



gopay.cz
platby

Platební brána GoPay

GoPay

Integrační manuál
verze 2.3



Obsah

| | |
|---|----|
| 1 Definice..... | 4 |
| 2 Platební brána..... | 5 |
| 3 Postup integrace..... | 7 |
| 3.1 Obchodník: poptávka služeb..... | 7 |
| 3.2 GoPay: vytvoření testovacího prostředí..... | 7 |
| 3.3 Obchodník: provedení integrace..... | 7 |
| 3.4 GoPay: ověření integrace..... | 8 |
| 3.5 GoPay: dokončení smlouvy, předání provozních údajů..... | 8 |
| 4 Úplná integrace..... | 9 |
| 4.1 Princip komunikace..... | 9 |
| 4.2 Vytvoření platby..... | 10 |
| 4.3 Přesměrování na platební bránu GoPay..... | 11 |
| 4.4 Ověření stavu platby..... | 11 |
| 4.5 URL platební brány GoPay..... | 12 |
| 5 Před-autorizovaná platba..... | 13 |
| 5.1 Založení platby..... | 13 |
| 5.2 Dokončení platby..... | 13 |
| 5.3 Zrušení před-autorizace..... | 14 |
| 5.4 Procesní diagramy..... | 15 |
| 6 Opakovaná platba..... | 16 |
| 6.1 Založení platby..... | 16 |
| 6.2 Opakování platby..... | 16 |
| 6.3 Zrušení opakování platby..... | 17 |
| 6.4 Procesní diagram..... | 19 |
| 7 GoPay služby..... | 21 |
| 7.1 Služby..... | 21 |
| 7.1.1 Vytvoření platby..... | 21 |
| 7.1.2 Stav platby..... | 21 |
| 7.1.3 Provedení opakované platby..... | 21 |
| 7.1.4 Zrušení opakování platby..... | 21 |
| 7.1.5 Provedení před-autorizované platby..... | 22 |
| 7.1.6 Zrušení před-autorizované platby..... | 22 |
| 7.1.7 Založení uživatele..... | 22 |
| 7.2 GoPayWS..... | 23 |
| 7.2.1 Definice WS..... | 23 |
| 7.3 Popis volání..... | 23 |
| 7.3.1 Použití WS..... | 24 |
| 8 GoPay notifikace..... | 25 |
| 8.1 Doručení notifikace..... | 26 |
| 8.2 HTTP notifikace..... | 26 |



| | |
|--|----|
| 8.3 XML-RPC notifikace..... | 27 |
| 8.4 Email notifikace..... | 28 |
| 8.5 SMS notifikace..... | 29 |
| 9 Vyúčtování..... | 30 |
| 10 Výpis pohybů z obchodního účtu..... | 31 |
| 11 Založení uživatele..... | 32 |
| 12 Testování plateb..... | 33 |
| 12.1 GoPay platební brána..... | 33 |
| 12.1.1 Platební metoda GoPay..... | 33 |
| 12.1.2 Platební metoda superCASH..... | 34 |
| 12.1.3 Opakovaná platba..... | 34 |
| 12.1.4 Platební metoda PremiumSMS..... | 34 |
| 12.2 GoPay Monitor..... | 35 |
| 13 Komunikační elementy..... | 36 |
| 13.1 Platební příkaz (paymentCommand)..... | 36 |
| 13.2 Opakovaní platby (recurrenceRequest)..... | 38 |
| 13.3 Platební session (paymentSession)..... | 39 |
| 13.4 Identita platby (paymentIdentity)..... | 40 |
| 13.5 Stav platby (paymentStatus)..... | 41 |
| 13.6 Výpis z účtu (eStatementRequest)..... | 43 |
| 13.7 Založení uživatele (buyerCreate)..... | 44 |
| 13.8 Výsledek vytvoření uživatele (buyerCreateResult)..... | 45 |
| 13.9 Výsledek volání (paymentResult)..... | 46 |
| 13.10 Kódy platebních metod..... | 47 |
| 13.11 Stav platby..... | 48 |
| 13.12 Doplnující stavy..... | 49 |
| 13.13 Doba životnosti platby..... | 50 |
| 13.14 Kód země..... | 51 |
| 13.15 Číselník měn..... | 52 |
| 13.16 Kód jazyka..... | 53 |
| 13.17 Podpis komunikačních elementů..... | 54 |
| 13.17.1 Vzorové implementace..... | 54 |
| 14 Vzorové implementace..... | 55 |
| 15 Kontaktní údaje..... | 56 |
| 16 Přílohy..... | 57 |
| 16.1 PHP..... | 57 |
| 16.2 Java API..... | 59 |
| 16.3 Python API..... | 60 |
| 16.4 ASP.NET..... | 62 |
| 16.5 Ruby on Rails API..... | 63 |

