticationIndicator | \ | 04 | 04 | \ |
| recurringFrequency | \ | \ | \ | \ |
| challengeIndicator | 04 | 04 | 04 | \ |
| authenticationIndicator, recurringFrequency in challengeIndicator se pošljejo v ThreeDSecureData | | | | |

*Slika 5: Tabela z indikatorji za CoF transakcije*

Znotraj CoF transakcij imamo 3 vrste možnih izvedb teh transakcij:

- **Začetna (Initial)**, kjer je znesek enak 0:
  Gre za transakcijo sproženo s strani kupca (CIT), kjer je potrebno vnesti podatke o kartici.
  - V polju »*transaction Type*« je potrebno transakcijo označit kot REGISTER, s čimer se kartico samo shranjuje. Lahko pa se izbere tudi PREAUTHORIZE, kjer indikatorji potem razlikujejo od REGISTER transakcije
  - V primeru REGISTER transakcije je polje »withRegister« prazno. V primeru uporabe PREAUTHORIZE transakcije pa je vrednost v polju »withRegister« TRUE.
  - Polje »*authenticationIndicator*« se pri PREAUTHORIZE transakciji nastavi na 04. V primeru REGISTER se polje pusti prazno. Vrednost »04« pomeni, da se kartica samo dodaja, brez vzpostavitve recurring/MIT serije.
  - Polje *»challengeIndicator«* se polni z vrednostjo »04«, kar pomeni, da je ob vsaki začetni transakciji iz strani kartičnih shem zahtevana močna avtentikacija (SCA).
  - Ostalih zgoraj označenih polj, ni potrebno napolniti.

- **Začetna (Initial)**, kjer je znesek različen od 0:
  Gre za transakcijo sproženo s strani kupca (CIT), kjer je potrebno vnesti podatke o kartici.
  - Polje »*transaction Type*« se označi kot DEBIT oz. PREAUTHORIZE odvisno od tega, kateri tip transakcije trgovec podpira oz. želi izvesti.
  - V polju »*transaction Indicator*« je transakcijo potrebno označiti s SINGLE. S tem se ve da se ve, da gre za samostojno transakcijo oz. nakup (serija se ne vzpostavi), obenem se pa se tudi shranjuje podatke o kartici. To se označi v polju »*withRegister*« z vrednostjo TRUE.
  - Polje »*authenticationIndicator*« se pri tej vrsti izvedbe transakcije nastavi na 04, saj je vezano na polje »*transaction Type*«.
  - Polje *»challengeIndicator«* se polni z vrednostjo »04«, kar pomeni, da je ob vsaki začetni transakciji iz strani kartičnih shem zahtevana močna avtentikacija (SCA).
  - Ostalih zgoraj označenih polj ni potrebno napolniti.

- **Nadaljnje (Subsequent)** transakcije s shranjeno kartico:
  Gre za transakcijo sproženo s strani kupca (CIT), kjer uporabi že prej shranjeno kartico.
  - V polju »*transaction Type*« je potrebno transakcijo označiti kot DEBIT oz. PREAUTHORIZE odvisno od tega, kateri tip transakcije trgovec podpira oz. želi izvesti.
  - V polju »*transaction Indicator*« je transakcijo potrebno označiti s CARDONFILE, s tem se ve, da gre za nadaljnjo transakcijo, ki pa je vezana na prvo.
    - Zato je potrebno polje »*referenceTransactionId*« napolniti z UUID (enkratnim indikatorjem) Register oz. začetne transakcije, ker sistem tako ve, da je bila močna avtentikacija opravljena že pri prvi transakciji in je tu ni potrebno izvajat.

Primer kode za INITIAL (prve) transakcije, v JSON formatu:

```
{
  "merchantTransactionId": "D-2022-04-07-624eb695e63d3",
  "extraData": {
    "userField1": "00",
  },
```

```
  "threeDSecureData":{
     "authenticationIndicator":"04"
  }
  "merchantMetaData": "Transaction:Debit;Description:test",
  "amount": "1.99",
  "currency": "EUR",
  "successUrl": " https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentOK.php",
  "cancelUrl": " https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentCancel.php",
  "errorUrl": " https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentNOK.php",
  "callbackUrl": " https://mydomain.com/PHPPaymentGatewayJson/examples/Callback.php",
  "description": "One pair of shoes",
  "withRegister": true,
  "transactionIndicator": "SINGLE",
  "customer": {
     "firstName": "Janez",
     "lastName": "Novak",
     "billingAddress1": "Street 1",
     "billingCity": "City",
     "billingPostcode": "1000",
     "billingCountry": "SI",
     "email": "test@email.com",
     "ipAddress": "91.208.168.48"
  },
  "language": "sl"
}
```

Primer kode za SUBSEQUENT (naslednje) transakcije, v JSON formatu:

```
{
  "merchantTransactionId": "D-2022-04-07-624ebfa95f6d9",
  "extraData": {
     "userField1": "00"
  },
  "merchantMetaData": "Transaction:Debit;Description:test",
  "referenceUuid": "a4189ed3a0ea2546826a",
  "amount": "1.99",
  "currency": "EUR",
  "successUrl": https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentOK.php",
  "cancelUrl": " https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentCancel.php",
  "errorUrl": " https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentNOK.php",
  "callbackUrl": "https://mydomain.com/PHPPaymentGatewayJson/examples/Callback.php",
  "description": "One pair of shoes",
  "withRegister": false,
  "transactionIndicator": "CARDONFILE",
  "customer": {
     "firstName": "Janez",
     "lastName": "Novak",
     "billingAddress1": "Street 1",
     "billingCity": "City",
     "billingPostcode": "1000",
     "billingCountry": "SI",
     "email": "test@email.com",
     "ipAddress": "91.208.168.48"
  },
  "language": "sl"
}
```

#### 3.5.3.2. Označevanje MIT (Merchant Initiated Transactions) transakcij:

| JSON | INITIAL - AMT 0 | | INITIAL | | SUBSEQUENT |
|---|---|---|---|---|---|
| Type | **MIT** establish | | **MIT** establish | | **MIT** |
| Card number | enter | stored | enter | stored | stored |
| transaction Type | PREAUTHORIZE | | PREAUTH or DEBIT | | DEBIT |
| transactionIndicator | INITIAL | | INITIAL | | CARDONFILE-MIT |
| referenceTransactionId (referenceUuid) | \ | UUID(register) | \ | UUID(regi | UUID(initial) |
| withRegister | TRUE | TRUE | TRUE | TRUE | \ |
| authenticationIndicator | 02 | 02 | 02 | 02 | \ |
| recurringFrequency | 1 | 1 | 1 | 1 | \ |
| challengeIndicator | 04 | 04 | 04 | 04 | \ |
| authenticationIndicator, recurringFrequency in challengeIndicator se pošljejo v ThreeDSecureData | | | | | |

*Slika 6: Tabela z inidikatorji za MIT transakcije*

Znotraj MIT transakcij imamo 3 vrste možnih izvedb teh transakcij:

- **Začetna (Initial)**, kjer je znesek enak 0 – MIT establish:

  S to transakcijo trgovec pridobi soglasje za proženje nadaljnih transakcij brez kupčeve prisotnosti. Gre za transakcijo, sproženo s strani trgovca (MIT), kjer je potrebno vnesti podatke o kartici ali pa je kartica že shranjena z Register transakcijo od prej. V obeh primerih se polja polni enako.

  - V polju »*transaction Type*« se transakcijo označi z PREAUTHORIZE.
  - V polju »*transactionIndicator*« je potrebno transakcijo označiti z INITIAL, saj gre za prvo transakcijo v seriji.
  - Polje »withRegister« označimo s TRUE, kar pomeni, da se kartica hrani za kasnejša plačila.
  - Polje »*authenticationIndicator*« se pri tej vrsti izvedbe transakcije avtomatsko nastavi na 02, saj je vezano na polje »*transaction Type*«. Vrednost »02« pomeni, da se kartica ob vzpostavitvi MIT serije tudi shranjuje.
  - Polje »recurringFrequency« se privzeto polni z vrednostjo »1«, ker po strogih pravilih kartičnih shem ne gre za recurring transakcijo, saj je pri MIT transakcijah znesek vsak mesec lahko različen. Tu se lahko vnese željeno število plačil v seriji.
  - Če gre za transakcijo, ko je kartica že shranjena, je potrebno uporabiti še polje »*referenceTransactionId*« in ga napolniti z UUID (enkratnim indikatorjem) Register oz. začetne transakcije v seriji, saj sistem tako ve, da je bila močna avtentikacija opravljena že pri prvi transakciji in je tu ni potrebno izvajati.
  - Polje »*challengeIndicator*« se polni z vrednostjo »04«, kar pomeni, da je ob vsaki začetni transakciji iz strani kartičnih shem zahtevana močna avtentikacija (SCA).

- **Začetna (Initial)**, kjer je znesek različen od 0 – MIT establish:

  Gre za transakcijo, sproženo s strani trgovca (MIT), kjer je ali potrebno vnesti podatke o kartici ali pa je kartica že shranjena z Register transakcijo od prej.

V obeh primerih se naslednja polja polni enako.

- S to transakcijo se vzpostavi serija MIT transakcij in se hkrati dejansko opravi tudi nakup dobrin, zato se polje »*transaction Type*« označi kot DEBIT oz. PREAUTHORIZE, odvisno od tega, kateri tip transakcije trgovec podpira oz. želi izvesti.

Za ostala polja veljajo enaka pravila, kot pri začetni transakciji, kjer je znesek 0.

- **Nadaljnje (Subsequent) MIT** transakcije:

Gre za transakcijo, sproženo s strani trgovca (MIT), kjer uporabi že prej shranjeno kartico.

- Polje »*transaction Type*« se označi z DEBIT, ker gre za enofazno transakcijo.
- V polju »*transaction Indicator*« je transakcijo potrebno označiti s CARDONFILE-MIT, s tem se ve, da gre za nadaljnjo transakcijo, ki pa je vezana na prvo.
  - Zato je potrebno polje »*referenceTransactionId*« napolniti z UUID (enkratnim indikatorjem) Initial oz. začetne transakcije, ker sistem tako ve, da je bila močna avtentikacija opravljena že pri prvi transakciji in je tu ni potrebno izvajat.
- Ostalih zgoraj označenih polj ni potrebno napolniti.

Primer kode za INITIAL (prve) transakcije v JSON formatu:

```
{
  "merchantTransactionId": "D-2022-04-07-624eb9146f6bf",
  "extraData": {
    "userField1": "00",
   }
    "threeDSecureData":{
     "authenticationIndicator": "02",
     "recurringFrequency": "1"
  },
  "merchantMetaData": "Transaction:Debit;Description:test",
  "amount": "1.99",
  "currency": "EUR",
  "successUrl": https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentOK.php",
  "cancelUrl": " https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentCancel.php",
  "errorUrl": " https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentNOK.php",
  "callbackUrl": "https://mydomain.com/PHPPaymentGatewayJson/examples/Callback.php",
  "description": "One pair of shoes",
  "withRegister": true,
  "transactionIndicator": "INITIAL",
  "customer": {
    "firstName": "Janez",
    "lastName": "Novak",
    "billingAddress1": "Street 1",
    "billingCity": "City",
    "billingPostcode": "1000",
    "billingCountry": "SI",
    "email": "test@email.com",
    "ipAddress": "91.208.168.48"
  },
  "language": "sl"
}
```

Primer kode za SUBSEQUENT (naslednje) transakcije, v JSON formatu:

```
{
  "merchantTransactionId": "D-2022-04-07-624ebdc06bed2",
```

```
  "extraData": {
    "userField1": "00"
  },
  "merchantMetaData": "Transaction:Debit;Description:test",
  "referenceUuid": "2a06c2a09b69c8eced14",
  "amount": "1.99",
  "currency": "EUR",
  "successUrl": https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentOK.php",
  "cancelUrl": " https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentCancel.php",
  "errorUrl": " https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentNOK.php",
  "callbackUrl": "https://mydomain.com/PHPPaymentGatewayJson/examples/Callback.php",
  "description": "One pair of shoes",
  "withRegister": false,
  "transactionIndicator": "CARDONFILE-MERCHANT-INITIATED",
  "customer": {
    "firstName": "Janez",
    "lastName": "Novak",
    "billingAddress1": "Street 1",
    "billingCity": "City",
    "billingPostcode": "1000",
    "billingCountry": "SI",
    "email": "test@email.com",
    "ipAddress": "91.208.168.48"
  },
  "language": "sl"
}
```


### 3.5.3.3. Označevanje Recurring transakcij:

| JSON | INITIAL - AMT 0 | | INITIAL | | SUBSEQUENT |
|---|---|---|---|---|---|
| Type | **RECURRING** establish | | **RECURRING** establish | | **RECURRING** |
| Card number | enter | stored | enter | stored | stored |
| transaction Type | PREAUTHORIZE | | PREAUTHORIZE or DEBIT | | DEBIT |
| transactionIndicator | INITIAL | | INITIAL | | RECURRING |
| referenceTransactionId(referenceUuid) | \ | UUID(register) | \ | UUID(regi | UUID(initial) |
| withRegister | TRUE | TRUE | TRUE | TRUE | \ |
| authenticationIndicator | 02 | 02 | 02 | 02 | \ |
| recurringFrequency | <>1 | <>1 | <>1 | <>1 | \ |
| challengeIndicator | 04 | 04 | 04 | 04 | \ |
| authenticationIndicator, recurringFrequency in challengeIndicator se pošljejo v ThreeDSecureData | | | | | |

*Slika 7: Tabela z indikatorji za Recurring transakcije*

Znotraj Recurring transakcij imamo zopet 3 vrste možnih izvedb teh transakcij:

- **Začetna (Initial)**, kjer je znesek enak 0 – Recurring establish:

  S to transakcijo se vzpostavi serija MIT transakcij. Gre za transakcijo, sproženo s strani trgovca (MIT), kjer je potrebno vnesti podatke o kartici ali pa je kartica že shranjena z Register transakcijo od prej.

  V obeh primerih se naslednja polja polni enako.

  - V polju »*transaction Type*« se transakcijo označi z PREAUTHORIZE.
  - V polju »*transactionIndicator*« je potrebno transakcijo označit z INITIAL, ker gre za prvo transakcijo v seriji.
  - Polje »withRegister« označimo z TRUE, kar pomeni, da se kartica hrani.
  - Polje »*authenticationIndicator*« se pri tej vrsti izvedbe transakcije avtomatsko nastavi na 02, saj je vezano na polje »*transaction Type*«. Vrednost »02« pomeni, da se kartica, ob vzpostavitvi MIT serije, tudi shranjuje.

- Polje »recurringFrequency« se privzeto polni z vrednostjo različno od 1 oziroma s številom transakcij, ki bodo v tej seriji izvedene. Tu se lahko vnese željeno število plačil v seriji.
- Če gre za transakcijo, ko je kartica že shranjena, je potrebno uporabiti še polje »*referenceTransactionId*« in ga napolniti z UUID (enkratnim indikatorjem) Register oz. začetne transakcije v seriji, ker sistem tako ve, da je bila močna avtentikacija opravljena že pri prvi transakciji in je tu ni potrebno izvajat.
- Polje *»challengeIndicator«* se polni z vrednostjo »04«, kar pomeni, da je ob vsaki začetni transakciji iz strani kartičnih shem zahtevana močna avtentikacija (SCA).

- **Začetna (Initial)**, kjer je znesek različen od 0 – Recurring establish:
  Gre za transakcijo, sproženo s strani trgovca (MIT), kjer je ali potrebno vnesti podatke o kartici ali pa je kartica že shranjena z Register transakcijo od prej.
  V obeh primerih se naslednja polja polni enako.
  - S to transakcijo se vzpostavi serija MIT transakcij in se hkrati dejansko opravi tudi nakup dobrin, zato se polje »*transaction Type*« označi kot DEBIT oz. PREAUTHORIZE, odvisno od tega, kateri tip transakcije trgovec podpira oz. želi izvesti.

