

EON MegaPOS 2.0

Dokumentacija

v 0.821

0 Vsebina

0	Vsebina	2
1	Uvod.....	3
1.1	Namen.....	3
1.2	Plačilna sredstva	3
1.3	Podpora	3
1.4	Delovanje	3
1.5	Varnost	4
1.6	Zahteve	4
2	Spletna plačila v sistemu MegaPOS.....	5
2.1	Nastavitev trgovine in procesnih centrov.....	5
2.2	Vrste transakcij.....	5
2.3	Življenski cikel transakcij.....	5
2.4	Operacije na transakcijah.....	6
2.5	Inicializacija transakcije in zajem kupčevih podatkov.....	6
3	Integracija sistema MegaPOS v spletno trgovino	7
3.1	Točke integracije MegaPOS-a v spletno trgovino	7
3.2	UPDATE stran	8
3.3	STATUS stran	9
3.4	Uporaba spletne storitve MegaPOS Processor	9
3.5	Procesor Activa Banka Koper.....	10
4	Definicija spletne storitve MegaPOS Processor.....	11
4.1	Operacije.....	11
4.2	Vhodni parametri.....	12
4.3	Izhodni parametri.....	13
4.4	Odgovorne oznake in sporočila	14
5	Implementacija.....	15
5.1	JAVA.....	15
5.2	PHP.....	16
6	Administrativni spletni vmesnik MegaPOS Commerce	17
7	Pogosto zastavljena vprašanja	17
7.1	Kakšne so razlike med MegaPOS 1.0 in MegaPOS 2.0?	17

1 Uvod

1.1 NAMEN

Sistem EON MegaPOS je **univerzalni sistem za procesiranje plačil** preko spleta. Skozi standarden vmesnik omogoča trgovcem najbolj varen način za zajemanje plačil na spletu preko številnih sistemov za procesiranje finančnih transakcij oz. plačilnih procesorjev. Trgovca razveže bremena implementacije zahtevnih heterogenih transakcijskih protokolov za vsak plačilni instrument posebej, saj mu skozi enoten in preprost MegaPOS vmesnik nudi **pester in celovit nabor plačilnih sredstev**, začnši z različnimi kreditnimi karticami, debetnimi karticami pa vse do spletnih bank in plačevanja z mobilnim telefonskim aparatom. Zaradi svoje učinkovite zasnove od trgovca ne zahteva hranjenja varnostno-občutljivih podatkov (kot npr. številke kreditnih kartic, pin ali cvd kod), ampak za **celoten življenski cikel varnostno-občutljivih podatkov poskrbi sistem sam**, vključno s korakom zajema podatkov.

1.2 PLAČILNA SREDSTVA

Sistem EON MegaPOS sam ne določa končnega nabora plačilnih sredstev, saj je le ta odvisen od zunanjih procesnih centrov, s katerimi ima trgovec sklenjeno pogodbeno razmerje za storitev procesiranja plačil. MegaPOS v tem trenutku podpira sistem **Activa Banke Koper**, ki se mu v letu 2006 pridružujejo še **eFunds** (bivši Plasis), **Moneta**, **NLB Klik** in **Abanet**.

Plačilni procesor Banke Koper Activa, omogoča procesiranje plačil s kreditnimi karticami EuroCard/MasterCard, VISA in Activa ter z različico debetnih kartic Activa Maestro¹, v pripravi pa je tudi podpora za kreditne kartice Diners in American Express.

1.3 PODPORA

Lastnik blagovnih znamk EON, varno internetno poslovanje, in EON MegaPOS je družba **UPC Telemach** d.o.o. Avtor in razvijalec storitve ter izvajalec tehnične podpore pa je družba **Parsek d.o.o.**

1.4 DELOVANJE

EON MegaPOS je sestavljen iz operativnega transakcijskega sistema **MegaPOS Processor**, dostopnega v obliki spletne storitve (Web Services), in administrativnega spletnega vmesnika **MegaPOS Commerce**, ki omogoča spremljanje in ročno proženje procesiranja ali stornacije plačil. Pri izvajanju plačil se uporablja tehnologija spletnih storitev, uporablja pa se tudi mehanizem prenosa uporabniške seje, ki deluje nad spletnim protokolom HTTP. Procesiranje transakcij poenostavljeno poteka po naslednjem postopku:

- kupec skozi proces spletnega nakupovanja predhodno napolni košarico, pregleda izračun in nato izbere plačilo z ustreznim sredstvom
- trgovec kliče spletno storitev za inicializacijo, pri čemer navede znesek transakcije, valute ter ime in priimek kupca

¹ Velja samo za debetne kartice Activa Maestro z vgrajenim varnostnim čipom.

- rezultat inicializacijskega klica je URL naslov spletne strani za nadaljevanje plačila, kamor trgovec preusmeri kupca
- kupec na spletni strani, kamor je bil preusmerjen, vnese potrebne plačilne podatke in potrdi plačilo
- sistem MegaPOS obvesti trgovca o novem statusu transakcije z asinhronim HTTP klicem
- uporabnika se preusmeri nazaj na spletno trgovino, na stran za prikaz stanja ali izida transakcije
- ... v primeru transakcije tipa nakup se po uspešni inicializaciji transakcije (znesek se avtorizira) izvede še korak dokončnega potrditve (bremenitve) ali preklic transakcije

1.5 VARNOST

Varnost igra v domeni finančnih transakcij odločilno vlogo, zato smo ji tekom razvoja sistema EON MegaPOS posvetili največ pozornosti. Osnovno podlago za zagotavljanje najvišje varnosti predstavlja kriptografski protokol Secure Sockets Layer (SSL 3.0), ki skrbi za enkripcijo popolnoma vseh podatkov z uporabo 128 bitnih asimetričnih ključev. Na vrhu tega je zahtevana na PKI-osnovana avtentifikacija obeh strani, kar pomeni, da tako odjemalec izkaže istovetnost MegaPOS-u kot tudi MegaPOS odjemalcu s svojim certifikatom. Iz certifikata posamezne strani izvirajo tudi asimetrični ključi, potrebni za enkripcijo podatkov.