1 Definice

| | |
|--------------------------|---|
| Obchodník | Provozovatel E-shopu. |
| Obchodní účet | Účet obchodníka v rámci GoPay, kde jsou evidovány platby z jednotlivých E-shopů. |
| E-shop | E-shop nebo internetová aplikace provozovaná obchodníkem. |
| Zákazník | Osoba využívající služeb obchodníka. |
| GoPay peněženka | Internetová peněženka pro rychlé každodenní platby, posílání nebo přijímání peněz po internetu. |
| Uživatel GoPay | Uživatel GoPay peněženky identifikovatelný pomocí Go přezdívky nebo E-mailové adresy. |
| Platební brána | Platební brána GoPay umožňuje obchodníkovi nebo uživateli GoPay peněženky přijímat platby na obchodní účet z níže popsaných platebních metod. |
| GoPay monitor | Prostředí obchodníka pro sledování plateb na vytvořených obchodních účtech, E-shopech. |
| GoPayWS | Webová služba umožňující přístup k systémovým funkcionalitám platební brány. |
| GoPay notifikace | Zajišťuje doručení informace o provedení platby obchodníkovi. |
| Platba (payment session) | Identifikace platby v rámci GoPay. Každá platba na platební bráně má přiřazeno jedinečné číslo <i>paymentSessionId</i> . |
| Registrační formulář | Součástí smlouvy mezi GoPay a obchodníkem. Definuje poplatky, nastavení E-shopů a vyúčtovací politiku. |



2 Platební brána

V rámci jediné smlouvy a jediné integrace získá každý obchodník hlavní platební metody na českém internetu za velmi výhodných podmínek. Hlavní předností je úspora nákladů v rámci jednotné správy financí.

Platební brána GoPay umožňuje obchodníkovi ve svém profilu evidovat více internetových obchodů na různých obchodních účtech. Přijaté platby jsou zaznamenány na obchodní účet odpovídající danému obchodu.

Převod prostředků na bankovní účet obchodníka je prováděn ve formě vyúčtování. Vyúčtování je prováděno pravidelně za určité období, nebo manuálně za aktuálně dostupné období. Parametry vyúčtování jsou blíže popsány ve smluvních podmínkách a v Registračním formuláři.

Obchodní účty lze napojit na různé bankovní účty a tím rozdělit příjmy z jednotlivých činností.

Fungování platební brány GoPay si můžete vyzkoušet na vzorovém E-shopu www.goshop.cz.

Platební brána GoPay je základním rozhraním pro provádění plateb prostřednictvím systému GoPay. Obchodník může jejím prostřednictvím přijímat platby následujícími platebními metodami:



| Platební metody | Typ platby | Provedení platby | Poskytovatel služby | Automatizované vrácení plateb | Před- autorizace | Opakované platby | Měna |
|----------------------------|---|------------------|--|-------------------------------|------------------|------------------|---------|
| VISA, MasterCard | E-commerce 3-D Secure – platba kartou prostřednictvím GoPay peněženky | on-line | UniCredit Bank- Global payments, 3D Secure | A | A | A | EUR,CZK |
| VISA, MasterCard | E-commerce 3-D Secure – platba kartou prostřednictvím GoPay peněženky | on-line | Česká spořitelna, a.s., 3D Secure | A | N | A | CZK |
| GoPay peněženka | elektronická peněženka | on-line | GoPay.cz | A | N | N | EUR,CZK |
| Mojeplatba | internetové bankovníctví | on-line | Komerční Banka a.s. | A | N | N | CZK |
| ePlatby | internetové bankovníctví | on-line | Raiffeisenbank a.s. | A | N | N | CZK |
| mPeníze | internetové bankovníctví | on-line | BRE Bank S.A. | A | N | N | CZK |
| Fio banka | internetové bankovníctví | on-line | Fio banka, a.s. | A | N | N | CZK |
| Sporopay | internetové bankovníctví | on-line | Slovenská sporiteľňa, a. s | A | N | N | EUR |
| Uniplatba | internetové bankovníctví | on-line | UniCredit Bank Slovakia a. s. | A | N | N | EUR |
| Bankovní převod - expresní | bankovní převod | offline | Bankovní účty GOPAY s.r.o. | A | N | N | EUR,CZK |
| PremiumSMS | SMS platba | on-line | XPay s.r.o. | N | N | N | EUR,CZK |
| M-platba | Platba mobilem | on-line | Advanced Telecom Services, s.r.o. | N | N | N | CZK |
| superCASH | terminály Česká pošta a.s., SAZKA a.s. | offline | Manum s.r.o. | N | N | N | CZK |