  Za ostala polja veljajo enaka pravila kot pri začetni transakciji, kjer je znesek 0.
- **Nadaljnje (Subsequent) Recurring** transakcije:
  Gre za transakcijo, sproženo s strani trgovca (MIT), kjer uporabi že prej shranjeno kartico.
  - Polje »*transaction Type*« se označi z DEBIT, ker gre za enofazno transakcijo.
  - V polju »*transaction Indicator*« je transakcijo potrebno označiti z RECURRING, s tem se ve, da gre za nadaljnjo transakcijo, ki pa je vezana na prvo.
    - Zato je potrebno polje »*referenceTransactionId*« napolniti z UUID (enkratnim indikatorjem) Initial oz. začetne transakcije, ker sistem tako ve, da je bila močna avtentikacija opravljena že pri prvi transakciji in je tu ni potrebno izvajati.
  - Ostalih zgoraj označenih polj, ni potrebno napolniti

Primer kode za INITIAL (prve) transakcije, v JSON formatu.

```
{
  "merchantTransactionId": "D-2022-04-07-624eb9951e2ac",
  "extraData": {
    "userField1": "05",
   "threeDSecureData":{
    "authenticationIndicator": "02",
    "recurringFrequency": "2"
  },
  "merchantMetaData": "Transaction:Debit;Description:test",
  "amount": "1.99",
  "currency": "EUR",
  "successUrl": https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentOK.php",
  "cancelUrl": " https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentCancel.php",
  "errorUrl": " https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentNOK.php",
  "callbackUrl": "https://mydomain.com/PHPPaymentGatewayJson/examples/Callback.php",
  "description": "One pair of shoes",
  "withRegister": true,
  "transactionIndicator": "INITIAL",
  "customer": {
    "firstName": "Janez",
```

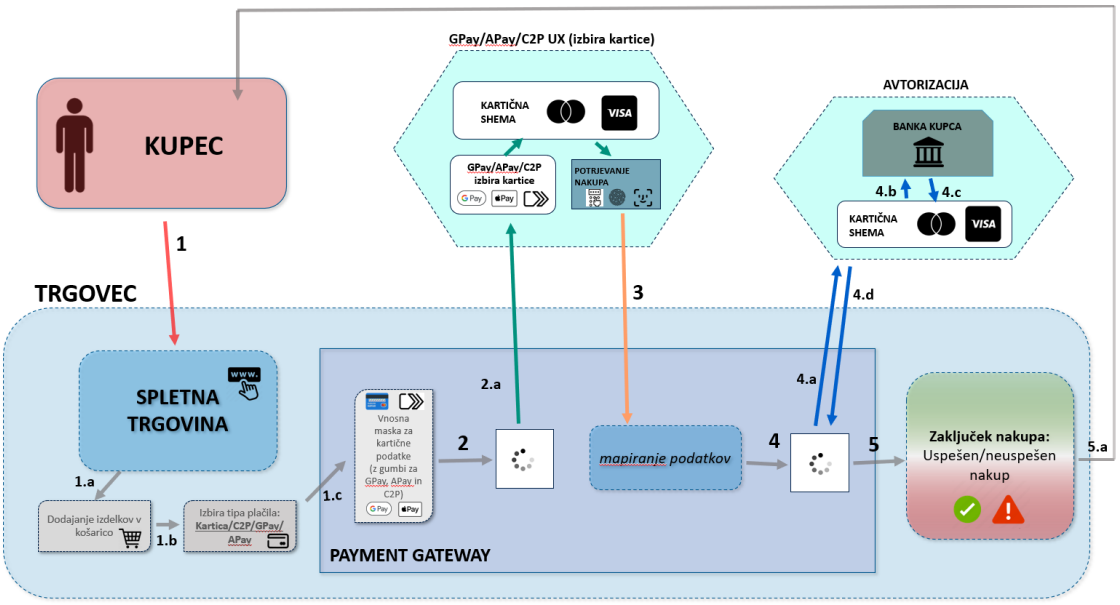
```
    "lastName": "Novak",
    "billingAddress1": "Street 1",
    "billingCity": "City",
    "billingPostcode": "1000",
    "billingCountry": "SI",
    "email": "test@email.com",
    "ipAddress": "91.208.168.48"
  },
  "language": "sl"
}
```

Primer kode za SUBSEQUENT (naslednje) transakcije, v JSON formatu:

```
{
  "merchantTransactionId": "D-2022-04-07-624ebf19af16d",
  "extraData": {
    "userField1": "00"
  },
  "merchantMetaData": "Transaction:Debit;Description:test",
  "referenceUuid": "4a58b83ad5794fa66041",
  "amount": "1.99",
  "currency": "EUR",
  "successUrl": https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentOK.php",
  "cancelUrl": " https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentCancel.php",
  "errorUrl": " https://mydomain.com/PHPPaymentGatewayJson/examples/PaymentNOK.php",
  "callbackUrl": "https://mydomain.com/PHPPaymentGatewayJson/examples/Callback.php",
  "description": "Par čevljev",
  "withRegister": false,
  "transactionIndicator": "RECURRING",
  "customer": {
    "firstName": "Janez",
    "lastName": "Novak",
    "billingAddress1": "Street 1",
    "billingCity": "City",
    "billingPostcode": "1000",
    "billingCountry": "SI",
    "email": "test@email.com",
    "ipAddress": "91.208.168.48"
  },
  "language": "sl"
```

## 3.6. Uporaba xPays plačilnih metod (Google Pay, Apple Pay, Click 2 Pay)

Bankartov Payment Gateway omogoča uporabo sodobnih plačilnih metod, kot so **Google Pay**, **Apple Pay** in **Click to Pay**.




*Slika 8: Prikaz poteka transakcije v primeru xPay plačilne metode*

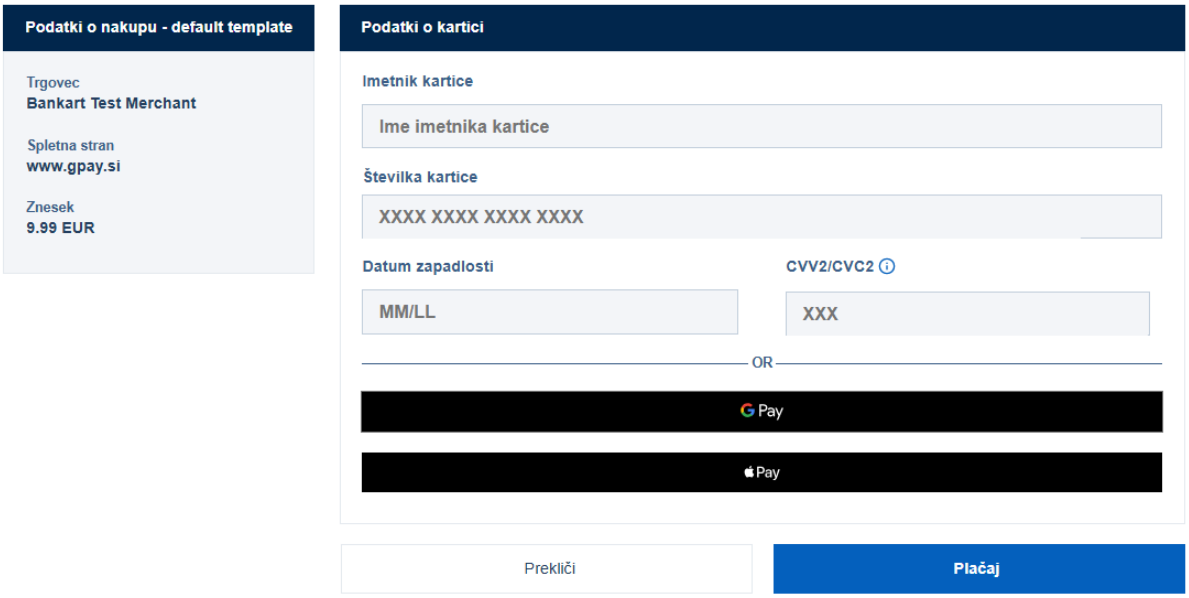
Za omogočanje teh plačilnih metod mora trgovec podpreti klasičen tip transakcije (*debit* ali *predavtorizacijo*), saj se le tako lahko prikaže HPP, kjer stranka izbere željeno plačilno metodo s klikom na ustrezen gumb. Na strani trgovca pri tem ni dodatnega dela, kot je prevzem dodatnih podatkov za integracijo (kar je sicer potrebno le pri drugem tipu integracije). Pri zadnjem izmed treh tipov integracije pa je treba dodatno vključiti še kodo za iFrame. V tem primeru bo za trgovca organiziran dodatni sestanek, z Bankartom ekipo in banko.

### 3.6.1. Tipi integracije pri Google Pay in Apple Pay

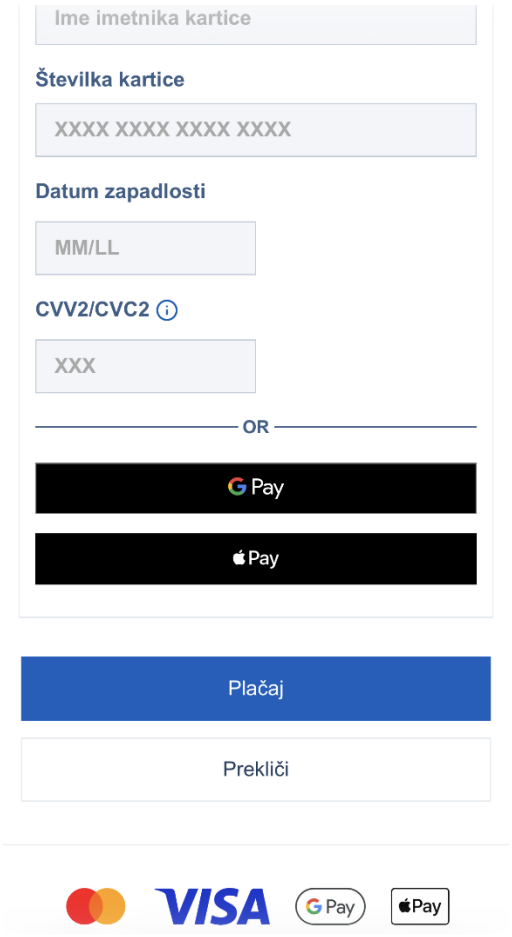
**Uporaba preko Bankartove vnosne maske (HPP)**

Če trgovec uporablja Bankartovo vnosno masko (**HPP – Hosted Payment Page**), celoten postopek registracije, pridobivanja certifikatov ter implementacije plačilnih metod uredita banka in Bankart. Trgovcu ostane le, da funkcionalnost preveri s testiranjem in jo nato lahko nemoteno začne uporabljati v produkciji.

Prikaz klasične HPP strani skupaj z Google Pay in Apple Pay gumboma:




*Slika 9: Prikaz gumbov za plačilo na HPP v brskalniku, na računalniku*



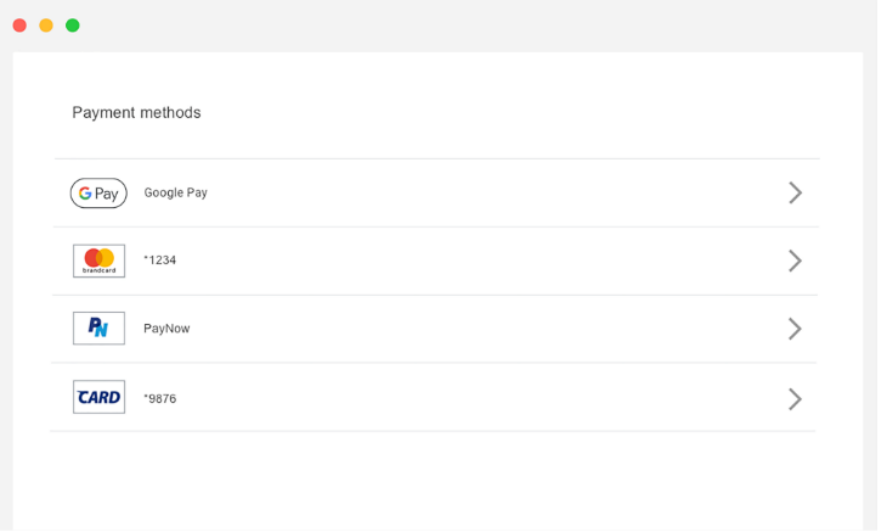

*Slika 9: Prikaz gumbov za plačilo na HPP v brskalniku, na iPhonu*

Trgovec mora na zaslonu za izbiro plačilne metode – tam, kjer omogoča kartično plačilo – dodati tudi možnost plačila z Google Pay in/ali Apple Pay. Prikaz naj bo npr. v obliki: »Kartično plačilo / Google Pay

/ Apple Pay«, saj Google in Apple zahtevata, da je možnost plačila s tema metodama vedno jasno prikazana.

Če trgovec omogoča le eno od teh dveh metod, se bo prikazal samo gumb za izbrano plačilno metodo

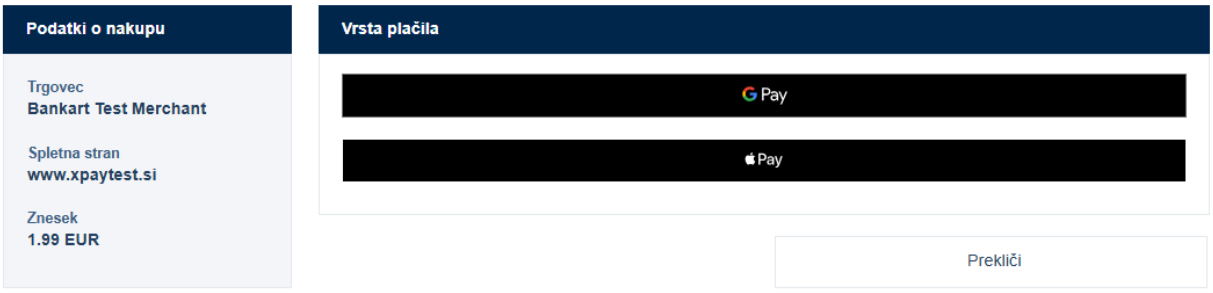
**Ločen terminal za Google Pay / Apple Pay (preko HPP)**
V primeru, da želi trgovec vzpostaviti **ločeno plačilno pot** za Google Pay ali Apple Pay preko prirejenega HPP, mora vsako izmed teh plačilnih metod na ekranu za izbiro plačilnih metod, prikazati kot **poseben način plačila**, skladno s smernicami Google in Apple.



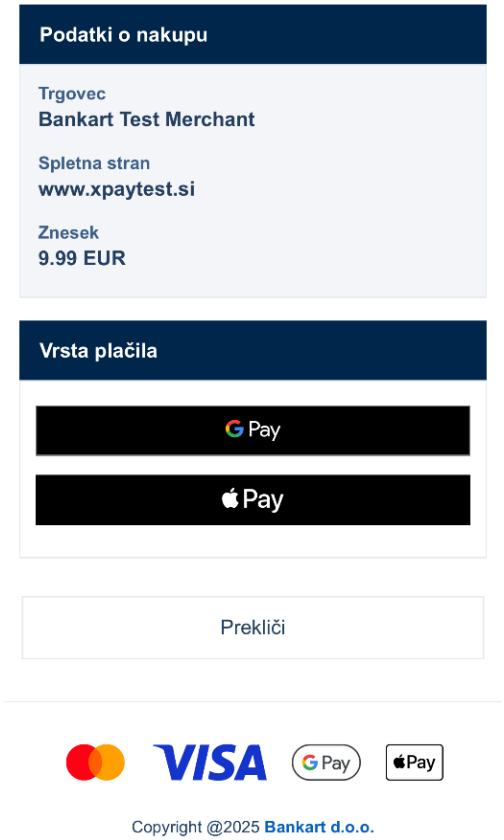

*Slika 11: Primer kako izgleda prikaz Google Pay kot ločen način plačila (posnetek zaslona direktno iz Googlovih smernic)*

Trgovec mora ob prevzemu podatkov za prijavo prevzeti še podatke za dodatni terminal, ki bo namenjen izključno plačevanju z eno od teh dveh plačilnih metod (Google Pay ali Apple Pay).
Na prilagojenem HPP-ju se bo trgovcu prikazal le gumb za tisto metodo, ki je na tem terminalu dovoljena – torej bodisi gumb za Google Pay ali gumb za Apple Pay, nikoli oba hkrati.
Na spodnjih posnetkih zaslona sta sicer prikazana oba gumba hkrati, vendar je to zgolj zaradi prikaza prilagojene HPP strani v tem dokumentu in ne odraža dejanskega delovanja v produkciji.

Prikaz prirejene HPP strani z Google Pay in Apple Pay gumboma:




*Slika 12: Prikaz gumbov za plačilo na prirejenem HPP v brskalniku, na računalniku*




*Slika 13: Prikaz gumbov na prirejeni HPP za plačilo v brskalniku, na iPhonu*

**Integracija pri trgovcih, ki za kartična plačila uporabljajo *payment.js* integracijo**

Če želi trgovec integrirati Google Pay in Apple Pay neposredno v svojo spletno trgovino, bo to možno le za Google Pay, saj Apple Pay takšne integracije ne podpira.

Google Pay gumb je mogoče prikazati neposredno na spletni strani trgovca z uporabo iFrame-a, ki omogoča sprožitev plačilne metode. V primeru Apple Pay pa takšna rešitev zaradi varnostnih omejitev na strani Appla ni izvedljiva.

Če trgovec uporablja *payment.js* integracijo za kartična plačila, bo moral za ponujanje Apple Pay plačilne metod uporabiti prejšnji tip integracije – torej ločen terminal za Google Pay/Apple Pay, ki deluje preko HPP-ja.

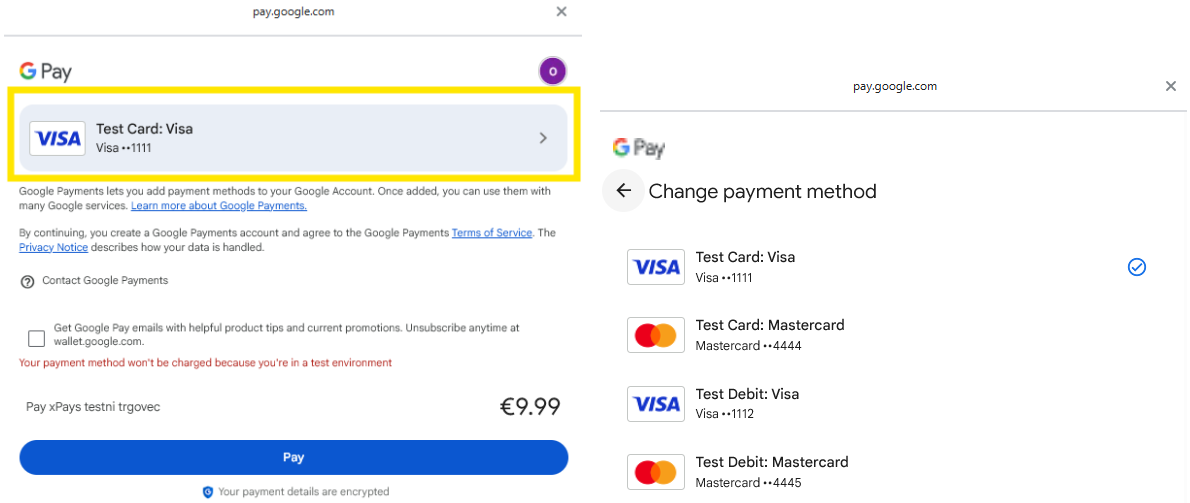
Če trgovcu ni sprejemljivo, da se ena plačilna metoda prikazuje na njegovi spletni strani, druga pa na Bankartovem HPP-ju, je priporočljiv nadomestni pristop – torej uporaba zgoraj omenjene integracije ločenega terminala za Google Pay in/ali Apple Pay.

Bankartova tehnična ekipa je trgovcem na voljo za podporo pri vseh vrstah integracij, ter pri testiranju.