Za avtentikacijo trgovca se izdelata dva certifikata, s katerima se nasprotni strani identificirata, bodisi aplikacija pri dostopu do spletne storitve, bodisi uporabnik pri dostopu do administrativnega vmesnika. Produkcijski certifikat je namenjen uporabi v produkcijskem okolju, testni pa v testnem okolju tekom razvoja.

Vsi certifikati so izdelani po standardu X.509. Za njihovo uporabo je potrebno nastaviti SSL enkripcijo in omogočiti avtentifikacijo s certifikatom, kar se med programskimi jeziki, razvojnimi orodji in izvršilnimi okolji razlikuje.

1.6 ZAHTEVE

Uvajanje storitve se prične po podpisu pogodbe s ponudnikom storitve EON MegaPOS in sklenitvi ustreznih pogodb s plačilnimi procesorji. Tehnična uvedba od razvijalca zahteva osnovno poznavanje zgoraj omenjenih konceptov in uporabo tehnologije, ki podpira:

- standard spletnih storitev WS-I (nabor je tu karseda širok, podpirajo jih vsi vodilni programski jeziki, kot so C++, C# , Java, PHP, Visual Basic in drugi),
- SSL protokol z dvostransko avtentikacijo preko certifikata

Aplikacija, ki naj bi storitev uporabljala, mora biti povezljiva do ponudnikovega internetnega omrežja na vratih 443 ter biti dosegljiva na poljubnem naslovu, na katerega se preusmeri sejo uporabnika. Podobno mora biti dosegljiva tudi na poljubnem naslovu, na katerega sistem pošilja asinhrona sporočila o izidu transakcije. Podatki prenosnih sporočil morajo biti kodirani v UTF-8 kodu.

2 Spletna plačila v sistemu MegaPOS

2.1 NASTAVITEV TRGOVINE IN PROCESNIH CENTROV

Za uporabo transakcijskega sistema za zajem plačil se trgovcu v MegaPOS-u pripravi novo trgovino in nastavi ustrezna plačilna sredstva oz. plačilne procesorje (**PaymentGateway**), s katerimi ima sklenjeno pogodbo o procesiranju plačil. Vsaka trgovina ima lahko hkrati nastavljenih enega ali več plačilnih procesorjev. Skozi aplikativni spletni vmesnik je mogoče prebrati, kateri procesni centri so na voljo na posamezni trgovini in tudi kakšne so njihove lastnosti:

- tip in ime procesorja, ki ga je potrebno navesti pri inicializacijskem klicu transakcije
- ali vrača odgovor v realnem času ali z zamikom
- ali uporablja asinhrona sporočila za obveščanje o izidu transakcije
- katere vrste transakcij podpira
- ali podpira preklic transakcij posamezne vrste (nakupa, naročila)
- katere plačilne instrumente podpira

2.2 VRSTE TRANSAKCIJ

MegaPOS omogoča procesiranje plačil preko izvajanja transakcij. Podpira dve vrsti transakcij:

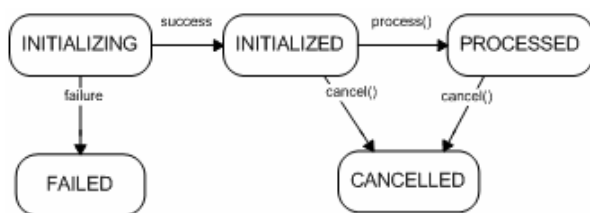
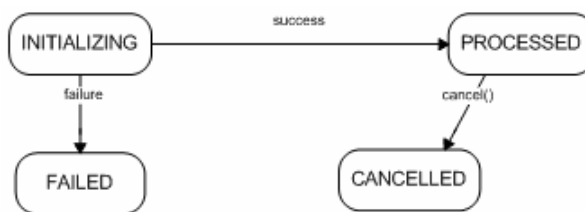
- **ORDER** oz. naročilo
- **PURCHASE** oz. nakup

Naročilo je dvofazna transakcija, saj jo v prvem koraku inicializiramo, v drugem koraku pa bodisi prekličemo ali obdelamo. Ta vrsta transakcij se navadno uporablja za prodajo fizičnih dobrin preko spleta, kjer se v prvem koraku inicializacije od kupca zajame podatke o plačilu in rezervira znesek, v naslednjem koraku pa se znesek v času odpreme še dokončno bremeni, ali pa v primeru preklica naročila stornira. Če trgovec tako želi, lahko zadnji korak v življenjskem ciklu naročila avtomatizira - uporabi aplikativni vmesnik spletne storitve in kliče ustrezno funkcijo za procesiranje ali preklic. V nasprotnem primeru, ko trgovec tega koraka v poslovnem procesu nima avtomatiziranega, ga lahko izvede ročno preko administrativnega spletnega vmesnika **MegaPOS Commerce**.

Nakup je preprostejša transakcija, saj vsebuje le en korak, ki pomeni hkratno rezervacijo in bremenitev sredstev. Ta vrsta transakcije se navadno uporablja za prodajo digitalnih dobrin preko spleta, saj se odprema lahko v primeru nakupa zgodi takoj - npr., elektronski dokument ali dostop do spletnega časopisa se omogoči takoj po zajemu plačila. Nekateri procesni centri puščajo možnost naknadnega preklica nakupa skozi aplikativni vmesnik, vendar je to bolj izjema kot pravilo.