3 Postup integrace

Integrace GoPay do prostředí E-shopu nebo internetové aplikace obchodníka je rozdělena do následujících etap.

3.1 Obchodník: poptávka služeb

Na www stránkách GoPay vyplní obchodník poptávkový formulář (<https://www.gopay.cz/partnerstvi>). Obchodní oddělení GoPay kontaktuje obchodníka nejpozději do 3 pracovních dnů.

3.2 GoPay: vytvoření testovacího prostředí

V případě, že se obchodník rozhodne využít nabídku GoPay, je třeba vyplnit a zaslat registrační formulář, na základě kterého bude vytvořeno testovací prostředí. Údaje pro nastavení testovacího prostředí jsou předávány obchodníkovi E-mailem, popřípadě SMS zprávou.

3.3 Obchodník: provedení integrace

V této fázi provádí obchodník integraci GoPay do prostředí svého E-shopu. Pro implementaci může zvolit:

- samostatná implementace
Pro samostatnou implementaci je připraveno API pro programová prostředí PHP a Java.
- GoPay modul
Dále je možné použít platební moduly GoPay pro open-source řešení jako je PrestaShop, Magento, Joomla, ZenCart (viz <https://www.gopay.cz/jak-funguje-gopay/platebni-moduly>).
Nebo lze využít služeb některého z našich certifikovaných dodavatelů (viz <https://www.gopay.cz/o-nas/nasi-partneri/platebni-reseni>).

Pro ověření implementace je k dispozici testovací prostředí GoPay, kreditovaný uživatel GoPay a testovací platební brána. Více o testování integrace naleznete v kapitole 12 - Testování plateb.

Před samotnou realizací je nutné předat na integrace@gopay.cz údaje viz níže uvedené úkoly.

| Úkol | Splněno |
|--|---------|
| Obchodník předal IP adresy pro přístup k testovacímu prostředí GoPay peněženka, GoPay Monitor. | ANO/NE |
| Obchodník předal URL pro doručení HTTP notifikace. | ANO/NE |

3.4 GoPay: ověření integrace

Jakmile je integrace na straně obchodníka dokončena, potom provede *Tester* (obchodník) níže uvedené úkoly a kontaktuje GoPay na adrese integrace@gopay.cz pro ověření integrace.

Po úspěšném provedení kontroly je integrace předána k ověření úplnosti smluv.

| Úkol | Splněno |
|--|---------|
| Tester vytvořil z prostředí E-shopu platbu na testovacím prostředí GoPay. | ANO/NE |
| Tester vytvořil z E-shopu platbu a následně provedl její úhradu on-line metodou. E-shop korektně zpracoval <i>successURL</i> . | ANO/NE |
| Tester vytvořil z E-shopu platbu a následně provedl její zrušení. E-shop korektně zpracoval <i>failedURL</i> . | ANO/NE |
| Tester vytvořil z E-shopu platbu a následně vybral platbu formou superCASH (opište si číslo superCASH - k pozdější úhradě). Dále se tester vrátil zpět na E-shop. E-shop korektně zpracoval <i>successURL</i> aniž by byla platba dokončena. | ANO/NE |
| Tester provede uhrazení superCASH platby z rozhraní viz 12.1.2. E-shop korektně zpracuje notifikaci o provedení platby. | ANO/NE |
| E-shop předal GoPay logo k prezentaci na platební bráně. | ANO/NE |

3.5 GoPay: dokončení smlouvy, předání provozních údajů

V této fázi je prováděna kontrola úplnosti smluv s GoPay. Za předpokladu úspěšně provedené kontroly, jsou obchodníkovi předány provozní údaje.