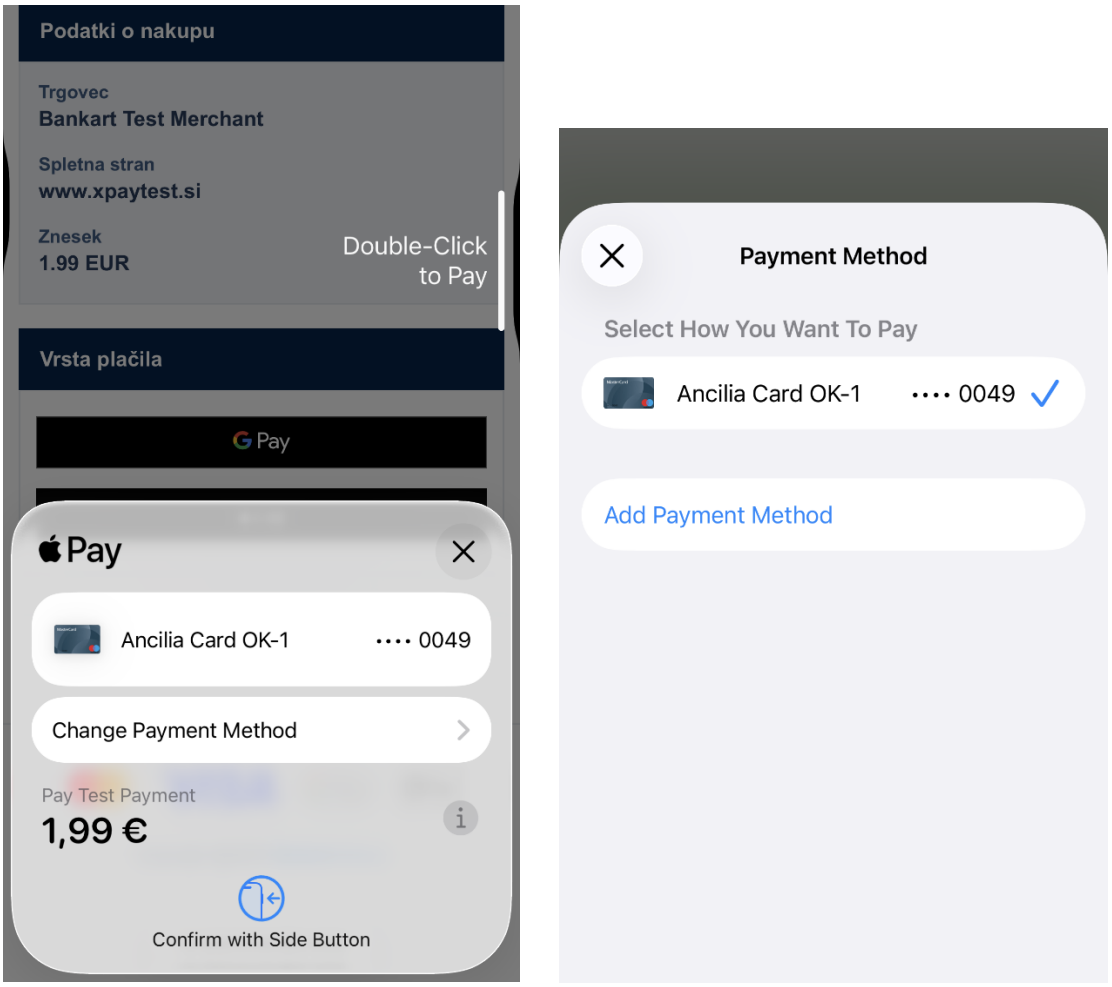
### 3.6.2. Uporabniška izkušnja stranke pri Google Pay in Apple Pay

Gumba za plačilo z Google Pay ali Apple Pay se prikažeta v vseh sodobnih brskalnikih, če trgovec podpira izbrano plačilno metodo. Edina izjema je gumb za Apple Pay, ki ni prikazan na napravah z operacijskim sistemom Android, medtem ko je gumb za Google Pay na voljo povsod.

S klikom na Google ali Apple Pay gumb se sproži uporabniška izkušnja v okolju Googla ali Appla – odpre se pojavno okno (pop-up), kjer stranka izbere želeno kartico iz svoje Google ali Apple denarnice (Wallet), povezane z njenim uporabniškim računom.

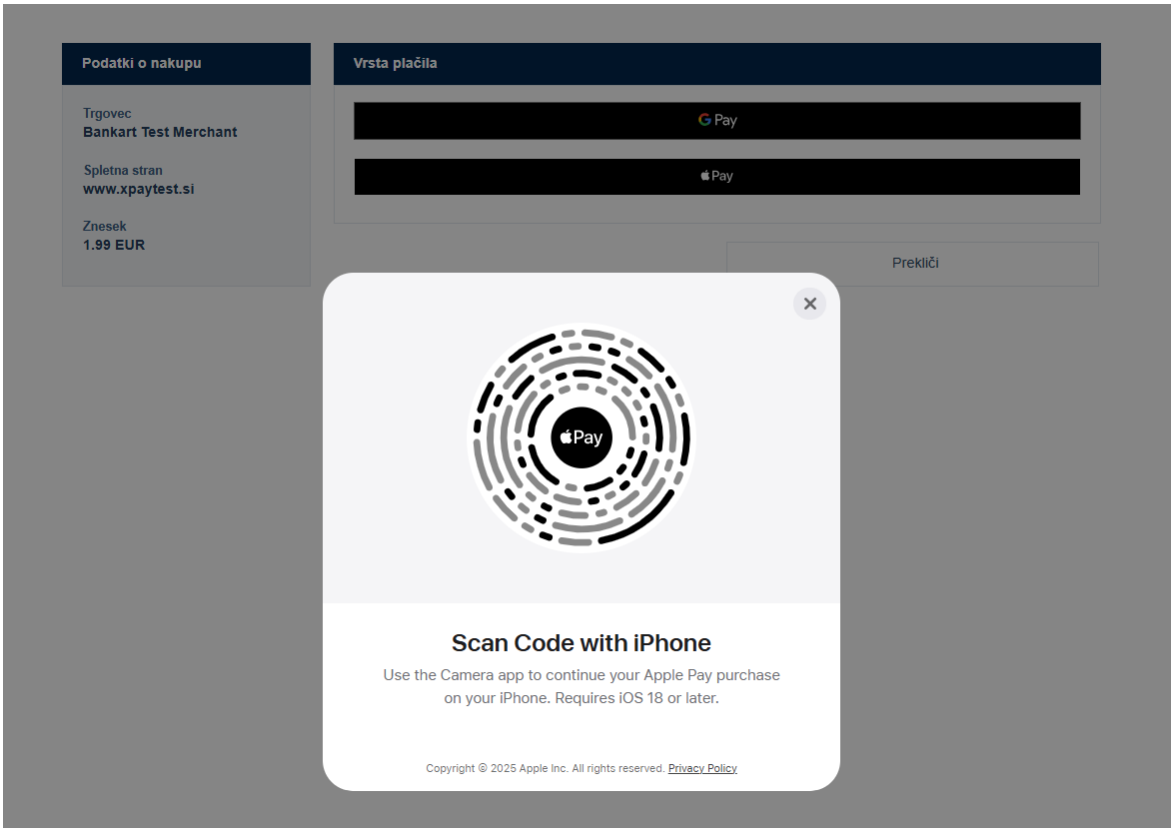



*Slika 14: Pojavno okno, v primeru plačila z Google Pay (levo) in možnost menjave kartice (desno)*




*Slika 15: Pojavno okno na iPhonu, v primeru plačila z Apple Pay (levo) in možnost menjave kartice (desno)*

V primeru, ko želi kupec plačati z Apple Pay na napravi, ki ni Applova (npr. Windows računalnik), ali v brskalniku, ki ni Applov (npr. Chrome ali Firefox), se mu prikaže Apple-ova različica QR kode. Kupec mora to kodo skenirati z iPhonom, nato pa transakcijo nadaljuje in zaključi na svojem telefonu.




*Slika 16:Applova QR koda, ki omogoča plačilo z Apple Pay v brskalnikih, ki niso Safari*

Po zaključku uporabniške izkušnje se samodejno izvede preusmeritev nazaj na HPP, kjer se sproži nadaljevanje plačilnega procesa. Podatki o plačilu se samodejno posredujejo na Payment Gateway, ki po izvedeni avtorizaciji trgovcu vrne rezultat transakcije.

Trgovec mora spremeniti tudi ekran, kjer prikaže rezultat transakcije, kjer mora v skladu z oblikovnimi in funkcionalnimi smernica Google in Apple dodat ekran, kjer piše, da je bila transakcija uspešno opravljena z eno izmed plačilnih metod. Podrobnosti so na voljo na povezavah. Informacijo, s katero plačilno metodo je bila izvedena transakcija, trgovec pridobi na callbackURLjih v polju *cardHolder.*

```
"returnData": {
    "_TYPE": "cardData",
    "type": "mastercard",
    "cardHolder": "ApplePay",
```

```
"returnData": {
    "_TYPE": "cardData",
    "type": "visa",
    "cardHolder": "GooglePay",
```

OPOMBA: Ti načini plačila niso na voljo trgovcem, ki uporabljajo integratorje oziroma povezavo server-to-server. So pa lahko dostopni za zgoraj opisane transakcije s Shranjevanje kartice, če so vključeni ustrezni indikatorji, kot je navedeno zgoraj.

## 3.7. Podpora spletnemu plačevanju na obroke

V okviru podpore spletnemu plačevanju je možno uvesti tudi način izvedbe plačila na obroke. Storitev je na voljo trgovcem, ki z banko, ki to storitev opredeljuje v svoji ponudbi, sklenejo pogodbo.
Pogodba vključuje podporo na strani Bankartovih sistemov (Pamyent Gateway, avtorizacijski sistem, sistem za pripravo podatkov za knjiženje), hkrati pa je potrebno na trgovčevi strani uvesti podporo, s katero bo ta storitev kot celota ponujena kupcu. V nadaljevanju podajamo opis podpore ter način uvedbe podpore na trgovčevi strani.
Osnova za procesiranje obrokov je, da trgovec v API klicu pošlje željeno število obrokov v katerem bo izvedeno plačilo s strani kupca. Ob nakupu bo avtorizacija na kartico narejena na osnovi celotnega zneska nakupa.

**Postopek uvedbe podpore spletnega plačevanja na obroke je naslednji:**

- Trgovec z banko sklene dogovor o uvedbi podpore plačevanja spletnih nakupov na obroke
- Banka Bankartu sporoči šifro trgovca, s katerim je sklenjena pogodba za plačevanje na obroke. Na osnovi tega podatka se na avtorizacijskem sistemu odpre možnost plačevanja na obroke. V sklopu kontrole na avtorizacijskem sistemu se poleg dovoljenja za izvajanje plačil na obroke naredijo še naslednje kontrole:
    - Število obrokov je določeno z bančno pogodbo. V kolikor je v zahtevku v polju za število obrokov poslana vrednost 0 ali 1, se takšen nakup šteje kot nakup brez obrokov
    - Ob prejetju nedovoljenih znakov (dovoljeni so samo numerični znaki) je transakcija zavrnjena
    - V kolikor za trgovca ni označena možnost dovoljevanja plačila na obroke, se vrednost v prejetem polju z obroki s strani trgovca ignorira, transakcija se izvede brez obrokov
- Za posredovanje obrokov do avtorizacijskega sistema je uporabljeno obstoječe polje v Payment Gateway-u imenovano »*userField1« (vidno v zgornjih primerih JSON klica za shranjevanje kartice)*. Do sedaj to polje ni bilo v uporabi (vpisana vrednost ni vplivala na procesiranje transakcij). V koliko ima trgovec podpisano pogodbo za podporo obrokom in uvedeno rešitev, vrednost v polju »*userField1«* vpliva na procesiranje.
- Vpis števila obrokov je izveden s strani uporabnika. V izogib težavam z vnosom neustreznih podatkov s strani kupca priporočamo uporabo padajočih menijev, radijskih gumbov ipd. in s tem omejitev možnih vpisanih vrednosti samo na število določeno v bančni pogodbi. Trgovec lahko uvede dodatne kontrole obrokov na svoji strani in po vpisu števila želenih obrokov s strani stranke glede na kupljene artikle oziroma znesek takšno transakcijo dovoli (jo posreduje v nadaljnjo obdelavo) ali pa izbiro kupca ne dovoli (npr. prepreči nakup v znesku 10€ na 5 obrokov, ki bi povzročil z vidika bančnih nadomestil prevelike stroške).
- Trgovec mora v postopku prikaza možnih plačilnih metod kupcu dati na izbiro metodo, ki pokriva plačevanje na obroke s karticami, za katere ima trgovec pogodbo. V kolikor kupec izbere način plačevanja na obroke, trgovec kupcu omogoči izbiro ali (manj priporočljivo) vpis želenega števila obrokov. Izbrano/vpisano število obrokov s strani kupca, trgovec pošlje v zahtevku v polju *userField1*.
- Po posredovanju zahtevka bo le ta obravnavan v skladu z rednimi produkcijskimi postopki z dodatkom kontrole polja z obroki. Avtorizacija je izvedena na celoten znesek nakupa.
- Trgovec naredi zajem (CAPTURE) v skladu z dinamiko odpošiljanja blaga ne glede na to, ali je bilo izbrano plačilo na obroke ali brez njih
- V kolikor bo trgovec zajeme (CAPTURE) nakupa delal v več delih, bo za vsak posamezen del veljalo, da bo obračunan v skladu z izbranim številom obrokov.

- V kolikor ima trgovec uvedeno podporo obrokom (na osnovi pogodbe z banko) in posreduje v zahtevku tudi število obrokov, lahko v primeru pozitivnega odgovora smatra, da bo kupec s strani banke obremenjen v okviru zneskov posameznih obrokov. Trgovec naj v takšnem primeru v potrdilu stranki zapiše, da je bil nakup izveden na obroke v številu, poslanem v zahtevku.
- Kritje bančnih stroškov obročnega poslovanja se izvede v skladu z dogovorom z banko.
- Trgovec implementacijo menija za izbiro števila obrokov naredi sam, saj podpora za to na HPP vnosni maski ni.

# 4. SMERNICE ZA TESTIRANJE

## 4.1. Osnovne informacije

Pred prehodom v produkcijsko okolje za uporabo kartičnega plačevanja je za uporabo Bankartovega Payment Gatewaya potrebno uspešno opraviti test, ki potrdi pravilno delovanje trgovčevega sistema. Pred začetkom testiranja vam bo Bankart ekipa za podporo uporabnikom (customer.support@bankart.si) posredovala naslednje podatke, s katerimi se boste povezali na Bankart Payment Gateway in tako lahko začeli testiranje.

| Naziv | Opis |
|---|---|
| **API uporabniško ime (Api username)** | Dostop do API vmesnika – razvijalec to uporabi v kodi. |
| **API geslo (Api password)** | Dostop do API vmesnika. |
| **API Ključ (API Key)** | Ključ za plačilno metodo. |
| **Deljena skrivnost (Shared Secret)** | Deljena skrivnost za podpisovanje in preverjanje verodostojnosti vsebine API klicev. |
| **Public Integration Key (PKI)** | Ključ za integracijo payment.jst elementa v svoji trgovini. |
| *Web dostop:* | |
| **URL** | Spletni naslov za dostop do portala in pregled transakcij (se navezuje na naslednje poglavje) |
| **Uporabniško ime (Username)** | Dodelejeno uporabniško ime za dosop do portala |
| **Geslo (Password)** | Dodeljeno geslo za prvo prijavo (potrebno spremeniti po prvi prijavi!) |

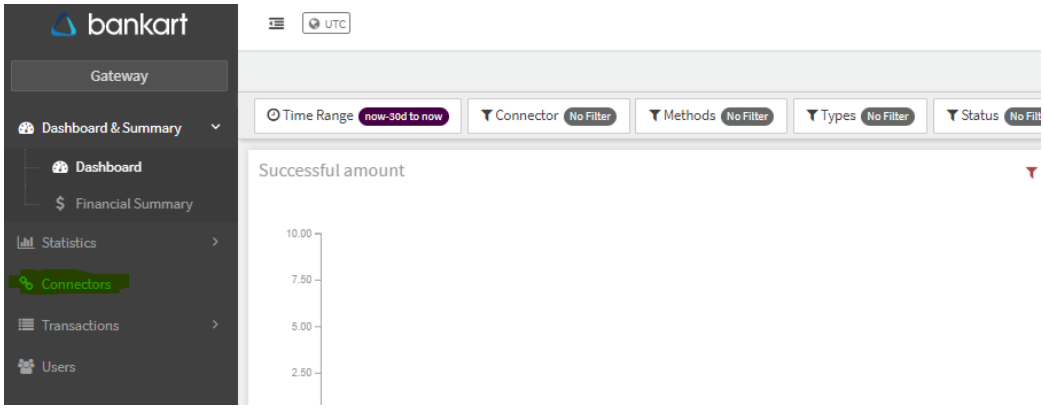
## 4.2. Prevzem podatkov za povezavo s Payment Gateway-om

Podatke za vstop do Payment Gateway portala pošljemo po dveh kanalih:

- **e-mail naslov**, ki je podan ob prijavi trgovca. Prek tega kanala bodo poslana navodila za integracijo ter geslo za dostop do portala, ki ga je potrebno zamenjati takoj ob prijavi.
- **SMS sporočilo** na številko, ki je podana ob prijavi trgovca. Prek tega kanala bo poslano uporabniško ime za dostop do portala.

Ko prejmete podatke za vpis na portal je potrebno tam pobrati še podatke za povezavo/integracijo terminalov, kar je po korakih prikazano spodaj:

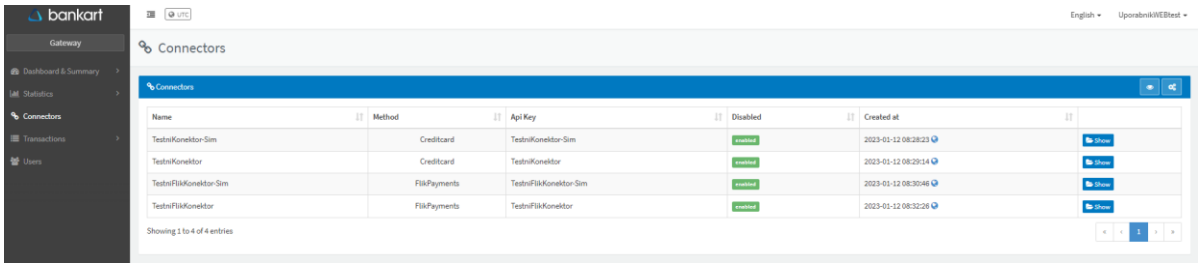
1. **korak:** Po vpisu na Payment Gateway portal na levi strani izberete zavihek »Connectors« (*označeno z zeleno*).