2.3 ŽIVLJENSKI CIKEL TRANSAKCIJ

Vsaka vrsta transakcije ima predhodno znane okvire svojega življenjskega cikla, ki so opisani skozi možna stanja transakcij in prehode med njimi.

Življenski cikel naročila (ORDER):**Življenski cikel transakcije nakupa (PURCHASE):**

INITIALIZING : Inicializacijski zahtev se je uspešno izvršil, transakcija pa je pripravljena na nadaljevanje. V večini primerov to pomeni, da čaka, da se kupca preusmeri na stran plačilnega procesorja, kjer vnese varnostno-občutljive podatke in potrdi plačilo.

INITIALIZED : Znesek naročila se je uspešno rezerviral, transakcija pa je pripravljena na naslednji korak: na preklic (stornacija) ali obdelavo (bremenitev).

PROCESSED : Znesek naročila ali nakupa se je uspešno bremenil. V večini primerov to pomeni, da je transakcija zaključena in je ni možno več preklicati. Določeni procesorji pa dopuščajo možnost preklica tudi iz tega stanja.

CANCELLED: Naročilo ali nakup je bil na zahtevo trgovca preklican in storniran.

FAILED: Inicializacija ni bila uspešna, transakcije ni mogoče nadaljevati.

2.4 OPERACIJE NA TRANSAKCIJAH

Transakcije premikamo po korakih življenjskega cikla oz. stanjih z uporabo naslednjih operacij:

- inicializacija (**initTx**) poskrbi za kreiranje nove transakcije in ustrezno finančno operacijo; t.j. avtorizacijo ali avtorizacijo in bremenitev - odvisno ali gre za naročilo ali za nakup
- procesiranje (**processOrder**) je možno samo v primeru naročila, saj se pri nakupu procesiranje zgodi istočasno z inicializacijo; procesiranje poskrbi za zaključno obdelavo - bremenitev že avtoriziranega zneska naročila
- preklic (**cancelTx**) pomeni popoln preklic transakcije in razbremenitev kupca morebitnih predhodnih avtorizacij ali bremenitev; ali je operacija možna tako na naročilu kot na nakupu, pa je odvisno od procesnega centra

2.5 INICIALIZACIJA TRANSAKCIJE IN ZAJEM KUPČEVIH PODATKOV

MegaPOS pozna dve vrsti procesorjev: tiste ki omogočajo informacijo o izidu v **realnem času** (realtime, npr.: Activa Banke Koper) in tiste, ki informacijo o izidu pošiljajo s **časovnim zamikom** (npr.: NLB Klik). Pri prvih je izid poslan trgovcu v realnem času, prikaže pa se ga tudi kupcu takoj po plačilu. Pri drugih pa se kupca za zajemom podatkov preusmeri na stran z obvestilom o nadaljnjem poteku (npr. "naročilo sprejeto, o uspehu boste naknadno obveščeni po elektronski pošti").

3 Integracija sistema MegaPOS v spletno trgovino

3.1 TOČKE INTEGRACIJE MEGAPOS-A V SPLETNO TRGOVINO

Zaradi same zasnove, ki izhaja iz podpore asinhronim plačilnim procesorjem, je integracija MegaPOSa potrebna na dveh nivojih:

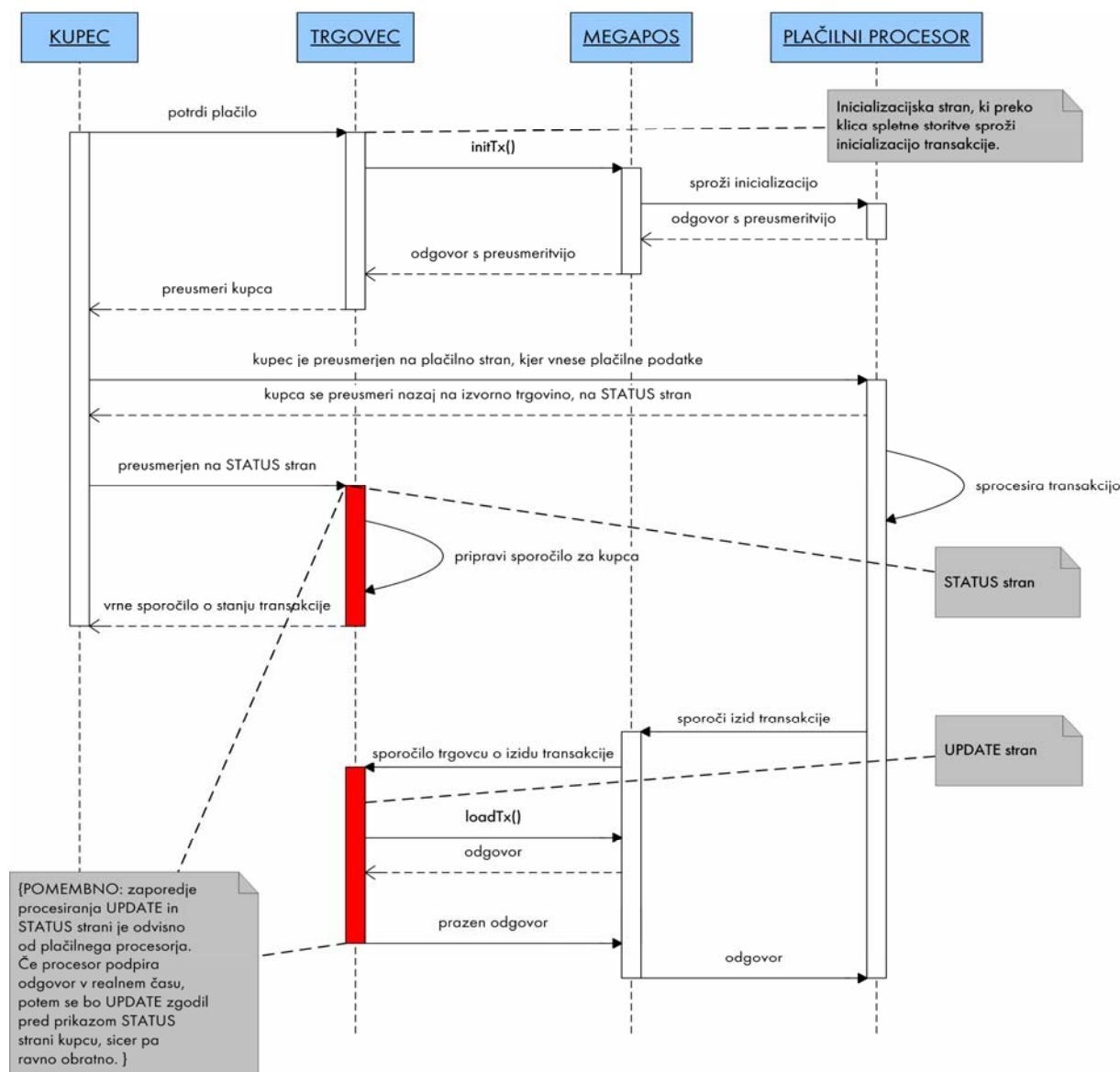
- na nivoju spletne storitve **MegaPOS Processor** po protokolu (SOAP/HTTPS), ki zaobjema vso operativno poslovno logiko za procesiranje plačil
- na nivoju **HTTP** protokola, za sprejemanje asinhronih sporočil s strani MegaPOS-a in za podporo preusmerjanja uporabniških sej