Provozní údaje obdržíte emailem a SMS zprávou na kontaktní údaje uvedené ve smlouvě. Technický kontakt obdrží provozní *GoID* E-shopu a *šifrovací klíč* (secure key). Obchodní kontakt obdrží přístupové údaje do prostředí GoPay Monitoru.

V tento okamžik je nutné nakonfigurovat E-shop s provozními údaji (GoID, secure key).

UPOZORNĚNÍ: Nezapomeňte změnit URL na provozní adresy GoPay.

Integrační proces je zakončen vytvořením platby na provozním prostředí a její kontrolou. Po provedení níže uvedených úkolů nás kontaktujte na integrace@gopay.cz.

| Úkol | Splněno |
|--|---------|
| Tester vytvořil z prostředí E-shopu platbu na provozním prostředí GoPay. | ANO/NE |
| E-shop prezentuje platební systém GoPay dle požadavků ve smlouvě. | ANO/NE |

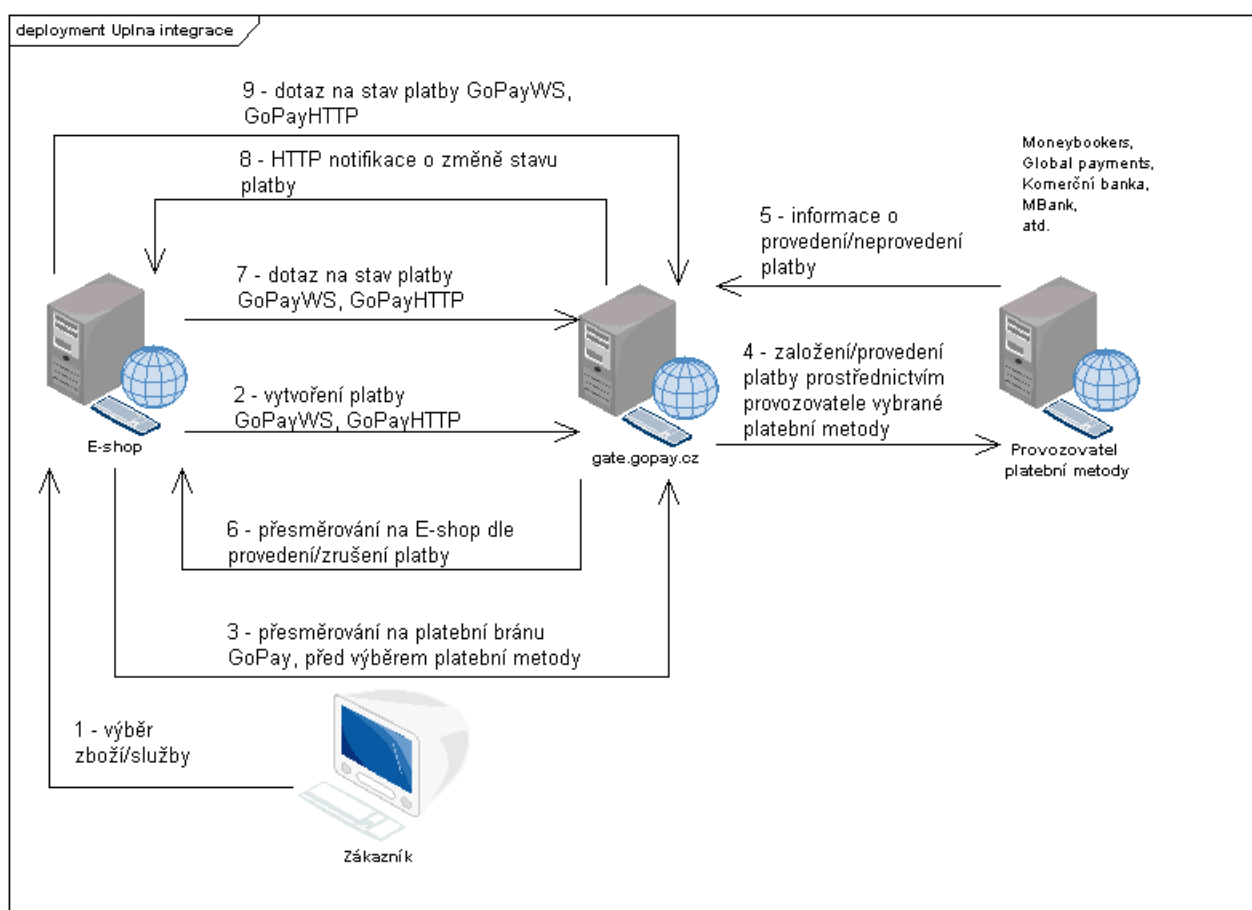
V případě, že nekontaktujete GoPay do 7 dnů, od předání provozních údajů, dojde k pozastavení E-shopu. Žádost o opětovnou aktivaci a prodloužení finálního testu zasílejte na integrace@gopay.cz.