*Slika 17: Izbor zavihka "Connectors" v Payment Gatewayu*

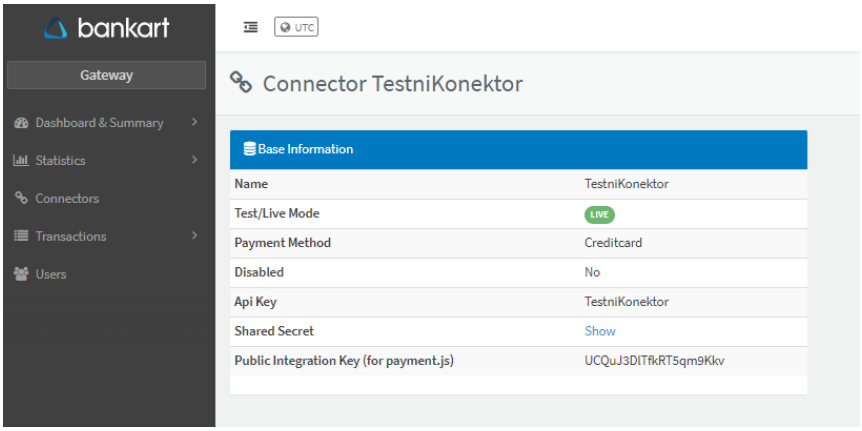
2. **korak:** Prikaže se seznam vseh terminalov oziroma konektorjev. Pri vsakem od konektorjev s klikom na gumb »Show« na desni strani pridete do podatkov, ki so pomembni za integracijo oz. povezavo na Payment Gateway (v naslednjem koraku). Konektorji, ki imajo na koncu »Sim«,

so simulacijski konektorji in se uporabljajo za testiranje integracije preden gre trgovec v produkcijo.
Več o samem testiranju s simulacijskimi konektorji je napisano nižje v dokumentu.



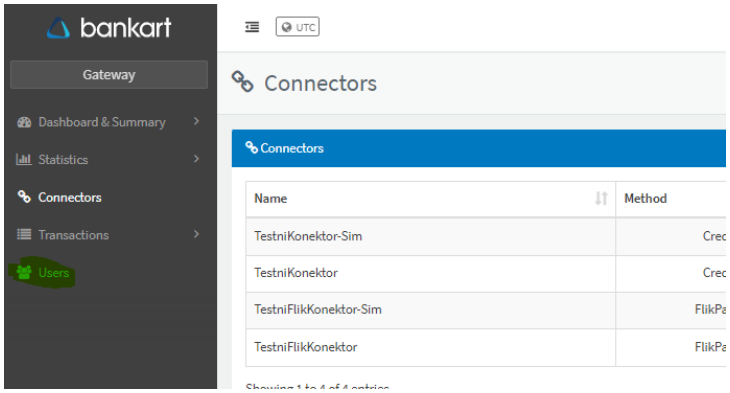

*Slika 18: Prikaz konektorjev v Payment Gatewayu*

3. **korak:** S klikom na gumb »Show« si lahko ogledate podrobnejše informacije o konektorju. Podatki, ki jih trgovec potrebuje za povezavo na Gateway, so:
    a. API Key
    b. Shared Secret (s klikom na »show« se prikaže podatek)
    c. Public Integration Key (Potrebujete v primeru *payment.js* integracije pri kartičnih plačilih – če trgovec ne želi preusmeritve na Bankartovo vnosno masko, ampak želi implementacijo na svoji strani, da izgleda, kot da stranka nikoli ne zapusti spletne strani/trgovine.)



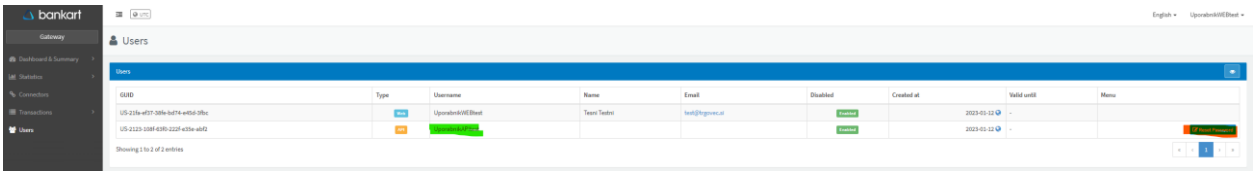

*Slika 19: Prikaz podatkov o konektorju, ki jih je potrebno prekopirat v trgovčeve sisteme*

4. **korak:** Po tem, ko so podatki za vsakega izmed željenih konektorjev skopirani v trgovčev sistem, na levi strani v meniju izberete zavihek »Users« (*označeno z zeleno*).



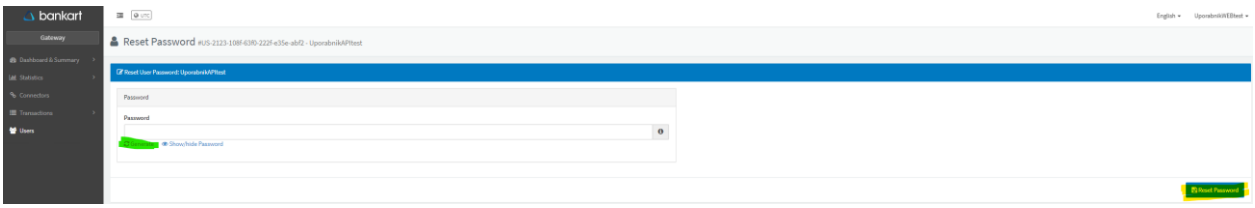

*Slika 20: Prikaz izbire zavihka "Users" v Payment Gatewayu*

5. **korak:** Prikaže se seznam vseh trgovčevih uporabnikov. Uporabnikov tipa WEB je lahko več, saj so to uporabniki, za katere trgovec omogoči dostop do Payment Gateway portala in pregleda transakcij (predlagamo do največ 3, ki jih kreira Bankart). Uporabnik tipa API je navadno samo eden, saj je preko le-tega omogočena povezava do Payment Gateway-a.
   Podatki za glavnega WEB uporabnika so bili trgovcu že poslani. Prav tako je trgovec že prevzel podatke o konektorju. Za konec pa je potrebno prevzeti še podatke o API uporabniku. Za ta del mora trgovec k sebi v sistem prekopirati še API uporabniško ime (*označeno z zeleno*), in pri tem istem API uporabniku je potrebno v Payment Gatewayu na desni strani strani treba klikniti na gumb »Reset Password« (*označeno z rdečo*).




*Slika 21: Prikaz uporabnikov v Payment Gatewayu s poudarkom na API uporabnikov*

6. **korak:** Prikaže se meni za generiranje API gesla. S klikom na opcijo »Generate« (*označeno z zeleno*) pod poljem Password se generira geslo. To geslo **skopirate v trgovčeve sisteme**, nato pa na desni strani spodaj kliknete gumb »Reset Password« (*označeno z rumeno*). S tem se geslo shrani v Payment Gateway sistem.




*Slika 22: Prikaz spremembe gesla pri API uporabniku*

**POMEMBNO:** Generiranje API gesla naj se naredi **samo enkrat oz. ob prvi prijavi**. Preden se dokonča postopek za generiranje gesla ga je potrebno shraniti in kopirati v trgovčeve zaledne sisteme, da se trgovec lahko poveže s Payment Gateway-om. Če se API geslo spreminja po tem, ko je trgovec že dlje časa v produkciji, in uporablja API geslo, generirano ob prvotni prijavi, se povezava po naknadni spremembi gesla med trgovcem in Payment Gateway-om **PREKINE**, prav tako pa se transakcije **NE morejo** pravilno prožiti in izvajati.

## 4.3. Testne kartice

**Za testiranje je potrebno uporabiti testne kartice, ki so navedene na desni.** V primeru nepričakovanih odgovorov se obrnite na kontaktni e-mail naslov customer.support@bankart.si.

Prilagamo še podatke o karticah, ki se lahko uporabijo za testiranje pravilnega delovanja transakcij za posamezno kartično shemo.
Najprej **številke testnih kartic za VISO in Mastercard** (slika na desni). Glede na željen rezultat (ki je zapisan o številki kartice), izberite eno izmed številk kartic. V polje kjer se preverja CVC koda, lahko vpišete katerokoli tri mestno kombinacijo, enako pa velja za datum veljavnosti, kjer se lahko vpiše katerikoli (veljaven) datum.

Test data

Credit cards

| Brand | Number | Result |
|---|---|---|
| Visa | 4111 1111 1111 1111 | Success |
| Visa | 4242 4242 4242 4242 | Failure |
| Mastercard | 5105 1051 0510 5100 | Failure |

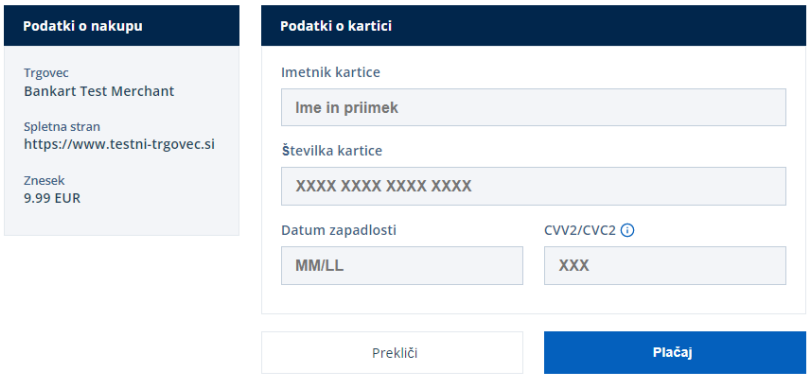
*Slika 23: Testne VISA in Mastercard kartice*

Spodaj so številke **Diners testnih kartic**, v primeru, da imate sklenjeno pogodbo z Diners oziroma Sparkasse Pay Slovenija:
**3800 0000 0000 06** – *pričakovan odgovor je* ***odobrena transakcija***
**3614 8900 6479 13** – *pričakovan odgovor je* ***zavrnjena transakcija***

Kot zgoraj, glede na željen rezultat, ob pravilni integraciji izberite eno izmed številk kartic. V polje, kjer se preverja CVC koda, lahko vpišete katerokoli tri mestno kombinacijo, enako pa velja za datum veljavnosti, kjer se lahko vpiše katerikoli (veljaven) datum.
**Testne kartice so namenjene zgolj za testiranje in jih NI dovoljeno uporabljati v produkciji.**
Pri testiranju v testnem sistemu se **NE** uporabljajo produkcijske kartice, izdane s strani bank, temveč samo testne kartice dogovorjene z Bankartom.

## 4.4. Testiranje s simulacijskim konektorjem

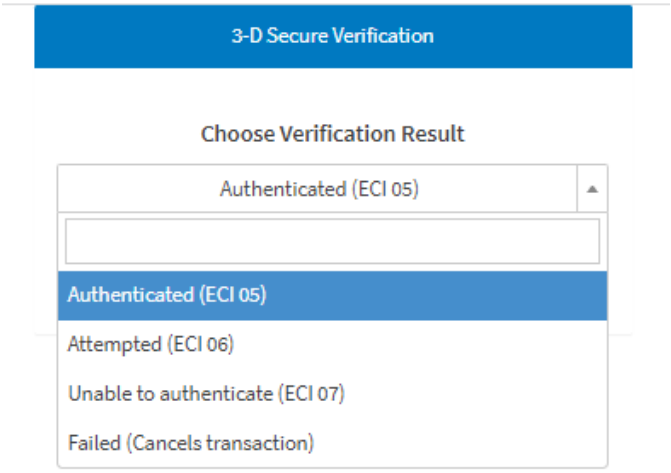
Za potrebe testiranja bomo definirali vašega testnega trgovca, s katerim preverite delovanje svojega sistema. Na plačnikovi strani bo odziv simuliran.
Po tem ko sprožite transakcijo se prikaže vnosna maska (na sliki), če uporabljate preusmeritev na HPP, ali pa se prikaže integrirana vnosna maska na vaši trgovini, če uporabljate payment.js integracijo. V obeh primerih je potreben vnos podatkov testne kartice, ki ste jih dobili s strani Bankarta oz. banke.




*Slika 24: Bankartova nova vnosna maska*

Po vnosu podatkov in kliku na gumb »Plačaj« se v testnem okolju (v produkcijem se ne pokaže) prikaže ekran za simulacijo 3D Secure avtentikacije in možnost izbire rezultata avtentikacije. Kartične sheme

kot indikator, da je bila izvedena močna avtentikacija (SCA) uporabljajo vrednost ECI 05, zato boste v večini primerov izbrali to opcijo ali pa opcijo »Failed«.




*Slika 25: Možnost izbire rezultzata avtentikacije pri simulacijskem konektorju*

- **ECI 05:** Transakcija je avtenticirana
- **ECI 06:** Pri transakciji je bil neuspešno izveden poskus avtentikacije
- **ECI 07:** Transakcija ni bila avtenticirana
- **Failed:** Avtentikacija se ne izvede, transakcija se prekine

Po kliku na gumb submit, se transakcija izvede do konca, kjer se potem prikaže ekran z rezultatom uspešnosti transakcije.

## 4.5. Testiranje xPays plačilnih metod

V primeru plačilnih metod xPays (npr. Apple Pay in Google Pay) bo v fazi testiranja potrebno izvesti nekaj dodatnih korakov.

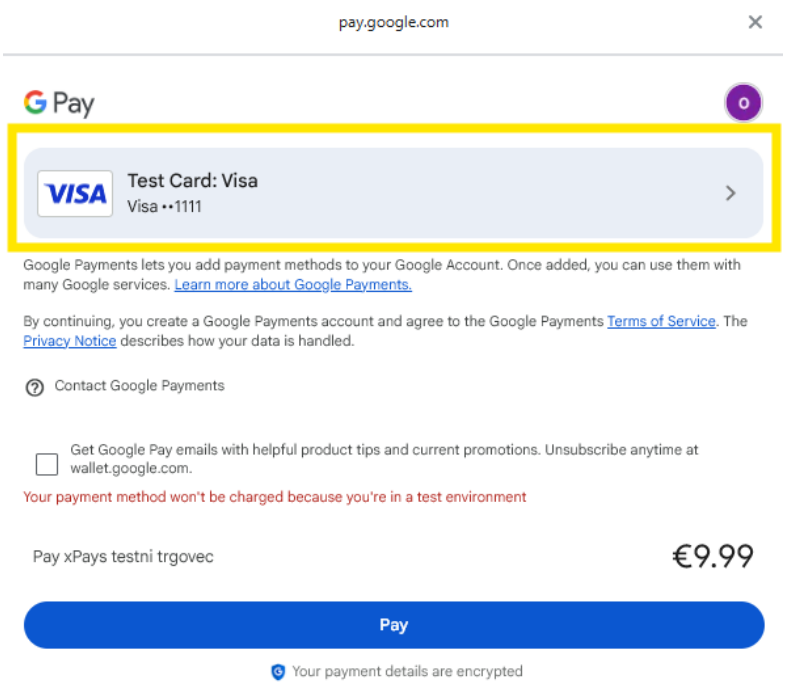
### 4.5.1. Testiranje Google Pay in Apple Pay

V primeru, ko trgovec uporablja integracijo preko Bankartove HPP strani, bodo gumbi za plačilne metode (oz. gumb za tisto metodo, ki jo trgovec podpira) že vključeni v simulacijski konektor. Trgovec bo moral le klikniti na gumba za Google Pay in/ali Apple Pay, izbrati kartico ter preveriti, ali integracija deluje pravilno.

Pri Google Pay se testne kartice prikažejo samodejno, njihova menjava pa je možna s klikom na trenutno izbrano kartico. Nato trgovec zaključi transakcijo s klikom na gumb »Plačaj«.

Za uspešno testiranje mora biti rezultat transakcije uspešen. V skladu s smernicami Googla in Appla, mora biti pravilno tudi prikazan ekran z rezultatom transakcije. Torej, da je v primeru uspešne transakcije, na ekranu omenjeno, da je bilo plačilo izvedeno z Google ali Apple Pay.

Če želi trgovec prikazati Google Pay gumb v iFrameu na svoji spletni trgovini, je potrebno preveriti pravilnost prikaza gumba in nato iti čez celotno uporabniško izkušnja plačevanja.




*Slika 26: Prikaz možnosti izbire testnih kartic pri GPay (označeno z rumeno)*

Pri Apple Pay pa je postopek nekoliko bolj zapleten. Priporočeno je testiranje na starejših iPhonih ali na Apple napravah, ki niso v redni uporabi, je potrebno vpisati se z namenskim testnim Apple ID-jem, ki je namenjen izključno dodajanju testnih kartic v Apple Wallet in preizkušanju Apple Pay funkcionalnosti na spletu.

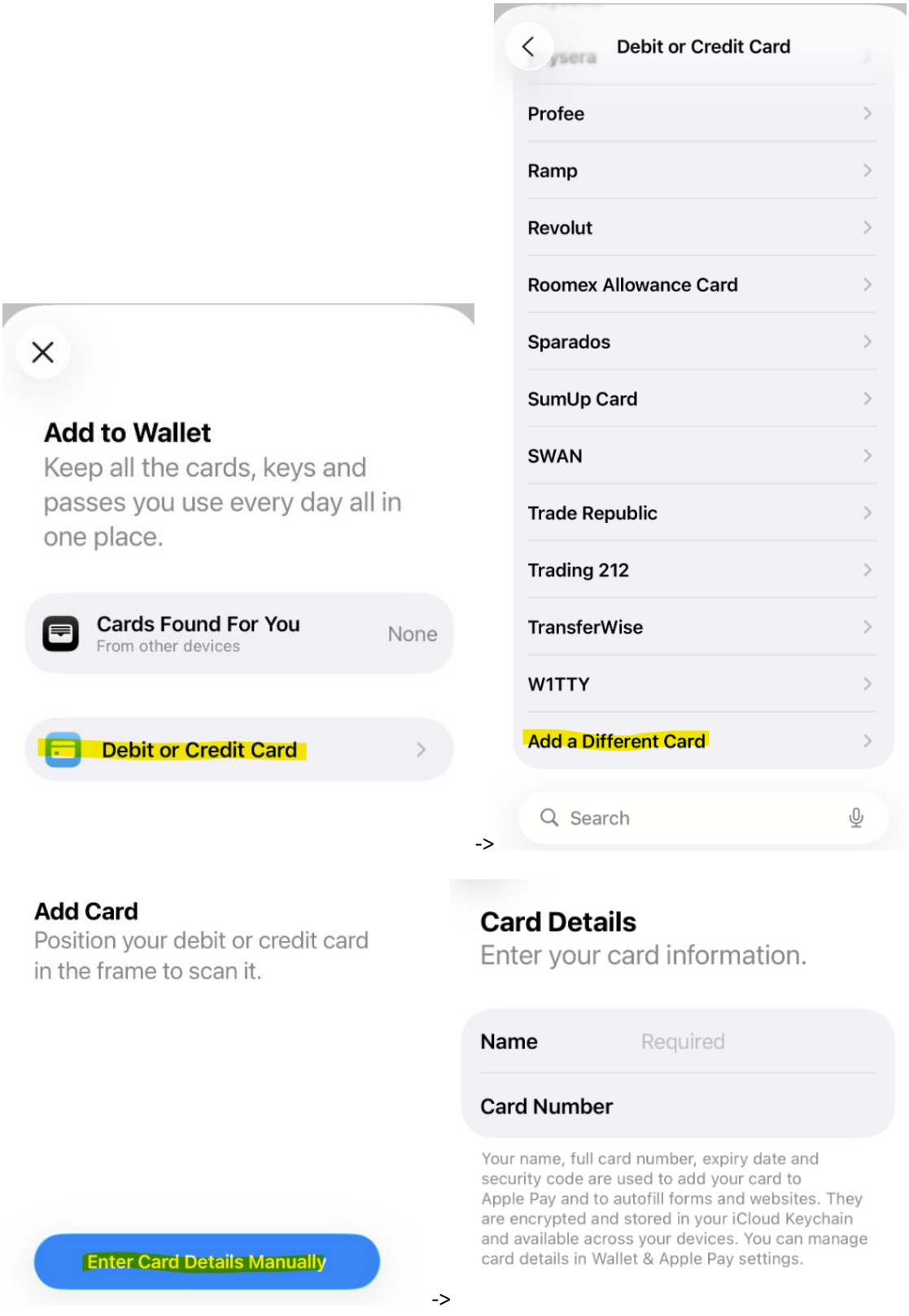
V tem primeru mora trgovec kontaktirati Bankartov Customer Support (najbolje kar z odgovorom na prejeti e-mail) in izraziti željo po testiranju Apple Pay integracije. Bankart bo dodal trgovčev testni e-mail naslov med testne uporabnike.