Uporaba spletne storitve se vrši na nižjem nivoju; ta je v poljubnem programskem okolju zelo enostavna, saj klici potekajo podobno preprosto kot pri uporabi lokalnih knjižnic oz. API-jev. Krmiljenje uporabniških sej na nivoju HTTP protokola pa se vrši na višjem nivoju. Od trgovca zahteva integracijo na dveh mestih oz. pripravo dveh dinamičnih spletnih strani:

- **UPDATE** stran je namenjena aplikativnim sporočilom MegaPOS-a za sporočanje izida transakcije. To je signal trgovcu, da je čas za osvežitev stanja transakcije in za morebitno post-procesiranja plačila (pošiljanje emaila, ažuriranje lokalne baze, ipd.).
- **STATUS** stran je dinamična spletna stran, ki predstavlja destinacijo kupca po zaključku plačevanja. Le-ta preveri shranjeno stanje transakcije in pripravi ustrezno sporočilo za prikaz kupcu.

Obe strani se kličeta s parametrom **txId**, ki predstavlja identifikator podatkov. Ostale podatke se v primeru UPDATE strani pridobi preko spletnega strežnika, v primeru STATUS strani pa iz lokalne podaktovne baze.

Če umestimo zgornji dve strani kot integracijski točki v proces inicializacije plačila, dobimo sledečo sliko:



3.2 UPDATE STRAN

Namenjena je aplikativnim sporočilom MegaPOS-a trgovcu za obveščanje o izidu transakcije. Je stran, ki sprejema HTTP POST sporočila s strani MegaPOS-a, ki kot parameter **txId** vsebujejo identifikator transakcije. Ti zahtevki signalizirajo trgovcu, da je potrebna osvežitev stanja transakcije v lokalni podatkovni bazi. Trgovec preko spletne storitve MegaPOS Processor kliče **loadTx()**, da prebere in shrani aktualno stanje transakcije v lokalno podatkovno bazo. Opcijsko, lahko izvede dodatno post-procesiranje plačila, npr.: pošiljanje računa po e-pošti v primeru uspešnega izida (pri plačilnih procesorjih, ki ne nudijo izida transakcije v realnem času).

Spletna stran mora preverjati izvorni IP naslov, s katerega je prišel HTTP POST zahtevek in zagotoviti izvedbo procesiranja samo v primeru, da gre za naslov sistema MegaPOS-a. Dovoljen izvor HTTP POST sporočila s strani MegaPOS-a so vsi naslovi iz naslednjega obsega: 213.143.91.128 - 213.143.91.255.

Spletna stran mora imeti odzivni čas enak ali krajši od 30 sekund.

3.3 STATUS STRAN

Je dinamična spletna stran, ki predstavlja destinacijo kupca po zaključku prvega koraka transakcije. Ob klicu strani se vedno doda parameter **txId**, ki je identifikator transakcije. Nato stran pregleda shranjeno stanje transakcije v lokalni podatkovni bazi trgovca (shranjevanje naj se zgodi po klicu `initTx` in ob procesiranju `UPDATE` strani) in tudi tip plačilnega procesorja, ter pripravi sporočilo za kupca. Status stran iz performančnih razlogov ne sme komunicirati z MegaPOS-om, ampak podatke pridobiva zgolj iz lokalne podatkovne baze.

Procesiranje ustreznih pogojev in priprave sporočil naj bo sledeče:

- V primeru stanja transakcije **INITIALIZING** in procesorja, ki podpira obveščanje o izidu transakcije v realnem času, naj se izpiše stran s sporočilom, da se transakcija procesira in bo končana v nekaj sekundah. V tem primeru je potrebno v HTML sporočilo dodati še meta-značko, da se bo stran v nekaj sekundah osvežila: "`<head><meta http-equiv='refresh' content='10;' /></head>`"
- V primeru stanja transakcije **INITIALIZING** in procesorja, ki obveščanja o izidu v realnem času ne podpira, naj se zgenerira zahvalno sporočilo kupcu, da so bili plačilni podatki uspešno zajeti, o izidu plačila pa bo obveščen naknadno (npr. po elektronski pošti)
- V primeru stanja transakcije **FAILED**, se izpiše sporočilo o neuspešnem plačilu.
- V primeru stanja transakcije **PROCESSED** ali **INITIALIZED** se izpiše, da je bilo plačilo uspešno opravljeno. V primeru transakcije nakupa je to **PROCESSED**, v primeru naročila pa **INITIALIZED**.
- V primeru stanja transakcije **CANCELLED** se izpiše sporočilo, da je bilo plačilo preklicano s strani trgovca.