4 Úplná integrace

Úplná integrace je určena E-shopy nebo webová řešení, která vyžadují automatizované zpracování provedení platby.

4.1 Princip komunikace

Platba je založena ze serverové strany E-shopu pomocí GoPay služeb. E-shop získá identifikátor platby *paymentSessionId* před samotným provedením/zrušením platby. Tímto je vytvořena unikátní vazba mezi objednávkou a platbou. Po dokončení platby je provedena on-line kontrola platby prostřednictvím GoPay služeb (viz 7 - GoPay služby).





Průběh platby s využitím úplné integrace je zobrazen níže:

- 1) Zákazník se pohybuje na E-shopu. Provádí výběr zboží/služby (plnění košíku).
- 2) E-shop – vytvoření Platby

Vytvoření platby je dostupné prostřednictvím GoPay služby (viz 4.2 - Vytvoření platby).

- 3) Přesměrování na platební bránu GoPay

Následuje přesměrování na URL platební brány GoPay (viz 4.3 - Přesměrování na platební bránu GoPay).

- 4) Výběr platební metody

Na platební bráně provede zákazník výběr platební metody. Nabízené platební metody jsou ovlivněny parametry předávanými při vytvoření platby.

O dalším postupu pro dokončení platby je zákazník informován na platební bráně (přechod na platební rozhraní vybrané platební metody, odeslání PremiumSMS, platba superCASH kupónu, atp.).

- 5,6,7) Provedení/neprovedení platby, ověření platby

V případě online plateb (viz 2 - Platební brána) dostává platební brána informaci o provedení platby okamžitě a zákazník je přesměrován dle výsledku na *successURL* nebo *failedURL*.

Pro případ plateb formou offline metod může zákazník přejít na stranu E-shopu prostřednictvím *successURL* aniž by byla platba dokončena.

Pro dokončení, resp. pro určení stavu platby je nutné provést ověření stavu platby (viz postup popsáný v 4.4 - Ověření stavu platby).

Po přesměrování zpět na E-shop je nutné, aby stav platby byl zákazníkovi vhodně prezentován.

- 8, 9) HTTP notifikace o změně stavu platby

HTTP notifikace zajišťuje bezpečné doručení informace o změně stavu platby. Doručení HTTP notifikace je provedeno do 15 vteřin od provedení/zrušení/vypršení životnosti platby.

Zpracování notifikace je blíže popsáno v kapitole 4.4 - Ověření stavu platby. Zpracování HTTP notifikace je povinné.

Notifikaci lze doručit dalšími kanály (viz 8 - GoPay notifikace).

4.2 Vytvoření platby

Vytvoření platby je dostupné prostřednictvím GoPay služby, (viz 7.1.1 - Vytvoření platby), předávané parametry odpovídají elementu 13.1 - Platební příkaz (*paymentCommand*). Jako výsledek je získán identifikátor platby v systému GoPay (*paymentSessionId*). Na straně E-shopu je nutné vytvořit relaci mezi objednávkou a platbou *paymentSessionId*.

Založení opakované platby je blíže popsáno v kapitole 6 Opakovaná platba.