Poslan e-mail naslov ne sme biti povezan z nobenim obstoječim Apple ID računom, saj v nasprotnem primeru testni uporabnik ne bo uspešno kreiran na Applovi strani. Za ta namen priporočamo uporabo e-mail naslova, ki že obstaja, a ni vezan na Apple ID (npr. službeni e-mail zaposlenega) ali pa ustvaritev novega začasnega e-maila, namenjenega izključno testiranju Apple Pay integracije.

Ko trgovec prejme povratno informacijo od Bankarta, da je bil njegov račun uspešno dodan med testne uporabnike, se mora z navedenim Apple ID-jem prijaviti na eno od Apple naprav (iPhone, iPad ali Mac).

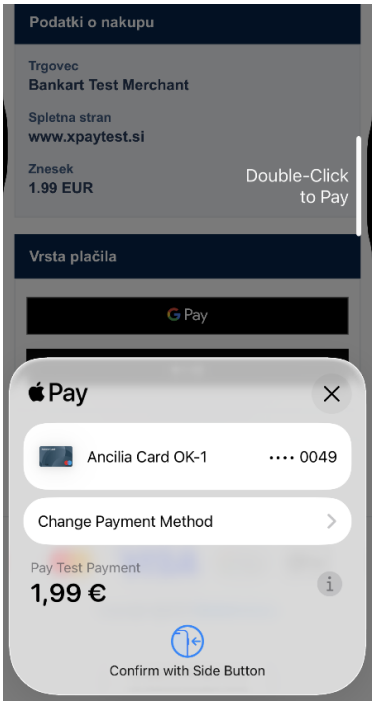
Nato je potrebno v aplikacijo Apple Wallet dodati dve testni kartici – eno Mastercard in eno Visa. Podatki do teh testnih kartic so na voljo na naslednji povezavi:
https://developer.apple.com/apple-pay/sandbox-testing/ .

Spodaj so prikazani koraki za dodajanje testne kartice v aplikacijo Apple Wallet.


Add to Wallet
Keep all the cards, keys and passes you use every day all in one place.
Cards Found For You
From other devices
None
Debit or Credit Card
Debit or Credit Card
Profee
Ramp
Revolut
Roomex Allowance Card
Sparados
SumUp Card
SWAN
Trade Republic
Trading 212
TransferWise
W1TTY
Add a Different Card
Search
->
Add Card
Position your debit or credit card in the frame to scan it.
Enter Card Details Manually
Card Details
Enter your card information.
Name
Required
Card Number
Your name, full card number, expiry date and security code are used to add your card to Apple Pay and to autofill forms and websites. They are encrypted and stored in your iCloud Keychain and available across your devices. You can manage card details in Wallet & Apple Pay settings.
->
Slika 27: Prikaz vnosa testne kartice v Apple Wallet

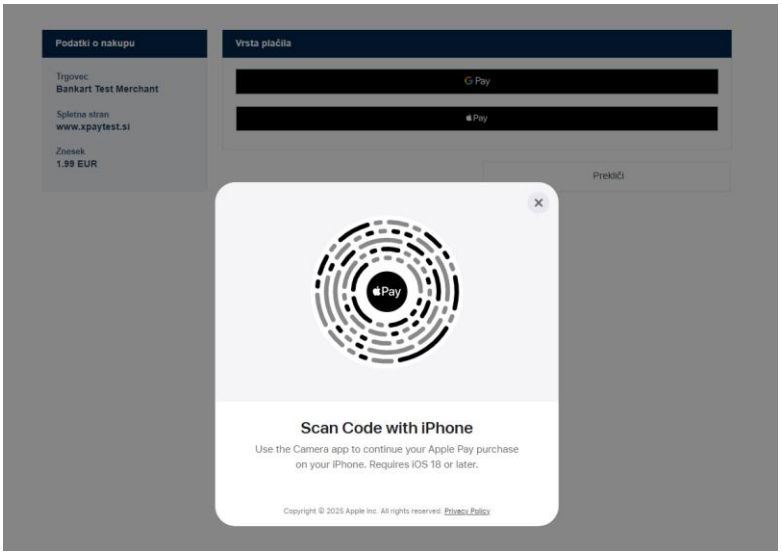
Ko je testna kartica uspešno dodana v Apple Wallet, je potrebno na Apple napravi (iPhone, iPad ali Mac) odpreti ali poslati povezavo do testne transakcije. Po kliku na gumb Apple Pay se – v primeru pravilno dodane kartice – prikaže spodnji zaslon.

Nakup se nato potrdi z dvojnim pritiskom na gumb za vklop/izklop naprave.




*Slika 28: Ekran za potrditev plačila na iPhonu*

Ko je delovanje uspešno testirano na Apple napravi, je potrebno izvesti še testiranje na napravah, ki niso Applove (npr. Windows računalnik ali Android naprava). V tem primeru se na zaslonu prikaže QR koda, ki jo je treba skenirati z iPhonom.
Po skeniranju kupec nadaljuje in zaključi nakup neposredno na svoji Apple napravi.




*Slika 29: Applova QR koda, ki omogoča plačilo z Apple Pay v brskalnikih, ki niso Safari*

Ko so testi uspešno zaključeni mora trgovec javiti Bankartovemu Customer Supportu, ki Apple in Google Pay terminal vklopi v produkciji.

## 4.6. Prehod v produkcijsko okolje

- Bankartov customer support preveri status integracije preko logov na Payment Gateway-u:
  Preden odobrimo produkcijski dostop, naša podpora za stranke preveri loge Payment Gateway-a, če so na simulacijskem adapterju poteki plačil pravilno izvedeni. Če je vse v redu oz. so transakcije pravilno izvedene in označene, bo Bankartova ekipa za pomoč uporabnikom ročno vklopila možnost uporabe produkcijskega terminala.
- Tehnična podpora samo v delovnem času:
  Pomembna opozorilo, ki ga vedno pošljemo v e-poštnih sporočilih za onboarding trgovcev, je ta, da je tehnična podpora, na email naslovu customer.support@bankart.si, na voljo samo v delovnem času (8:00-15:00) med delovnim dnem, saj lahko samostojni razvijalci delajo tudi izven rednih delovnih ur ali med vikendom.
- Trgovci naj **NE** testirajo v produkcijskem okolju!
  Pomembno je, da trgovci ne izvajajo sprememb na produkcijski instanci svoje povezave s Payment Gateway-em. S tem namenom trgovcem nudimo testno okolje, v katerem se potrdi delovanje , ista konfiguracija pa se nato omogoči v produkcijskem okolju. Na ta način zagotovimo, da v trenutku, ko trgovec prejme produkcijski dostop je njegova integracija s Payment Gateway-om delovala. Če naknadno pride do sprememb v produkcijskem okolju BREZ dodatnega testiranja, odgovornost za morebitne težave pri izvedbi poameznih transakcij prevzema trgovec. Če trgovec naredi kakršne koli pomembne spremembe v nastavitvah v produkcijskem okolju, zlasti izven delovnega časa, vam takojšnje tehnične pomoči ne moremo zagotoviti.

## 4.7. Postopek ukrepanja ob zavrnitvi transakcij z določenim razlogom

V določenih primerih bo trgovec v Postback zahtevi na Callback URL prejel tudi podatek o razlogu, zakaj je bila določena transakcija zavrnjena. Ta podatek se bo nahajal v spodnjem polju, kjer je "x" številka/koda, ki se pošlje:

```
"extraData": {
    "psp:fn.respCdeCat": "x"
}
```

Na podlagi vrednosti v polju *"psp:fn.respCdeCat",* znotraj objekta *"extraData"* je trgovec v določenih primerih dolžan ukrepati, kar vključuje dodatno opozorilo stranki, da za določeno obdobje ne izvaja več transakcij na njegovi strani, ali pa jo celo začasno blokira, če s to kartico poskusi izvesti nove transakcije.

Spodaj so navedene kode (*"x"* v zgornjem primeru), njihove razlage ter ustrezni ukrepi, ki jih mora trgovec sprejeti v posameznih primerih:

- **"psp:fn.respCdeCat":"1"**
  Transakcija z vnešenimi kartičnimi podatki kupca ni bila odobrena, kar pomeni, da je bil vnesen en ali več napačnih podatkov. Kupec mora ponovno vnesti pravilne podatke, da bi lahko transakcija uspešno stekla.
- **"psp:fn.respCdeCat":"2"**
  Transakcija z vnešenimi kartičnimi podatki trenutno ne bo odobrena. Priporočamo, da kupec poskusi kasneje. V določenih primerih, odvisno od statusa kartice, se lahko v objektu "extraData" pojavita dve dodatni polji, ki določata časovni okvir za ponovni poskus transakcije:
  - najprej polje **"psp:fn.retryTim",** ki vsebuje vrednost od 0 do 99 in označuje čas v minutah, urah ali dnevih, ki ga je potrebno počakati pred ponovnim poskusom.
  - in še polje **"psp:fn.retryPrd",** ki označuje enoto časa, vezano na prejšnje polje:
    - **"0"** pomeni minute,
    - **"1"** pomeni ure,
    - **"2"** pomeni dneve.

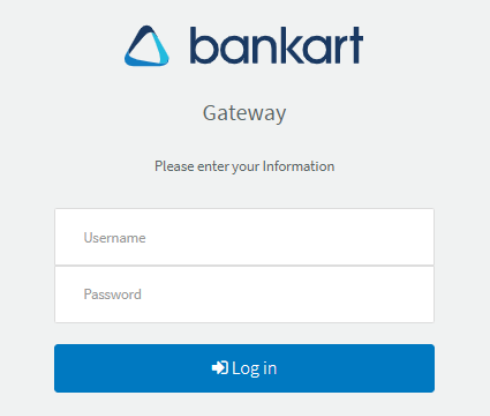
- **"psp:fn.respCdeCat":"3"**
  Transakcija z vnešenimi kartičnimi podatki kupca ne bo nikoli odobrena, saj je kartica blokirana na strani kartične sheme. Kupec naj s to kartico ne poskuša več izvesti transakcij.

Za zgoraj navedene primere mora trgovec pripraviti ustrezne statusne strani za napake, ki bodo vsebovale obvestila, ki jasno odražajo pomen posameznih statusov zavrnitve.
Pomembno je poudariti, da se te kode zavrnitve ne bodo pojavile pri vsaki transakciji, temveč le v primerih, ko je bila transakcija izvedena s kartico ene izmed tujih bank.

# 5. BANKART PAYMENT GATEWAY: PREGLED UPORABNIŠKEGA VMESNIKA

## 5.1. Dostop do portala

Spletni vmesnik je dostopen na naslovu:
https://gateway.bankart.si/en/login
Uporabniško ime in geslo za spletni dostop sta ločena od dostopa za API vmesnik. Ob urejanju formalnosti z banko boste posredovali tudi e-mail naslov, ki bo uporabljen tudi v primeru potrebe ponastavljanje gesla.
Vse spremembe podatkov urejate z vašo banko.
V kolikor imate težave s prijavo se prosim obrnite na customer.support@bankart.si za pomoč. Tja se lahko obrnete tudi v primeru nejasnosti ali težav pri uporabi vmesnika.

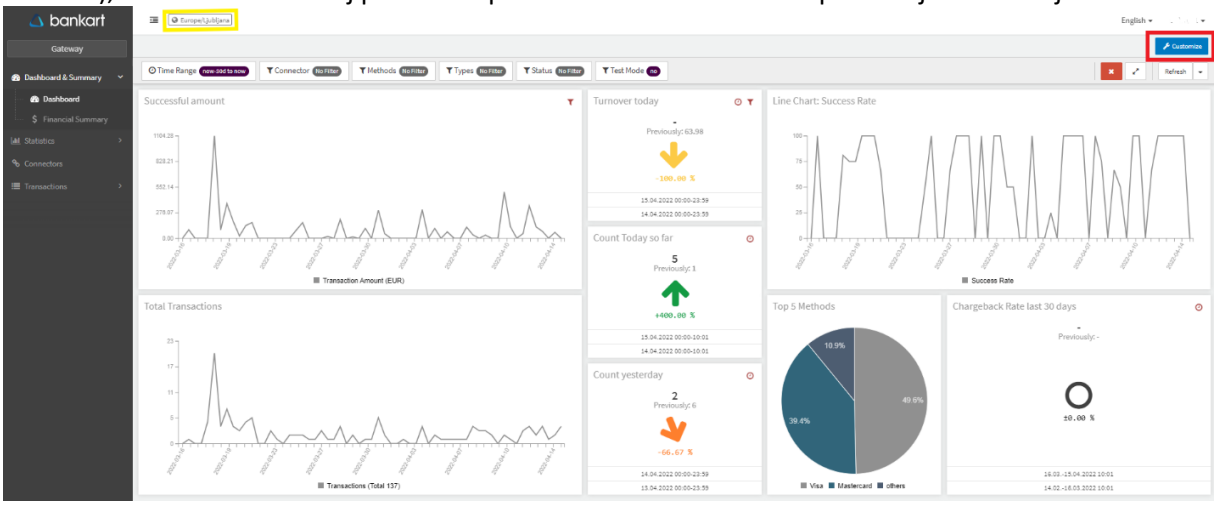



*Slika 30: Vstopna točka za Payment Gateway portal*

## 5.2. Pregled grafičnega vmesnika

Po vnosu podatkov in vpisu, se pojavi domača stran oz. dashboard portala.
Na levi strani v temno sivem delu so na voljo različni zavihki, kjer lahko izbirate za prikaz želenih podrobnosti. Najbolj uporabna zavihka za vas sta **Dashboard & Summary** ter **Transactions**.

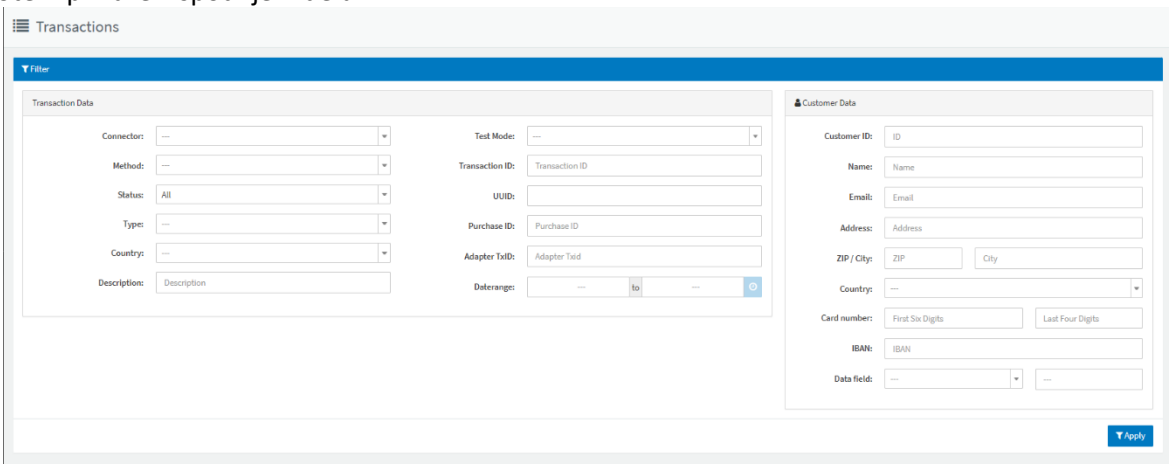
Po uspešni prijavi se prvi prikaže zavihek grafičnega vmesnika, »Dashboard«, kjer je pregled nad statistiko vaših transakcij. Tu lahko s klikom na desni zgornji gumb »Customize« *(označeno z rdečo)* prilagajate filtre/parametre in glede na vaše želje prikažete željene grafe in številke.
Ob prvi prijavi levo zgoraj preverite še nastavljen časovni pas na Europe/Ljubljana (označeno z rumeno), da bo čas transakcij prikazan v pravem času in ne bo težav pri iskanju transakcij.




*Slika 31: Prva stran v Payment Gateway portalu*

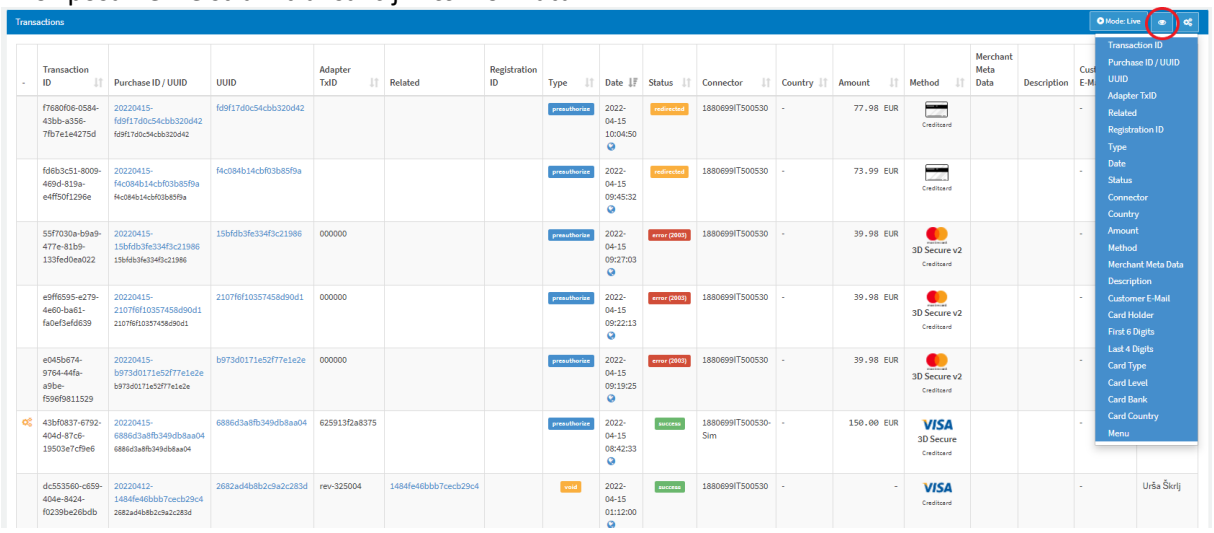
Na levi strani v sivem meniju z več zavihki, lahko izberete različne funkcije, ki jih uporabniški vmesnik ponuja. Kot omenjeno, bo za vas bo verjetno najbolj uporaben zavihek »Transactions«, kjer dostopate do seznama in informacij o transakcijah proženih preko vaše spletne trgovine.