3.4 UPORABA SPLETNE STORITVE MEGAPOS PROCESSOR

Večina programskih okolij omogoča enostavno uporabo spletnih storitev (Web Services) preko razredov nastavkov (stub classes). T.i. nastavke se zgenerira s pomočjo orodja, specifičnega posameznemu programskemu okolju, kateremu se za vhod nastavi WSDL MegaPOS Processorja.

MegaPOS.wsdl je dokument WSDL v samoopisnem formatu XML, ki vsebuje vse potrebne podatke za uporabo aplikacij preko standarda Web Services. MegaPOS podpira dve vrsti kodiranja SOAP sporočil: `document-wrapped/literal` in `RPC/literal`. Dostopna sta na naslovu:

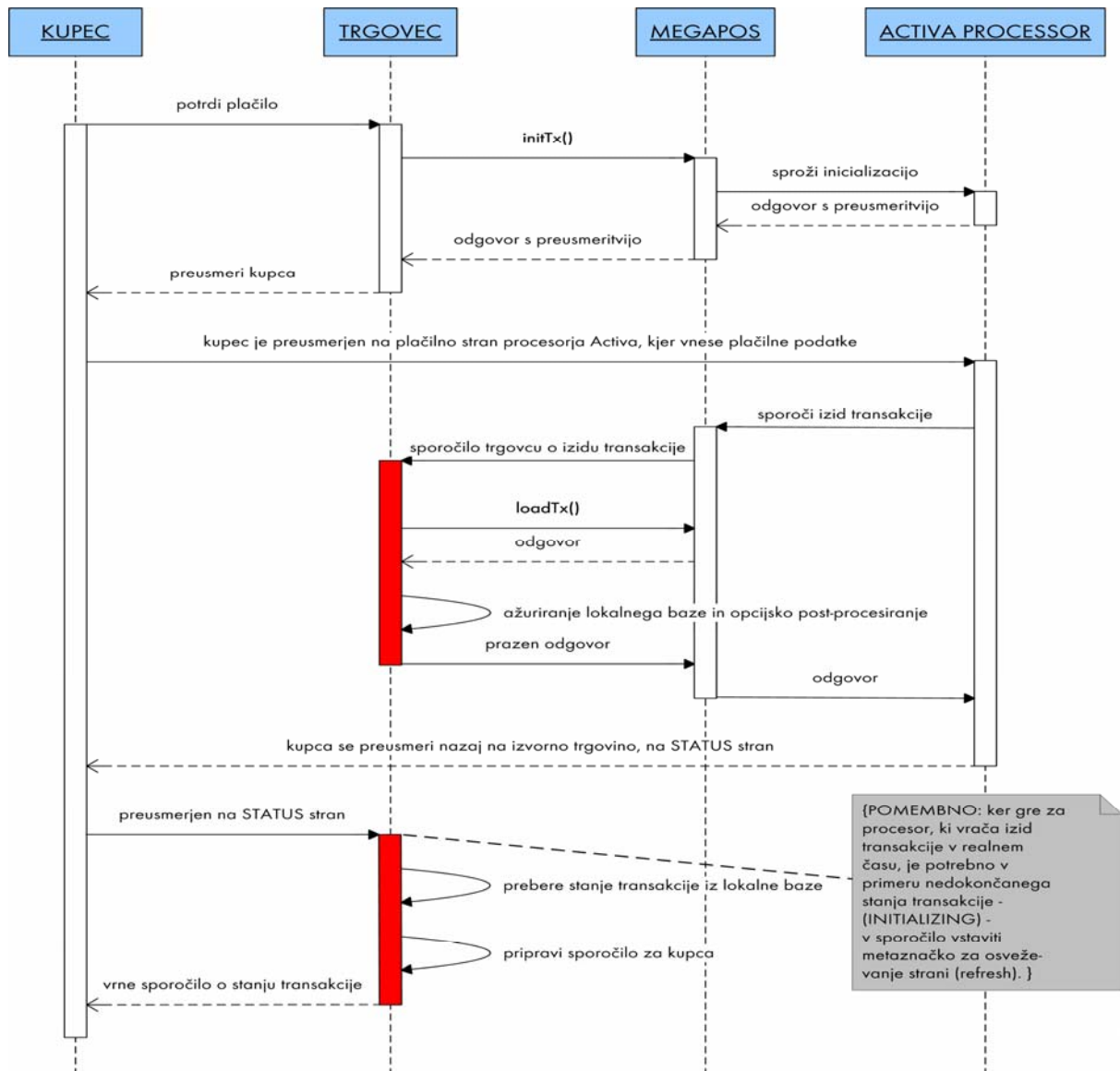
<https://service.megapos.si/ws/megapos2.0/wrapped/processor?wsdl>

<https://service.megapos.si/ws/megapos2.0/rpc/processor?wsdl>

Priporočamo uporabo načina kodiranja `document-wrapped/literal`, saj je edini, ki je kompatibilen s standardom WS-I (Web Services Interoperability). RPC način je ponujen zgolj za okolja, ki prvega ne podpirajo (PHP).