V závislosti na platební metodě GoPay požaduje po zákazníkovi množinu osobních údajů. Tyto údaje

GoPay předává provozovateli platební metody. Pro usnadnění ovládání platební brány lze zákaznické údaje předávat jakou součástí platebního příkazu.

Od okamžiku vytvoření je možné zjišťovat stav platby pomocí GoPay služeb (viz 7.1.2 - Stav platby). Životní cyklus platby je popsán v 13.11 - Stav platby.

Vzorové implementace vytvoření platby naleznete v příložených zdrojových kódech.

| Prostředí | implementace založení platby |
|---------------|--|
| PHP | viz 16.1, vytvoření platby naleznete ve skriptu <i>payment.php</i> . |
| Java | viz 16.2, vytvoření platby naleznete ve třídě <i>SimplePayment</i> , metoda <i>pay</i> . |
| ASP.NET | viz 16.4, vytvoření platby naleznete ve třídě <i>Payment</i> metoda <i>Pay</i> . |
| Python | viz 16.3, vytvoření platby naleznete ve skriptu <i>SimplePayment.py</i> . |
| Ruby on Rails | viz 16.5, vytvoření platby naleznete ve skriptu <i>SimplePayment.rb</i> . |

4.3 Přesměrování na platební bránu GoPay

Přesměrování obsahuje parametry (viz 13.3 - Platební session (*paymentSession*)). Jednotlivé parametry (vyjma parametru *paymentChannel*) obsahují prefix *sessionInfo*.

URL nastavte podle 4.5 - URL platební brány GoPay.

Přesměrování na platební bránu GoPay

<https://gate.gopay.cz/gw/pay-full-v2?sessionInfo.paymentSessionId=3803928540&sessionInfo.targetGold=1803628540&sessionInfo.encryptedSignature=25ee53a1eccc253a8317bc9487174d09ba6b00a0f5267d2de6b483f58af9676d883e26600ce3316a>

Vzorové implementace pro sestavení přesměrování naleznete ve shodných zdrojových kódech s vytvořením platby.

4.4 Ověření stavu platby

Ověření stavu platby je prováděno při zpětném přesměrování z GoPay na E-shop, nebo po příchodu HTTP notifikace. URL přesměrování zpět na E-shop, ale i HTTP notifikace jsou doplněny shodnými parametry (viz 13.4 - Identita platby (*paymentIdentity*)).

Na *successURL* je zákazník přesměrován po provedení platby, nebo po úspěšném založení platby vybranou platební metodou (offline metody a internetová bankovníctví v nočních hodinách). Z toho

důvodu je nutné provádět kontrolu stavu platby prostřednictvím GoPay služby 7.1.2 - Stav platby.

Na *failedURL* je zákazník přesměrován pokud je provedeno zrušení platby. I v tomto případě lze provádět kontrolu dotazem na stav platby.

Zpracování HTTP notifikace zajišťuje doručení informace o provedení platby i v situaci, kdy na straně zákazníka dojde k přerušení připojení k internetu, atp. Z tohoto důvodu je zpracování povinnou součástí úplné integrace.

Vzorové implementace zpracování návratu na E-shop naleznete v příložených zdrojových kódech.

| Prostředí | implementace ověření stavu platby |
|---------------|--|
| PHP | viz 16.1, kontrola parametrů přesměrování a stavu platby naleznete ve skriptu <i>callback.php</i> . Kontrola na základě příchozí notifikace je obsažena ve skriptu <i>notify.php</i> . |
| Java | viz 16.2, kontrola po přesměrování zpět na platební bránu, nebo po příchodu HTTP notifikace je připravena ve třídě <i>SimplePayment</i> , metoda <i>successUrl</i> . |
| ASP.NET | viz 16.4, kontrola parametrů přesměrování a stavu platby naleznete ve třídě <i>Callback</i> , metoda <i>Call</i> . Kontrola na základě příchozí notifikace je obsažena ve třídě <i>Notify</i> , metoda <i>Notification</i> . |
| Python | viz 16.3, kontrola po přesměrování zpět na platební bránu, nebo po příchodu HTTP notifikace je připravena ve skriptu <i>SimplePayment.py</i> . |
| Ruby on Rails | viz 16.5, kontrola po přesměrování zpět na platební bránu, nebo po příchodu HTTP notifikace je připravena ve skriptu <i>SimplePayment.rb</i> . |

4.5 URL platební brány GoPay

URL pro přesměrování na platební bránu je uvedeno níže. V průběhu integrace používejte testovací instanci, která Vám umožní plné ověření veškerých funkcionalit systému.