V zgornjem delu strani lahko s pomočjo različnih parametrov poiščete točno določeno transakcijo, ki se potem prikaže v spodnjem delu:




*Slika 32: Prikaz filtrov za iskanje transkacij v Payment Gatewayu*

V spodnjem delu se v osnovi prikaže seznam vseh proženih transakcij, kjer so transakcije časovno razvrščene, od novejše proženih, proti starejšim. Na spodnji sliki sta desno zgoraj v modri vrstici vidna dva gumba. S klikom na levi gumb *(označen z rdečo na spodnji sliki)* se prikaže podmeni, kjer prilagodite vidnost posameznih polj na seznamu glede na vaše potrebe. Če je ime polja obarvano z modro, potem je to polje prikazano, če pa je obarvano sivo pa ni prikazano. Z desnim gumbom pa lahko naredite hitri izvoz posamezne strani transakcij v *.csv* formatu.




*Slika 10: Možnost izbire prikaza polj v zavihku transakcije*

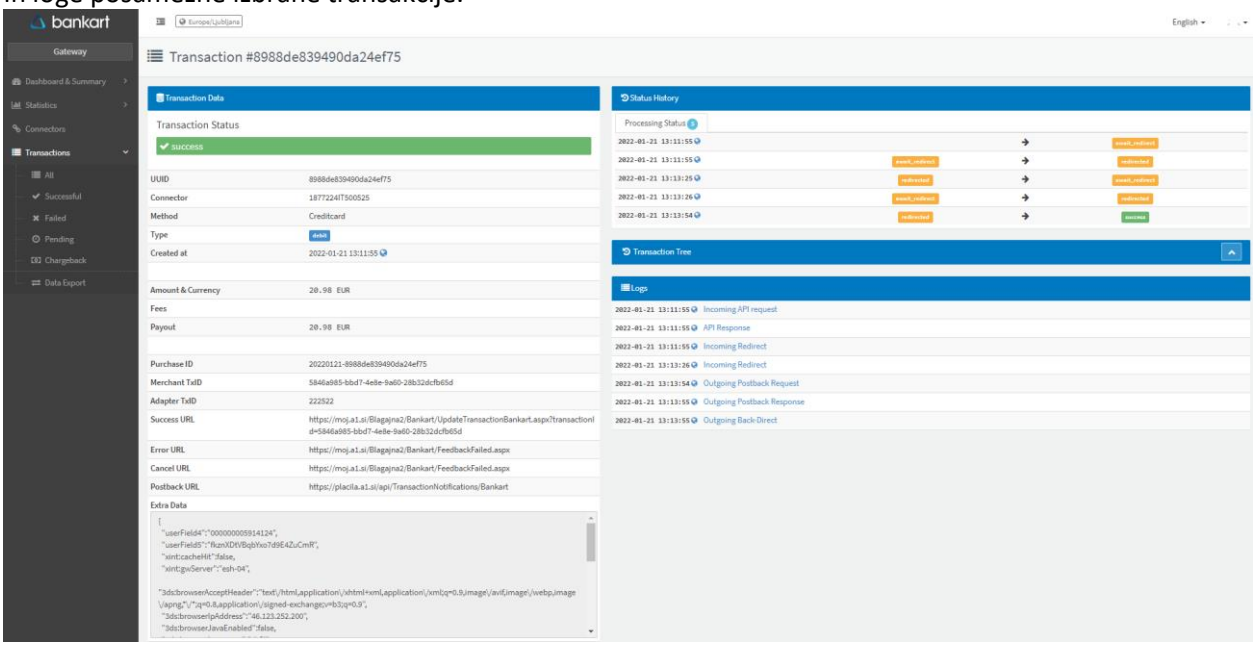
Pomembnejša polja na seznamu transakcij so:

- ***Transaction ID***: identifikator transakcije, ki ga nastavi in v sistem pripelje trgovec, ter mora biti unikaten za vsako transakcijo. Pri predavtorizaciji ter posledično zajemu oz. stornaciji morajo biti ID-ji različni)
- ***Purchase ID/UUID:*** pod to vrednostjo lahko uparjate predavtorizacijo z zajemom (capture) oziroma stornacijo (void). UUID je zgolj unikaten ID transakcije, ki ga nastavi sam Payment Gateway.

- ***Type:*** tip transkacije, ki je bil prožen (debit, predavtorizacija, zajem, stornacija…)
- ***Status:*** uspešnost transakcije
- ***Amount:*** znesek transakcije
- ***Method:*** kartična shema kartice in verzija avtentikacije uporabnika

### 5.2.1. Pregled podrobnosti (logov) posamezne transakcije

Na seznamu transakcij s klikom na gumb »Details« (ki se pri vsaki transakciji v seznamu nahaja skrajno desno), ali pa s klikom na številko v polju UUID, se prikaže vmesnik, kje si lahko ogledate podrobnosti in loge posamezne izbrane transakcije.

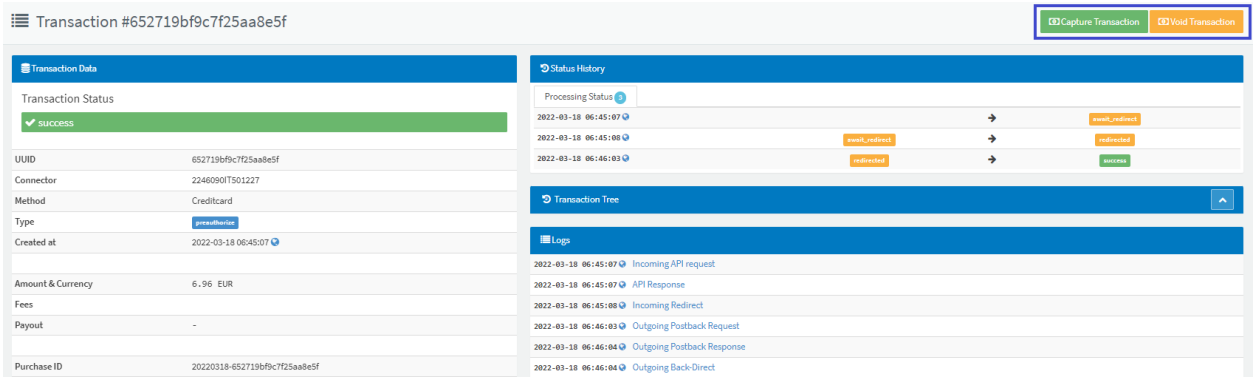



*Slika 34: Prikaz strani s podrobnostmi posamezne transakcije*

Tukaj lahko v razdelku »Logs« razvijalci sami preverijo tudi vsebino izmenjanih sporočil med trgovčevim sistemom in Payment Gateway sistemom (opomba: vidna so le uspešno avtenticirana sporočila).

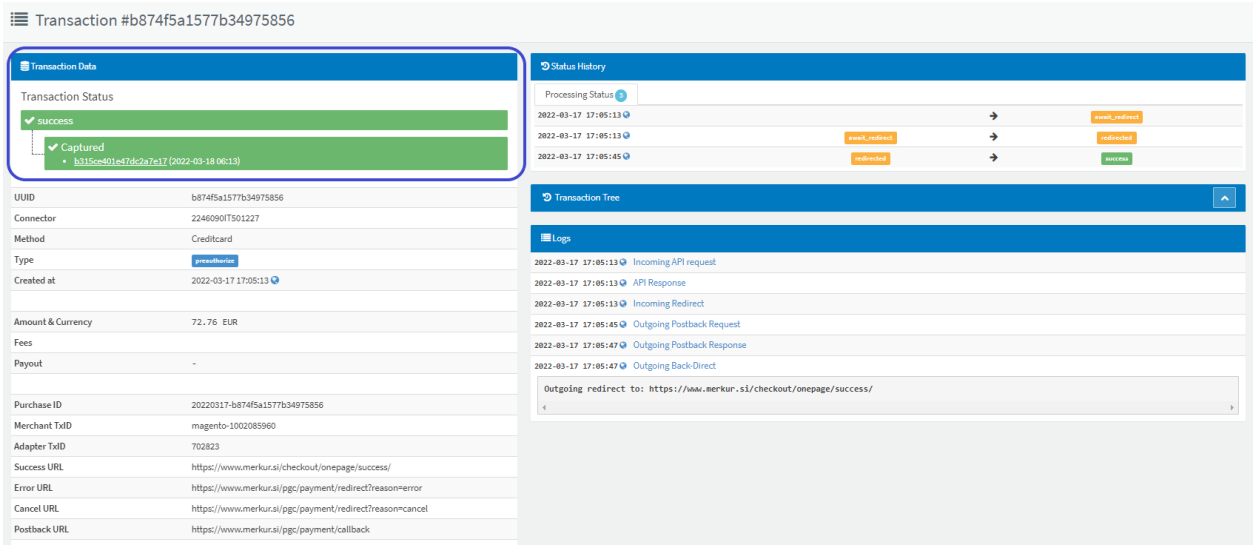
### 5.2.2. Zajemi in stornacije transakcij

V primeru transakcije tipa predavtorizacija, vam je znotraj uporabniškega vmesnika zgoraj desno ponujena možnost ročnega zajema (Capture) ali stornacije (Void) *(označeno z modro)*.




*Slika 35: Prikaz izbire zajema/storno posamezne transakcije*

Ko boste transakcijo zajeli, oz. je predavtorizacija že zajeta, bo to vidno v predelu »Transaction Status«. Podana bo tudi povezava do zajete transakcije. Ob kliku na povezavo boste preusmerjeni na to zajeto transakcijo.

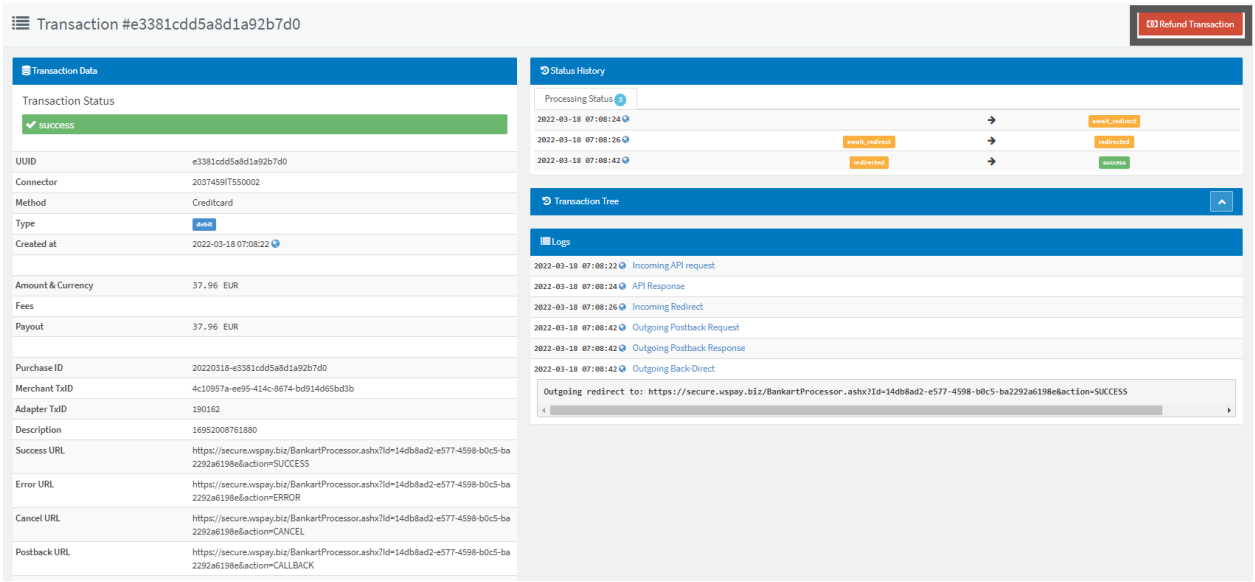



*Slika 36: Prikaz kje se vidi referenčna (zajeta, strornirana) transakcija*

*Opomba:* zneski »fees« in »payout« niso vidni. Za nakazila in provizije se je potrebno prijaviti na trgovski portal, kjer so vidni finančni izpiski.

### 5.2.3. Povračilo zneska transakcij (refund)

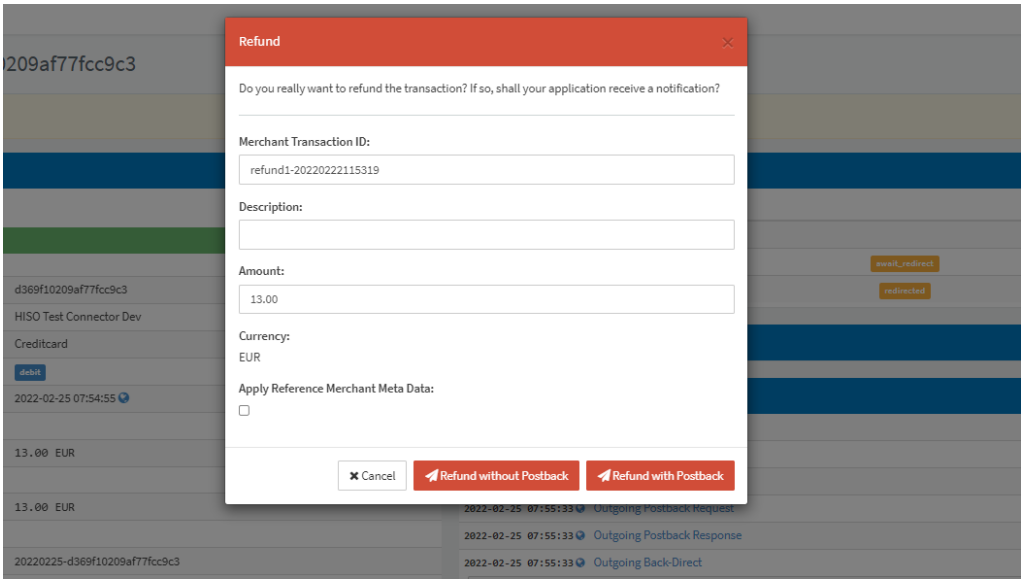
Če kupec s kupljenim izdelkom ni zadovoljen oz. do kupca ne pride ustreznem stanju IN ima trgovec možnost vračila stroškov (refunda), lahko to akcijo proži kar preko grafičnega vmesnika ob vpogledu podrobnosti transakcije. Gumb za »Refund« se prikaže na zgornji desni strani (*označeno s temno sivo*).




*Slika 37: Prikaz kako znotraj Payment Gateway portala narediti refund/vračilo*

Ob proženju refunda preko uporabniškega vmesnika, se odpre pop-up, ki ponudi dve opciji: »Refund without Postback« in »Refund with Postback«.

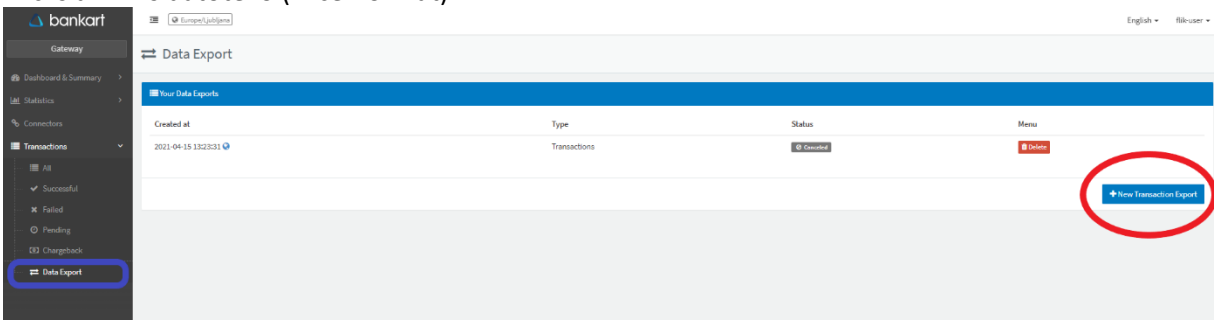
Opcija »Refund with Postback« pošlje nov status transakcije na trgovčev prej definiran Callback URL, če ta ni določen, pa trgovec ob proženju refunda preko uporabniškega vmesnika Payment Gateway-a NE dobi spremembe statusa transakcije.




*Slika 38: Ekran, ki se pojavi ob proženju refunda prek portala*

### 5.2.4. Izvoz seznama izvedenih transakcij

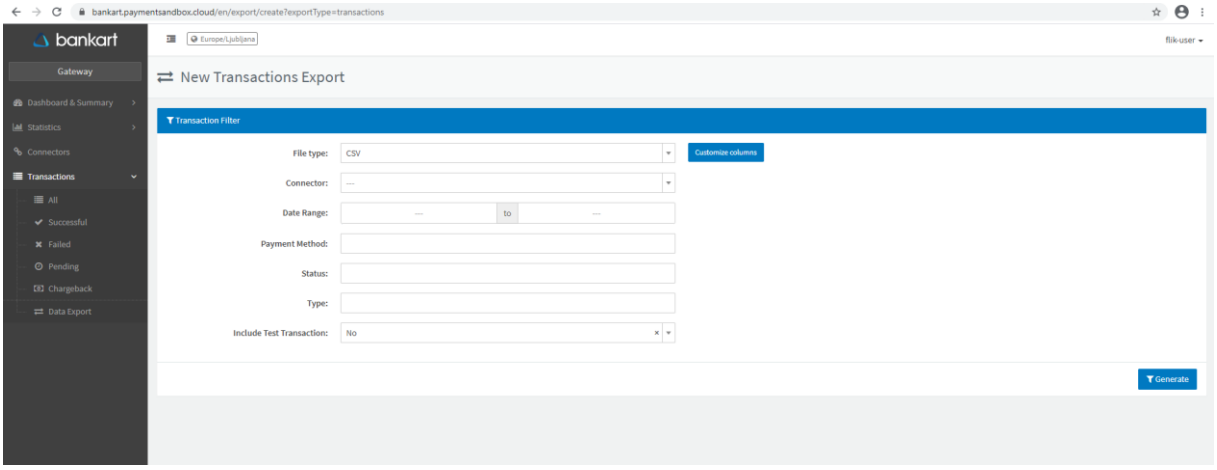
V podzavihku **Data Export** (*na spodnji sliki označen z modro*) lahko prefiltrirate transakcije in jih izvozite v .cvs ali .xlxs datoteko (Excel format).