3.5 PROCESOR ACTIVA BANKA KOPER

3.5.1 Potek inicializacije pri procesorju Activa



3.5.2 Aplikativne značilnosti procesorja

- oznaka vrste procesorja: **ACTIVA_PGW**
- ali izide transakcij vrača v realnem času: **da**
- uporablja asinhrona sporočila za obveščanje o izidu transakcije: **da**
- katere vrste transakcij podpira: **PURCHASE, ORDER**
- podpira preklic nakupa: **ne**
- podpira preklic naročila: **da**
- podprti plačilni instrumenti: **CARD_VISA, CARD_MASTERCARD, CARD_ACTIVA_MAESTRO**

4 Definicija spletne storitve MegaPOS Processor

4.1 OPERACIJE

Na spletni storitvi MegaPOS Processor lahko kličete pet metod, ki so opisane v nadaljevanju. Vse vračajo enotno strukturiran objekt **OperationResult** ali izpeljavo le-tega, ki ga sestavljajo atributi **success**, **code** in **msg**. Vse operacija na transakcijah vračajo **TxOperationResult**, ki poleg omenjenih atributov vrača še atribut **txId** ter strukturo **tx** s podatki o transakciji.

4.1.1 initTx()

Inicializira transakcijo. Kliče zahtevek za transakcijo na zahtevani procesni center. Kot rezultat v primeru uspeha vrača URL naslov, kamor mora trgovec preusmeriti kupca za nadaljevanje plačila.

Vhodni parametri: store (string), txId (string), data (struktura TxInitData), pdata (struktura TxProcessData)

4.1.2 processOrder()

Nadaljuje procesiranje transakcije naročila, ki je v stanju INITIALIZED, kar sproži bremenitev predhodno (v inicializaciji) avtoriziranega zneska. Procesiranje je možno klicati s podatki za bremenitev z manjšim zneskom od predhodno avtoriziranega, navedenim v parametru pdata, ki v nasprotnem primeru ni potreben.

Pri transakciji nakupa operacija ni veljavna.

Vhodni parametri: store (string), txId (string), pdata (struktura TxProcessData)

4.1.3 cancelTx()

Popoln preklic transakcije, ki razbremeni kupca predhodne avtorizacije ali bremenitve. V kakšnem stanju mora biti transakcija za izvedbo te operacije, določa tudi sam procesni center, saj vsi ne podpirajo preklica transakcije iz stanja PROCESSED, večina pa jih podpira iz stanja INITIALIZED. Transakcije po uspešnem klicu te operacije ne bo možno več nadaljevati, saj potem konča v zaključenem stanju CANCELLED.

Vhodni parametri: store (string), txId (string)

4.1.4 loadTx()

Naloži transakcijo. Je metoda, s katero lahko preverimo stanje transakcije.

Vhodni parametri: store (string), txId (string)

4.1.5 listPaymentGateways()

Vrne vse procesne centre na dani trgovini in plačilne instrumente, ki jih le-ti podpirajo.

Vhodni parametri: store (string)

Rezultat: ListPaymentGatewaysResult, ki poleg ostalih podatkov vsebuje še niz procesnih centrov.

4.2 VHODNI PARAMETRI

Seznam vseh vhodnih parametrov z njihovimi podatkovnimi tipi:

4.2.1 store (String)

Ime trgovine, ki je tudi njen enolični identifikator.

4.2.2 txId (String)

Identifikator transakcije, ki jo določi trgovec ob inicializaciji transakcije. Identifikator mora enolično določati transakcijo v dani trgovini. Identifikator je lahko sestavljen iz alfanumeričnih znakov, podčrtajev in vezajev, ter ne sme presegati dolžine 50-ih znakov.

4.2.3 data (TxInitData)

Podatki o transakciji. Vsebuje sledeča polja:

4.2.3.1 statusURL (String)

Naslov, na katerega se preusmeri kupca zatem, ko opravi nakup.

4.2.3.2 updateURL (String)

Naslov, na katerem aplikacija trgovine posluša za asinhrona obvestila o spremembi stanja transakcije. MegaPOS na ta naslov pošlje HTTP POST zahteve, ki dajo trgovcu znak, da je čas za posodobitev statusa transakcije.

4.2.3.3 paymentGateway (String)

Procesni center, preko katerega se bo transakcija procesirala. Trenutno podprt je ACTIVA_PGW (Banka Koper Activa).

4.2.3.4 testMode (Boolean)

Pove, ali je transakcija testna ali produkcijska. Možni vrednosti sta true in false.

4.2.3.5 txType (String)

Vrsta transakcije. Veljavni sta dve vrednosti, 'PURCHASE' in 'ORDER', ki določata, ali gre za nakup ali naročilo. Nakup je namenjen digitalnim dobrinam, saj znesek istočasno avtorizira in bremeni. Naročilo je namenjeno fizičnim dobrinam, saj znesek samo avtorizira, ko pa trgovec blago odpošlje kliče še processOrder, da se znesek tudi bremeni.

4.2.3.6 custName (String)

Ime kupca.

4.2.3.7 custSurname (String)

Priimek kupca.

4.2.3.8 addInfo (String)

Morebitni dodatni podatki o kupcu, transakciji, artiklu, ki ga kupuje, ... Neobvezno polje.

4.2.4 pdata (ProcessData)

Podatki o znesku in valuti, sestavljeni iz atributov:

4.2.4.1 amount (Decimal)

Znesek transakcije, določen z dvema decimalkama natančno. V primeru zneska z več decimalkami bo znesek zaokrožen na dve decimalni mesti.

4.2.4.2 currency (String)

Valuta. Trenutno je podprta samo valuta SIT.

4.3 IZHODNI PARAMETRI

Seznam vseh izhodnih parametrov in njihovih podatkovnih tipov:

4.3.1 success (Boolean)

Uspešnost transakcije. Možni vrednosti sta true in false.