*Slika 39: Zaslona za export transakcij*

Ob kliku na podzavihek se prikažejo vsi dokumenti, ki so bili do določenega trenutka zgenerirani za izvoz. Za vsak izvozni dokument so vidni datum izvoza transakcij, tip podatkov v izvoznem dokumentu, status in možnost izbrisa (Delete) ali prenosa (Download).

Dokument izvozite s klikom na gumb *»New Transaction Export«* (*označen z rdečo na zgornji sliki*).




*Slika 40: Prikaz zaslona s filtri pri exportu transakcij*

Ko se odpre novo (zgornje) okno izberete željene filtre in tip datoteke za izvoz ter izbiro potrdite s klikom na gumb *»Generate«.* Sledi preusmeritev na prejšnjo stran, kjer je potrebno počakati nekaj trenutkov, da se dokument zgenerira in je ponujena možnost prenosa/brisanja.
Če se po 30 sekundah ti možnosti ne prikažeta, lahko sami osvežite stran.

# 6. NAVODILA ZA UPORABO STORITVE PAY BY LINK - PREKO UPORABNIŠKEGA VMESNIKA NA BANKART PAYMENT GATEWAYU

## 6.1. Opis storitve Pay by Link

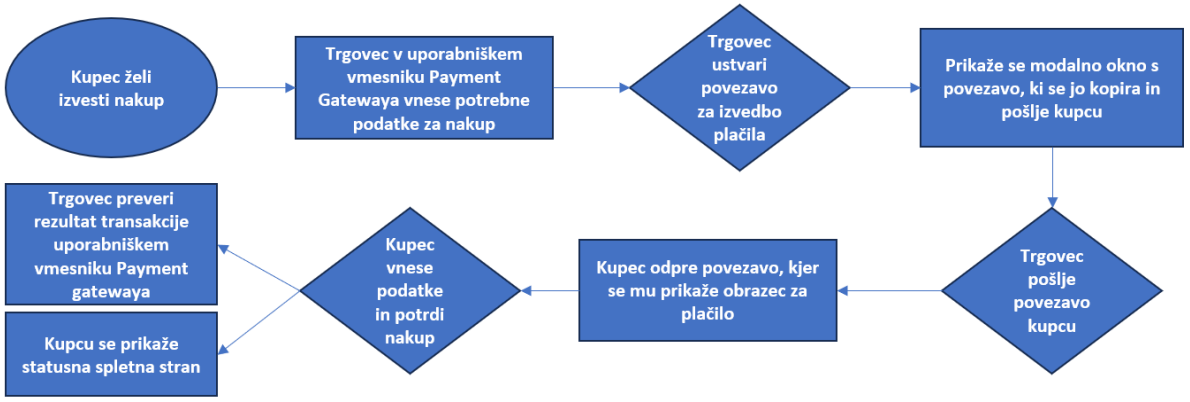
Pay by Link je storitev, ki je ponujena v sklopu Bankartovega Payment Gatewaya in omogoča proženje spletnih plačil preko povezave za izvedbo plačila. Povezavo za izvedbo plačila trgovec kreira v uporabniškem vmesniku in jo kupcu pošlje preko komunikacijskega kanala s katerim poteka komunikacija med trgovec in kupcem (elektronsko sporočilo, SMS sporočilo, spletni pogovor ali družbena omrežja). V trenutni fazi je ustvarjeno povezavo potrebno ročno kopirati (s klikom na gumb) in jo prilepiti v izbran kanal komunikacije. Avtomatsko pošiljanje povezave in informacij o transakciji preko elektronske pošte je še v razvoju.

## 6.2. Prednost storitve Pay by Link

Prednost storitve je, da za uporabo trgovec ne potrebuje tehničnega predznanja implementacije plačilnih instrumentov preko spleta, saj je rešitev za trgovca zelo enostavna in pripravljena za takojšnjo uporabo. Storitev je primerna v kolikor trgovec nima svoje spletne strani ali spletne trgovine in v večjem deležu izvaja promocijo oz. ponudbo preko socialnih omrežij ali telefonskih klicev (npr. infulencerji, lastniki počitniških apartmajev, itd.).

## 6.3. Shema delovanja poteka storitve Pay by Link

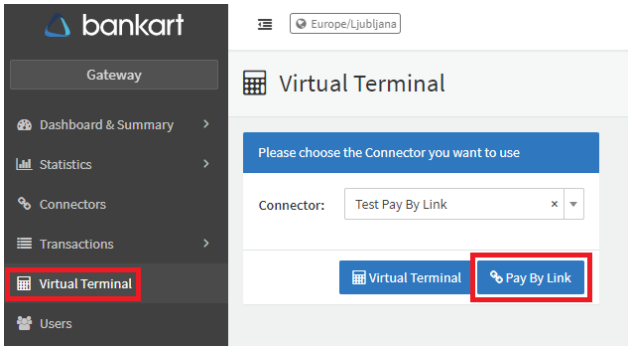
Potek trenutnega delovanja je prikazan v shemi spodaj. Gre za podporo v najbolj osnovni obliki, v naslednjih mesecih pa bodo nekatere funcionalnosti še dodane.




*Slika 41: Shema delovanja*

## 6.4. Potek izvedbe Pay by Link transakcije preko Bankartovega Payment Gateway portala

Ko se trgovec prijavi v uporabniški vmesnik Bankartovega Payment Gatewaya (v nadaljevanju: uporabniški vmesnik) se mu ob levi strani zaslona prikaže stranski meni. Da bi uporabili Pay by Link način plačevanja se izbere možnost »Virtual Terminal«, kjer se odpre razdelek za izbiro konektorja. V nadaljevanju se izbere konektor, ki je dodeljen za uporabo storitve ter se nadaljuje s klikom na gumb »Pay by Link« (na sliki označeno z rdečo barvo).

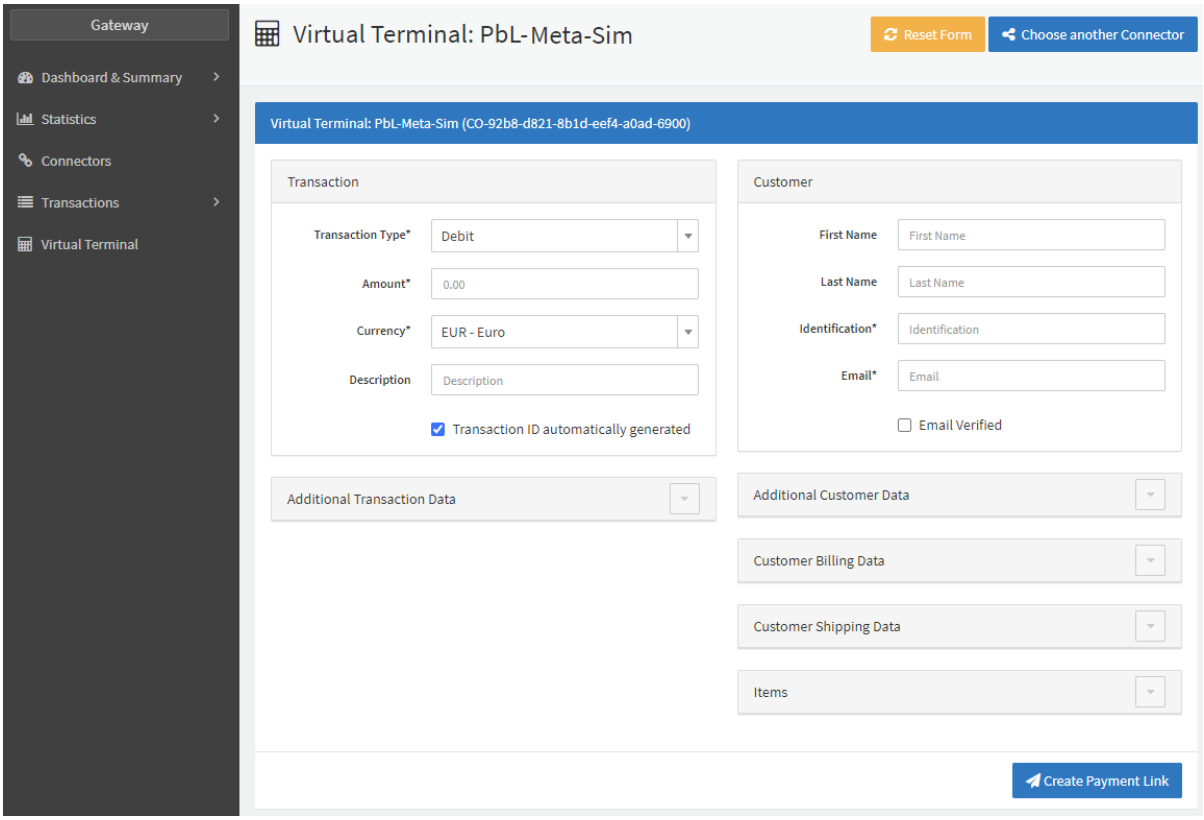



*Slika 42: Kje najti konzolo za inciacijo Pay by Link transakcije*

Razdelki znotraj uporabniškega vmesnika so podrobneje opisani v naslednjem podpoglavju.

**Obvezni podatki za vnos po razdelkih so**:

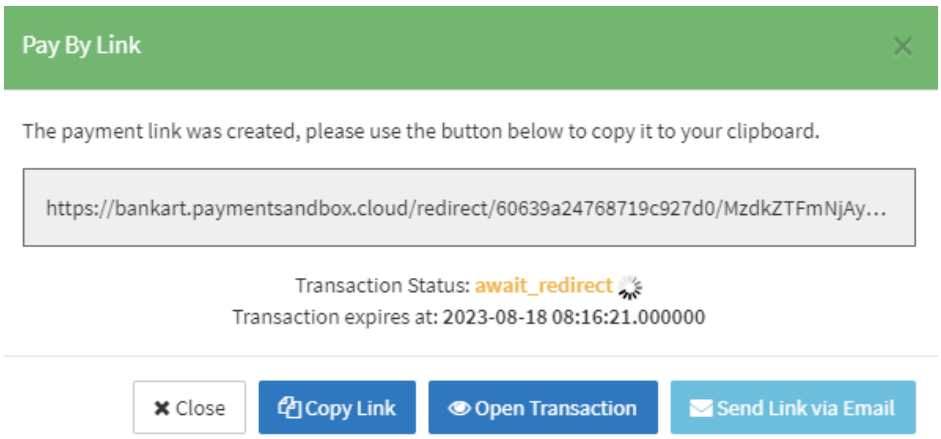
- Razdelek Transakcija (angl. Transaction):
  - tip transakcije, ki bo podprt na strani trgovca,
  - znesek za plačilo,
  - valuta v kateri bo transakcija prožena (trgovec lahko uporabi le valuto, ki mu jo banka omogoči).
- Razdelek Stranka (angl. Customer):
  - identifikacija kupca,
  - kupčev elektronski naslov.
- Razdelek Naslov za račun (angl. Customer Billing Data):
  - naslov 1,
  - mesto,
  - poštna številka,
  - država.




*Slika 11: Konzola Pay by Link za vnos podatkov na Payment Gateway portalu*

S klikom na gumb »**Create Payment Link**« v spodnjem desnem kotu strani, se ustvari povezava za plačilo in odpre modalno okno, na katerem so naslednje možnosti izbire:

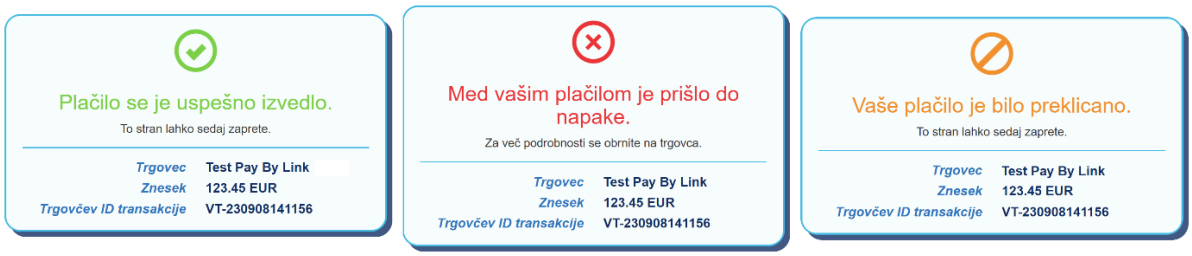
- **Copy Link** – s klikom na gumb se kopira povezava, ki jo potem lahko trgovec ročno prilepi kupcu v kanal, prek katerega poteka komunikacija,
- **Open Transaction** – s klikom na gumb se odpre zaslon s podrobnostmi o transakciji,
- **Send Link via Email** – s klikom na gumb se povezava pošlje preko elektronske pošte, ki je vpisana v razdelku »Stranka« (trenutno ni na voljo),
- **Close** – s klikom na gumb se zapre modalno okno.




*Slika 12: Modalno okno, ki pokaže opcije kaj lahko naredimo*

V oknu se prav tako nahaja podatek o statusu transakcije in podatek o poteku transakcije. Status transakcije je vedno možno preveriti v zavihku s klikom na gumb »Open Transaction« V tem razdelku so na voljo tudi dodatni podatki o transakciji (za kakšno transakcijo gre, potek povezave, ali so na voljo podtransakcije in logi same transakcije).
Povezavo v modalnem oknu trgovec kopira in pošlje kupcu. S klikom na povezavo se kupcu odpre obrazec za spletno plačilo kamor se vnese kartične podatke. Po vnosu podatkov in s klikom na gumb »Submit« se izvede avtorizacija. V večini primerov je pri Pay by Link transakcijah potrebna tudi avtentikacija oz. potrditev spletnega nakupa. Po avtentikaciji in avtorizaciji, je kupec preusmerjen na statusno stran, z informacijo o statusu transakcije/plačila:

- Plačilo je bilo uspešno,
- Plačilo je bilo neuspešno,
- Plačilo je bilo preklicano.




*Slika 38:Statusne strani, z rezultati ob koncu transakcije*

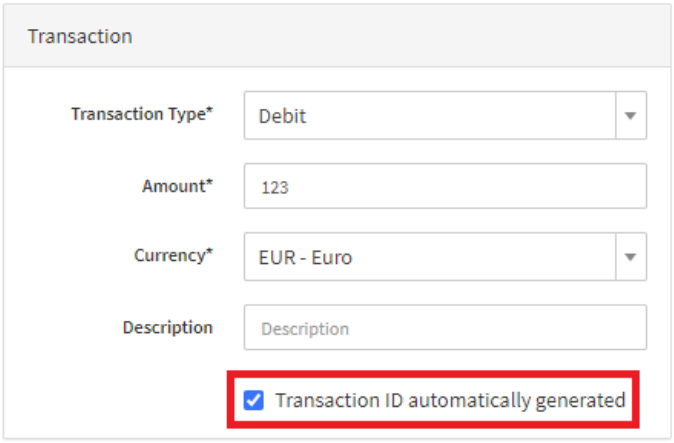
Rezultat transakcije lahko trgovec preveri v uporabniškem vmesniku na strani izvedenega Pay by Link plačila, kjer je na voljo tudi možnost podrobnega pregleda transakcije.

## 6.5. Podroben opis razdelkov za vnos podatkov v maski Pay by Link

Tu so opisani vsi razdelki, ki so relevantni za proženje Pay by Link transakcije. Če razdelek ni opisan, potem polj, ki jih vsebuje ni potreno polniti, saj le ti niso relevantni za proženje transakcije.

### 6.5.1.1. Razdelek Transakcija (angl. Transaction)

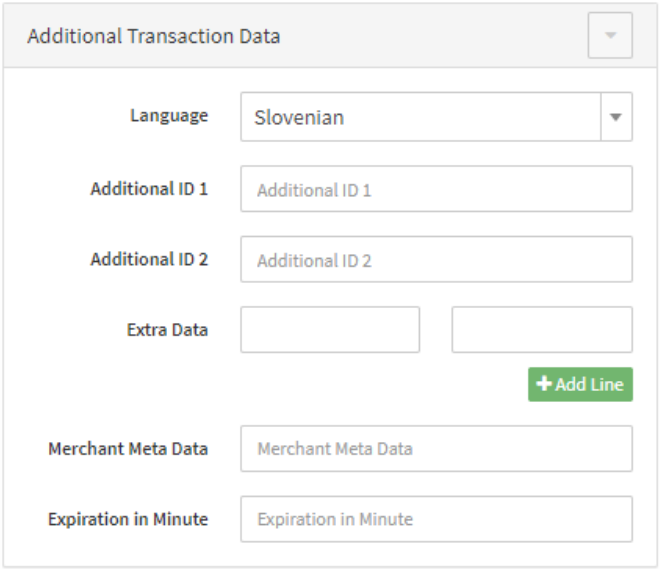
V tem razdelku je potrebno obvezno izpolniti že zgoraj omenjena polja. Najprej mora trgovec določiti enega izmed Tipov transakcij (Transaction Type*), ki so iz strani banke dovoljeni trgovcu.
Obvezno je potrebno določiti tudi Znesek (Amount) za plačilo in Valuto (Currency) države trgovca (EUR, MKD, BAM). V primeru, da se izbere tip transakcije ali valuta, ki ni podprta, se temu primerno izpiše pojavno okno z opozorilom. Dodatno polje Opis (Description), kjer se poljubno vnese podatke za trgovčevo evidenco, kot je recimo ime produkta ali storitve, ki jo trgovec prodaja.
Prav tako je možno določiti, da se ID transakcije samodejno ustvari, če je le to označeno s kljukico (na sliki označeno z rdečo barvo). V primeru, da se trgovec ne odloči za avtomatsko generiran ID nakupa, ga lahko sam določi. *Primer: Trgovec ustvari avtomatsko generirano transakcijo na datum 20.09.2023 ob 8:45:18 uri, ki jo sistem poimenuje »VT-23092084518«. V primeru, da je trgovec pozabil zaračunati dodatno storitev in jo želi dodatno kasneje zaračunati ter želi imeti boljšo evidenco »povezanih« transakcij, lahko na koncu doda »-01« in tako dodatno transakcijo poimenuje »VT-23092084518-01«, ki bo poimenovana skoraj enako kot prvotna transakcija. Trgovec pa lahko le-to določi tudi po nekem svojem ključu, ki ga uporablja za tak ID.*




*Slika 39: Razdelek Transakcija po poljih*

#### 6.5.1.2. Razdelek Dodatni podatki o transakciji (angl. Additional Transaction Data)

V tem razdelku ni obveznih polj, so pa vseeno polja, ki lahko vplivajo na transakcijo. Eno izmed takih polj je izbira jezika (Language) v katerem se vnosna maska za vnos kartičnih podatkov naloži na strani kupca. Ta maska je podprta v več kot 15 različnih jezikih.
Ostala polja so manj pomembna in jih za potrebe proženja transakcij ni obvezno izpolniti in zaenkrat tudi ni predvideno, da se polnijo. Izjema je zadnje polje »Expiration in Minute«.
V primeru, da trgovec želi časovno omejiti čas dostopnosti do kreirane povezave je pomembno, da v polju »Expiration in Minutes« omeji dostopnost povezave z željenim omejenim številom minut. Trgovec vpiše krajši ali enak čas trajanja kot je nastavljen v osnovi: 4320 minut (oz. 72 ur/3 dni). V primeru, da je to polje prazno, bo po privzeto določenem času povezava veljala 72 ur.
*Primer: če bo trgovec vpisal v to polje »60« bo le ta veljal 60 minut in ne 4320 minut (oz. 72 ur).*

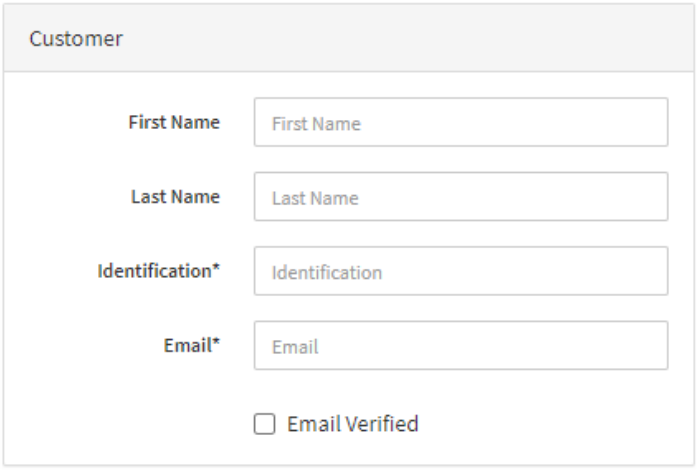



*Slika 40: Razdelek Dodatni podatki o transakciji po poljih*

#### 6.5.1.3. Razdelek Stranka (angl. Customer)

V tem razdelku trgovec vpiše podatke za identifikacijo kupca. Ime (First name) in priimek (Last name) nista obvezna polja. Obvezno polje je identifikacija kupca (Identification), ali pa z drugo besedo rečeno tudi referenčna oznaka kupca. S tem podatkom, bi trgovec lažje identificiral redne stranke in izpolnil tudi polja v naslednjem razdelku »Bančni podatki stranke«. V tem polju se lahko vnesejo črke, številke in simboli.

Obvezno je tudi polje za vnos elektronskega naslova (Email). To polje se uporablja tudi za avtomatsko pošilanje povezave do transakcije, prav tako pa se uporabi za pravilno procesiranje transakcije pri kartičnih shemah.



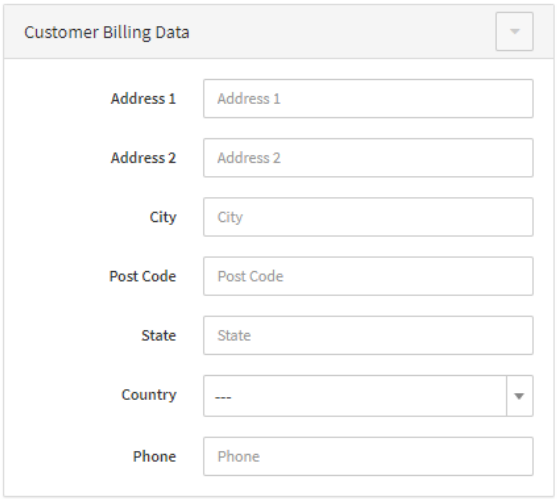

*Slika 41: Razdelek Stranka po poljih*

#### 6.5.1.4. Razdelek Naslov za račun (angl. Customer Billing Data)

Tu je potrebno izpolnit polja, ki določajo prebivališče kupca. Znotraj tega razdelka so določena polja obvezna, čeprav niso označena z »*«, saj so to podatki, ki so pomembni za pravilno procesiranje transakcij pri kartičnih shemah in se razlikuejo med regijami.
Obvezna polja, ki jih je potrebno vnesti v Sloveniji za uspešno procesiranje transakcije so:

- naslov kupca (Address 1),
- mesto (City),
- poštna številka (Post Code),
- država (Country).

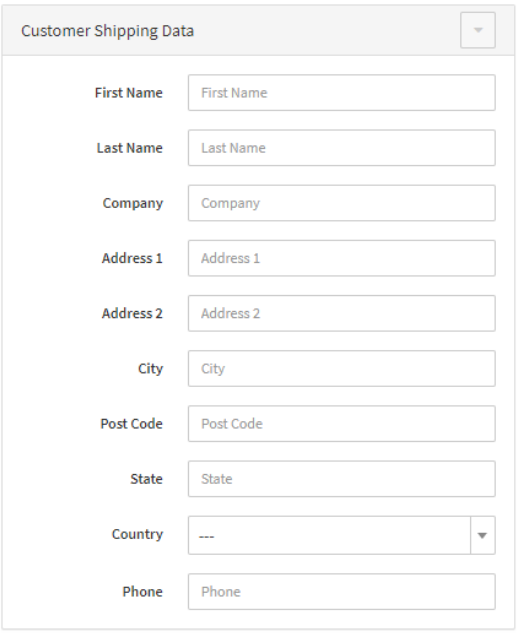
Prav tako lahko trgovec (neobvezno) vpiše telefonsko številko kupca (Phone).




*Slika 42: Razdelek Naslov za račun po poljih*

#### 6.5.1.5. Razdelek Naslov za pošiljanje (angl. Customer Shipping Data)

Vsa polja v tem razdelku so neobvezna. Ti se izpolnijo v primeru, da je dostava naslovljena na drugo osebo ali pa na podjetje, kot na podatke, ki so bili podani v razdelku »Naslov za račun«.

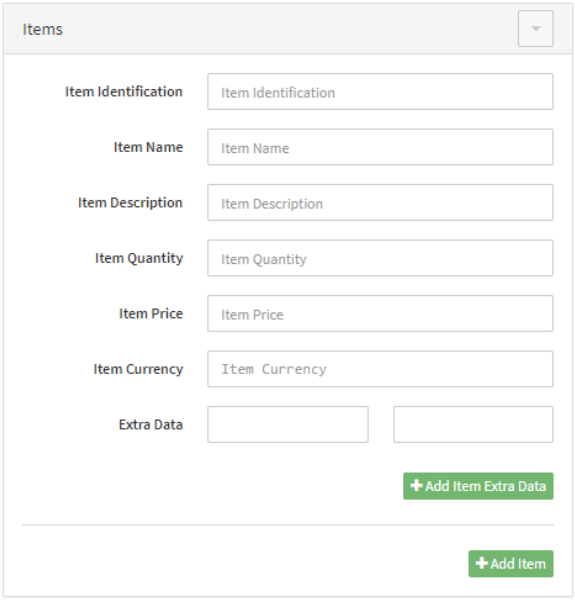



*Slika 13.1: Razdelek Naslov za pošiljanja po poljih*

#### 6.5.1.6. Razdelek Izdelki (angl. Items)

Tudi v tem razdelku so vsa polja neobvezna. V tem razdelku je možno dodati nekaj osnovnih podatkov o prodanih izdelkih/storitvah. Vnese se lahko ID številko izdelka (Item Identification), naziv izdelka (Item name), opis izdelka (Item description), količino (Item Quantity) za vsak posamezen kupljen izdelek, ceno izdelka (Item Price) in ponovno tudi valuto (Item Currency). S klikom na gumb »*Add Item*« je možno dodati več dodatnih izdelkov.

Vnešeni podatki v tem razdelku se kupcu trenutno ne bodo nikjer prikazali in so prikazani le trgovcu znotraj uporabniškega vmesnika na strani pregleda transakcije, kasneje, ko bo podprto, pa bodo prikazani v avtomatsko generiranem emailu ob kreiranju plačilne povezave.




*Slika 14: Razdelek Izdelki po poljih*

# 7. NAVODILA ZA UPORABO STORITVE PAY BY LINK - PREKO API KLICA

Za uspešno implementacijo storitve Pay by Link API (v nadaljevanju PBL API) na strani trgovcev je potrebno imeti osnovno znanje o uporabi Payment Gateway vmesnika, storitve Pay by Link ter razumevanje delovanja API-jev. V tem dokumentu bomo podrobno in postopno predstavili navodila za uporabo storitve PBL API.

## 7.1. Obvezni in neobvezni parametri v PBL API klicu

Pri iniciaciji PBL transakcije preko API klica gre za debit transakcijo, kar pomeni, da se uporablja POST klic, sproži debit transakcijo (opisano v dokumentaciji na Payment Gateway zgoraj). Ključna razlika v API klicu je, da je vključen dodatni array *payByLink*, ki vsebuje nove parametre (v tabeli označena z modro barvo), s čimer se ta klic razlikuje od običajnega debit klica.

| Ime polja | Opis | Obvezen podatek |
|---|---|---|
| amount | Znesek transakcije | DA |
| currency | Valuta uporabljena za transakcijo | DA |
| successUrl | Stran, ki se prikaže kupcu v primeru, da je transakcija uspešno izvedena | DA |
| cancelUrl | Stran, ki se prikaže kupcu v primeru, da je bila transakcija prekinjena iz strani kupca | DA |
| errorUrl | Stran, ki se prikaže kupcu v primeru, da je transakcija neuspešna iz kakršnegakoli razloga | DA |
| callbackUrl | URL, kamor trgovec dobi rezultat in podatke o transakciji | DA |
| sendByEmail | V osnovi je »false«. Več spodaj v podpoglavju | DA |
| merchantTransactionId | ID transakcije pri trgovcu | DA |
| billingAddress1 | Pridobivanje kartičnih podatkov od kupca (opisano na strani 8, klikni podčrtan tekst) | DA |
| billingCity | Pridobivanje kartičnih podatkov od kupca (opisano na strani 8, klikni podčrtan tekst) | DA |
| billingPostcode | Pridobivanje kartičnih podatkov od kupca (opisano na strani 8, klikni podčrtan tekst) | DA |
| billingCountry | Pridobivanje kartičnih podatkov od kupca (opisano na strani 8, klikni podčrtan tekst) | DA |
| email | Email naslov kupca | DA |
| firstName | Ime kupca | NE |
| lastName | Priimek kupca | NE |
| language | Jezik v katerem se prikaže vnosna maska | NE |
| expirationInMinute | Omejitev veljavnosti transakcije. Več info spodaj v podpoglavju. | NE |

**Primer API klica v JSON formatu:**

```
{
            "merchantTransactionId": "20250425123456",
            "amount": "9.99",
            "currency": "EUR",
            "successUrl": "https://primer.si/en/finalize/SUCCESS=1",
            "cancelUrl": " https://primer.si/en/finalize/CANCEL=1",
            "errorUrl": " https://primer.si/en/finalize/ERROR=1",
            "callbackUrl": " https://primer.si/en/finalize/postback",
            "customer": {
                        "firstName": "Janez",
                        "lastName": "Novak",
                        "billingAddress1": "Testni naslov 1",
                        "billingCity": "Ljubljana",
                        "billingPostcode": "1000",
                        "billingCountry": "SI",
                        "email": "janez.novak@kupec.si",
                        "ipAddress": "98.765.432.10"
},
           "language": "en",
           "payByLink": {
                    "sendByEmail": false,
                     "expirationInMinute": 120
           }
}
```


## 7.2. Dodatni parametri v PBL API klicu (payByLink array) - iniciacija transakcije

V API klicu se torej pošlje dodaten array *payByLink*, ki je napolnjen z dvema dodatnima poljema, ki sta podrobnje opisana tu.

### 7.2.1. Polje sendByEmail

Če trgovec želi prožiti Pay by Link API transakcije mora v API Requestu poslati (obvezna polja za API) **"sendByEmail": false**, in **"email": "janez.novak@kupec.si"**, kjer trgovec iz svoje domene/naslova pošlje kupcu mail, ki vsebuje povezavo do transakcije. Poveza do transkacije (vnosne maske) se nahaja v **"redirectUrl"** od koder lahko trgovec prekopira povezavo in jo vključi v mail. (primer "redirectUrl": https://gateway.bankart.si/3ds/challenge/1234a567bc89def0ghij).

### 7.2.2. expirationInMinute

Če trgovec pošlje (neobvezno polje) **"expirationInMinute": 60**, omeji veljavnost transakcije na 60 minut. V primeru, da vnešena vrednost presega Bankartovo privzeto vrednost 4320 minut (72ur oz. 3 dni) se transakcija omeji na 4320 minut.
Dodatna dokumentacija z opisom API polje je na voljo na povezavi: https://gateway.bankart.si/documentation/apiv3?php#transaction-data-pay-by-link

## 7.3. Dodatni parametri v PBL API (payByLinkData array) - odgovor transakcije

Podatki v payByLinkData arrayu so pomembni zato, ker trgovcu povejo do kdaj bo transakcija aktivna in kater URL naslov morajo klicati, če želijo transakcijo prekiniti pred tem časom.

**Primer API JSON odgovora, kjer je izpostavljen omenjen array:**


```
"payByLinkData ": {
      "expiresAt": "2025-04-03 12:34:56 UTC",
      "cancelUrl": "cancelUrl": " https://primer.si/en/finalize/CANCEL=1"
}
```


### 7.3.1. Polje expiresAt

V primeru, da trgovec v API Requestu pošlje vrednost "expirationInMinute", dobi v API Response klicu parameter "expiresAt", ki pokaže datum in čas, do kdaj je veljavnost povezave aktivna.

### 7.3.2. Polje cancelUrl

Trgovec ima prav tako možnost preklicati izvedbo prožene Pay by Link transakcije na način, da z API klicom pokliče "cancelUrl", ki ga dobi v API responsu (enak, kot ga je poslal trgovec).
V vednost: Iz tehničnih razlogov preklic plačila prek povezave Pay by Link ne zagotavlja vedno preprečitve zaključka transakcije. Če je kupec že odprl povezavo do plačilne strani, lahko kljub preklicu še vedno izvede in zaključi plačilo.
Transakcijo je mogoče preklicati le, dokler kupec še ni odprl povezave in sprožil procesa plačila. Da bi se temu izognili, priporočamo, da z uporabo parametra "expirationInMinute" ustrezno omejite veljavnost transakcije glede na vaše potrebe.

## 7.4. Pošiljanje povezave kupcu

Pošiljanje povezave s strežnika banke ali Bankarta trenutno ni podprto. Kljub temu lahko povezavo brez tehnične omejitev kopirate, prilepite in jo kupcu posredujete prek poljubnega komunikacijskega kanala. Povezavo lahko pošljete kupcu preko:

- E-pošte,
- Socialnih omrežij,
- Trgovčeve spletne strani,
- Ali poljubnega kanala komunikacije.

Ko kupec klikne na povezavo, bo preusmerjen na varno plačilno stran, kjer bo lahko vnesel kartične podatke.

## 7.5. Status uspešnih, neuspešnih in preklicanih transakcij

URL naslovi za uspešne, neuspešne in preklicane transakcije so v API Request klicu mandatorni. Trgovec lahko izbere v katerem jeziku se bo kupcu prikazala vnosna maska kot tudi statusna stran transakcije. V primeru, da trgovec ne pošlje izbranega jezika oz. pošlje nepodprto kodo jezika, se stranki plačilna stran prikaže v angleškem jeziku. Trenutni podprti jeziki so opisani na strani 8 v poglavju 3.4.1. Oblikovanje plačilne strani z vnosno masko (zgoraj)
Kaj pomenijo tipi URL naslovov:

- **Uspešna transakcija:** Če kupec uspešno zaključi plačilo, bo preusmerjen na **success_url**. Na tej strani se mora kupcu ob uspešnem plačilu prikazati uspešna statusna stran (stran 35, slika 32).
- **Neuspešna transakcija:** Če se tekom izvajanja transakcije zgodi tehnična napaka bo kupec preusmerjen na **errorUrl**. Na tej strani se mora kupcu ob neuspešnem plačilu prikazati neuspešna statusna stran (stran 35, slika 32).
- **Preklicana transakcija:** Če kupec prekliče plačilo bo preusmerjen na **cancel_url**. Na tej strani se mora kupcu ob preklicanem plačilu prikazati preklicana statusna stran (stran 35, slika 32).

## 7.6. Spremljanje statusa plačila preko Callback URL

Za spremljanje statusa transakcije mora trgovec nastaviti in v API klicu poslati **callback_url**, kamor bo le ta prejel obvestila o statusu transakcije (npr. uspešno plačilo, neuspešno plačilo itd.). Več o tem si lahko preberete na strani 7, poglavje 3.2. Tehnična dokumentacija in zahteve (zgoraj)

# 8. VODENJE SPREMEMB V NAVODILIH (CHANGELOG)

- marec 2025:
  - Ravnanje trgovcev v primeru zavrnjene transakcije (stran 25, podpoglavje 4.6.);
  - Navodila za uporabo storitve Pay by Link preko API klica (stran 40, poglavje 7.).
- april 2025:
  - Večkratni delni zajem ene transakcije – Multiple Capture transakcije (stran 10, podpoglavje 3.5.1.1).
- december 2025
  - Uporaba xPays plačilnih metod (Google Pay, Apple Pay, Click 2 Pay) - stran 19, podpoglavje 3.6.;
  - Testiranje xPays plačilnih metod – stran 31, podpoglavje 4.5;
  - Odstranitev Večkratni delni zajem ene transakcije – Multiple Capture transakcije, dokler ne bo polne podpore.