4.3.2 code (String)

Šifra rezultata operacije.

4.3.3 msg (String)

Opis oz. razlaga šifre odgovora.

4.3.4 gateways (ListPaymentGatewaysResult)

Struktura, ki vsebuje podatke o vseh procesnih centrih in podprtih plačilnih instrumentih v dani trgovini.

4.3.5 tx (Tx)

Struktura, ki vsebuje vse inicializacijske podatke transakcije in podatke o stanju transakcije:

4.3.5.1 txId (Store)

Identifikator transakcije.

4.3.5.2 store (String)

Identifikator oz. ime trgovine.

4.3.5.3 initData (TxInitData)

Že opisan med vhodnimi parametri.

4.3.5.4 stateData (TxStateData)

Struktura s podatki o aktualnem stanju transakcije, ki jo sestavljajo atributi:

type (String)	Vrsta stanja. V odvisnosti od vrste transakcije, lahko nastopajo naslednja stanja: INITIALIZING, INITIALIZED, PROCESSED, FAILED, CANCELLED
timestamp (Date)	Datum nastopa aktualnega stanja.
processData (TxProcessData)	Struktura, ki nosi informacijo o znesku in valuti, in je že opisana med vhodnimi parametri.
resultType (String)	Vrsta rezultata, ki je lahko INFO, ERROR ali NONE, v primeru stanja INITIALIZING pa REDIRECT - tedaj result vsebuje URL naslov za nadaljevanje plačila.
result (String)	Vrednost, ki je napolnjena glede na polje resultType. V primeru REDIRECT resultType, je tukaj URL naslov za preusmeritev kupca.

4.4 ODGOVORNE OZNAKE IN SPOROČILA

Pari odgovorne oznake in sporočila so prisotni v rezultatih večine operacij na storitvi MegaPOS Processor: v `initTx()`, `processOrder()`, `cancelOrder()` in tudi `listPaymentGateways()`. V osnovi poznamo dve vrsti oznak: negativna števila, ki signalizirajo, da je med izvajanjem prišlo do napake; in pozitivna števila oz ničla, ki sporoča, da je bila operacija uspešno izvedena. Oznake so tisti del rezultata, ki se ga aplikativno upošteva v programski kodi odjemalca. Sporočila pa so tisti del rezultata, ki je dinamičen in kot tak ni namenjen aplikativnemu procesiranju, temveč zgolj bolj podrobni informiranosti o napaki za lažje razhroščevanje.

Poznamo sledeče odgovorne oznake:

- **-2000** napaka pri izvajanju transakcije
- **-2100** napaka načina transakcije (testni ali produkcijski)
- **-2200** napaka stanja transakcije
- **-2300** napaka vhodnega parametra pri izvajanju transakcije
- **-3000** napaka plačilnega procesorja
- **-5000** napaka trgovine
- **-5100** trgovina ni dostopna
- **-7000** varnostna napaka
- **-7100** odjemalec ni avtenticiran
- **-7200** odjemalec ni avtoriziran

5 Implementacija

5.1 JAVA

Programsko okolje Java podpira najnovejše standarde spletnih storitev in nudi širok nabor SOAP knjižnic ter orodij (npr. Axis, Axis-2, X-Fire, ...). Zaradi dobre podprtosti priporočamo uporabo optimalnejšega in bolj standardnega načina kodiranja SOAP sporočil `document/literal`. V nadaljevanju je opisana vzpostavitev in referenčne implementacije spletne trgovine v Java okolju, ki uporablja Axis-1.4 SOAP knjižnico.

5.1.1 Vzpostavitev referenčne implementacije spletne trgovine

Vzpostavitev delujoče referenčne implementacije, ki jo lahko uporabljate tudi za testiranje, dosežete v sledečih korakih:

1. Nastavitev omrežja: Poskrbeti morate, da je računalnik, ki bo poganjal spletno trgovino, dosegljiv iz zunanega omrežja. Če se računalnik priklaplja v internet skozi deljeno povezavo preko routerja, potem je potrebno nastaviti *Port Forwarding*, za preusmerjanje zunanjih zahtevkov na računalnik spletne trgovine. Rezultat nastavitve naj bi bil zunanji naslov (URL ali IP naslov) in vrata (port), na katerih bo dosegljiva spletna aplikacija za asinhrona sporočila s strani MegaPOS-a.

2. Namestitev prostodostopnega *servlet container*-ja Apache Tomcat (<http://tomcat.apache.org>): V konfiguracijski datoteki `tomcat/conf/server.xml` locirajte element *Connector*, na katerem ustrezno nastavite *address* in *port* atribut. Ta dva atributa določata naslov in vrata mrežnega vmesnika, ki ga bo zasedla spletna trgovina. Privzete nastavitve zasedejo vse lokalne mrežne vmesnike na vratih 8080.

3. Namestitev referenčne implementacije spletne trgovine: Razkrčite `mgps-rsi.zip` v mapo `tomcat/webapps`. Odprite datoteko `tomcat/webapps/mgps-rsi/WEB-INF/web.xml`. Tukaj lahko nastavljate naslov MegaPOS strežnika, kamor se spletna trgovina povezuje; naslov trgovine na MegaPOS-u, ki jo spletna trgovina uporablja; `txUpdateURL` in `txStatusURL` ter nastavitve certifikatov za avtentikacijo MegaPOS-u in obratno.

`TxUpdateURL` nastavite na naslov oblike `http://trgovec:vrata/mgps-rsi/updateTx`, kjer vstavite zunanji naslov in vrata za dostop do spletne trgovine. `TxStatusURL` je namenjen za prikaz stanja transakcije kupcu. Njegovo vrednost nastavite na naslov oblike `http://trgovec:vrata/mgps-rsi/txStatus.jsp`, kjer vstavite zunanji naslov in vrata za dostop do spletne trgovine.

V parametra `keyStore` in `keyStorePassword` ustrezne vstavite absolutno pot na vaš certifikat (formata PKCS12), ki se bo uporabil za dostop do MegaPOS strežnika in geslom za dostop do certifikata.

V parameter `trustStore` vstavite absolutno pot do `tomcat/webapps/mgps-rsi/WEB-INF/crt/service.megapos.si.jks`, kjer se nahaja javni del certifikata strežnika MegaPOS. V parameter `trustStorePassword` vpišite geslo 'megapos'.

4. Poženite *servlet container* Apache Tomcat: V mapi `tomcat/bin` poženite `startup.bat` oz. `startup.sh`.

5. Stestirajte spletno trgovino: Pojdite na naslov oblike `http://trgovec:vrata/mgps-rsi`, kjer navedete prej uporabljene nastavitve in stestirajte spletno trgovino.

5.1.2 Vsebina mgps-rsi.zip spletne trgovine

mgps-rsi/	celotna spletna aplikacija
./*.jsp, *.html, *.css	dinamične spletne strani inicializacijo, preklic in procesiranje transakcije; začetna stran ter datoteka s slogi
./WEB-INF/web.xml	splošne nastavitve spletne aplikacije in nastavitve spletne trgovine
./WEB-INF/classes	java razredi spletne aplikacije
./WEB-INF/classes/com/parsek/megapos/client/*.class	generirani razredi nastavki (t.i. stub classes) s pomočjo Axis-1.4 SOAP knjižnice
./WEB-INF/classes/com/parsek/megapos/client/rsi	servlet za sprejem MegaPOS-ovih asinhronih obvestil ter servlet za inicializacijo transakcije
./WEB-INF/classes/log4j.xml	nastavitve logiranja
./WEB-INF/cert	mapa s certifikati za dostop do MegaPOS-a in javnim delom certifikata strežnika MegaPOS
./WEB-INF/lib	mapa s javanskimi knjižnicami (jar)

5.2 PHP

Trenutno je za podporo spletnih storitev v PHP na voljo nekaj tehnologij (pear::SOAP, nuSOAP, ...). PHP ima standardiziran način razširljivosti v okviru ogrodja PEAR, ki ponuja vrsto komponent, med katerimi je spletnim storitvam namenjen pear::SOAP. Zaradi njegove nezapletenosti ga priporočamo tudi za podporo MegaPOS. Priporočena verzija je 0.8.1. Z njo je narejen tudi testni primer, ki vsebuje tudi vse potrebne PEAR datoteke (ni potreba instalacija).

5.2.1 Certifikati

pear::SOAP uporablja za komunikacijo z omrežjem komponento cURL, ki podpira vse zahtevnejše protokole, med katerimi je tudi vzpostavitev povezave prek SSL z avtentikacijo s certifikatom, ki jo zahteva MegaPOS.

V php.ini je potrebno zaradi uporabe cURL odkomentirati vrstico "extension=php_curl.dll".

MegaPOS_stub.php je t.i. "stub" razred, na katerem se kliče metode. Stub razred lahko sicer pear::SOAP ustvari sam s pomočjo dokumenta WSDL. Ta funkcionalnost je v datoteki getstub.php testnega primera.

Zaradi potrebne nastavitve certifikata po inicializaciji smo zgeneriranemu stubu dodali še 3 spremenljivke za pot do certifikatov in ključa pri klicu vsake metode.

Base.php je del pear::SOAP paketa. Ker je pri klicu metod parameter amount tipa long, je potrebno v originalnem Base.php (\$PHP_HOME/PEAR/SOAP/Base.php) zaradi neustrezne konverzije tipa long v int vrstico 617 (odvisno od pear::SOAP verzije):

```
616:     case 'long' :
```

```
617:    $type = 'int';
```

spremeniti v:

```
617:    $type = 'long';
```

V priloženem primeru je datoteka že popravljena.

6 Administrativni spletni vmesnik MegaPOS Commerce

MegaPOS Commerce je spletni vmesnik, kjer lahko pregledujete seznam transakcij. Dostopen je na naslovu <https://service.megaapos.si/transactions?store=storename> Storename je MegaPOS oznaka oz. ime vaše trgovine. Za dostop do MegaPOS Commerce morate v brskalnik uvoziti certifikat trgovine (format PKCS12), do katere želite dostopati.

MegaPOS Commerce vam omogoča pregled transakcij in pregled prometa - batchev. Batchi so skupine transakcij, ki so pošiljajo na ustrezen procesni center v obdelavo. Uporabnost takšnega pogleda je pri ugotavljanju, katere transakcije so zajete v posameznem nakazilu banke. Razpoložljivost te funkcionalnosti je odvisna od posameznega plačilnega procesorja. Pri plačilnem procesorju Banke Koper se batchi ne pošiljajo, saj se celotna transakcija opravi pri njih.

7 Pogosto zastavljena vprašanja

7.1 KAKŠNE SO RAZLIKE MED MEGAPOS 1.0 IN MEGAPOS 2.0?

- Kupec vnaša občutljive podatke na strani procesnega centra – trgovec in MegaPOS jih ne zajemata več.
- klica authorize in authorizeAndSettle sta zdaj združena v initTx, nastavita pa se z txType, ki je v prvem primeru ORDER, v drugem pa PURCHASE
- MegaPOS 2.0 podpira tudi plačilne procesorje, ki uporabljajo asinhrono obveščanje (spletne banke, moneta, NLB Klik, Aactiva Banka Koper, ...)
- Znesek se ne podaja več s stotini, ampak se morebitne stotine navede za decimalno vejico.