Nastavení test aplikace je nutné i při používání API funkcionalit. Více informací naleznete vždy v příkladech užití jednotlivých API.

| Prostředí | URL |
|-----------|---|
| Test | http://testgw.gopay.cz/gw/pay-full-v2 |
| Produkce | https://gate.gopay.cz/gw/pay-full-v2 |

5 Před-autorizovaná platba

Před-autorizace je druh platby, která zajišťuje blokaci peněz na účtu zákazníka. V okamžiku dokončení platby na platební bráně nejsou prostředky převedeny na obchodní účet, ale je vytvořena blokáce na straně zákazníka. Blokáci lze na základě pokynu obchodníka zrušit nebo provést dokončení platby. Veškeré operace s autorizovanou platbou lze provádět po dobu 7dnů od založení platby. Platební metody podporující před-autorizace jsou uvedeny viz 2 - Platební brána.

Pro realizaci platby jsou použity principy úplné integrace viz 4 Úplná integrace. Platba je po založení a dokončení na platební bráně ve stavu *AUTHORIZED* viz 13.11 Stav platby. Pokud je platba ve stavu *AUTHORIZED*, potom je možné provést její dokončení nebo zrušení viz níže.

5.1 Založení platby

Princip fungování před-autorizovaných plateb je založen na úplné integraci 4. Pro založení platby je nutné nastavit, že se jedná o před-autorizaci viz 13.1 - Platební příkaz (paymentCommand), vlastnost *preAuthorization*. Implementace založení platby je připravena pro jednotlivé programovací jazyky viz níže.

| Prostředí | implementace ověření stavu platby |
|---------------|---|
| PHP | viz 16.1, resp. vytvoření platby je umístěno ve třídě GoPaySoap, funkce <i>createPreAuthorizedPayment()</i> . |
| Java | viz 16.2, resp. vytvoření platby je připraveno ve třídě SimplePayment, metoda <i>createPayment</i> . |
| ASP.NET | viz 16.4, resp. vytvoření platby je umístěno ve třídě GopayHelperWS, funkce <i>CreatePreAuthorizedPayment()</i> . |
| Python | viz 16.3, resp. vytvoření platby je připraveno ve třídě SimplePayment.py. |
| Ruby on Rails | viz 16.5, vytvoření platby naleznete ve skriptu <i>SimplePayment.rb</i> . |

Po úspěšném provedení před-autorizace je zákazník vrácen na *successURL* a platba se nachází ve stavu *AUTHORIZED* viz 13.11 Stav platby.

5.2 Dokončení platby

Dokončení platby je provedeno voláním webové služby 7.1.3 Provedení opakované platby. Implementaci dokončení před-autorizované platby je připravena pro jednotlivé programovací jazyky viz níže.

| Prostředí | implementace ověření stavu platby |
|-----------|---|
| PHP | viz 16.1, resp. dokončení platby je umístěno ve třídě GoPaySoap, funkce |

| | |
|---------------|--|
| | <i>capturePayment()</i> . |
| Java | viz 16.2, resp. dokončení platby je připraveno ve třídě <i>SimplePayment</i> , metoda <i>capturePayment</i> . |
| ASP.NET | viz 16.4, resp. dokončení platby je připraveno ve třídě <i>CapturePayment</i> , metoda <i>Capture</i> . |
| Python | viz 16.3, resp. dokončení platby je připraveno ve skriptu <i>RecurrentPaymentGPE.py</i> , metoda <i>capturePayment</i> . |
| Ruby on Rails | viz 16.5, resp. dokončení platby je připraveno ve skriptu <i>RecurrentPaymentGPE.rb</i> , metoda <i>capturePayment</i> . |

Dokončení před-autorizace lze volat pouze na platbách, které jsou založeny jako před-autorizované a platbách ve stavu *AUTHORIZED*. Po dokončení platby proběhne přechod do stavu *PAID*.

5.3 Zrušení před-autorizace

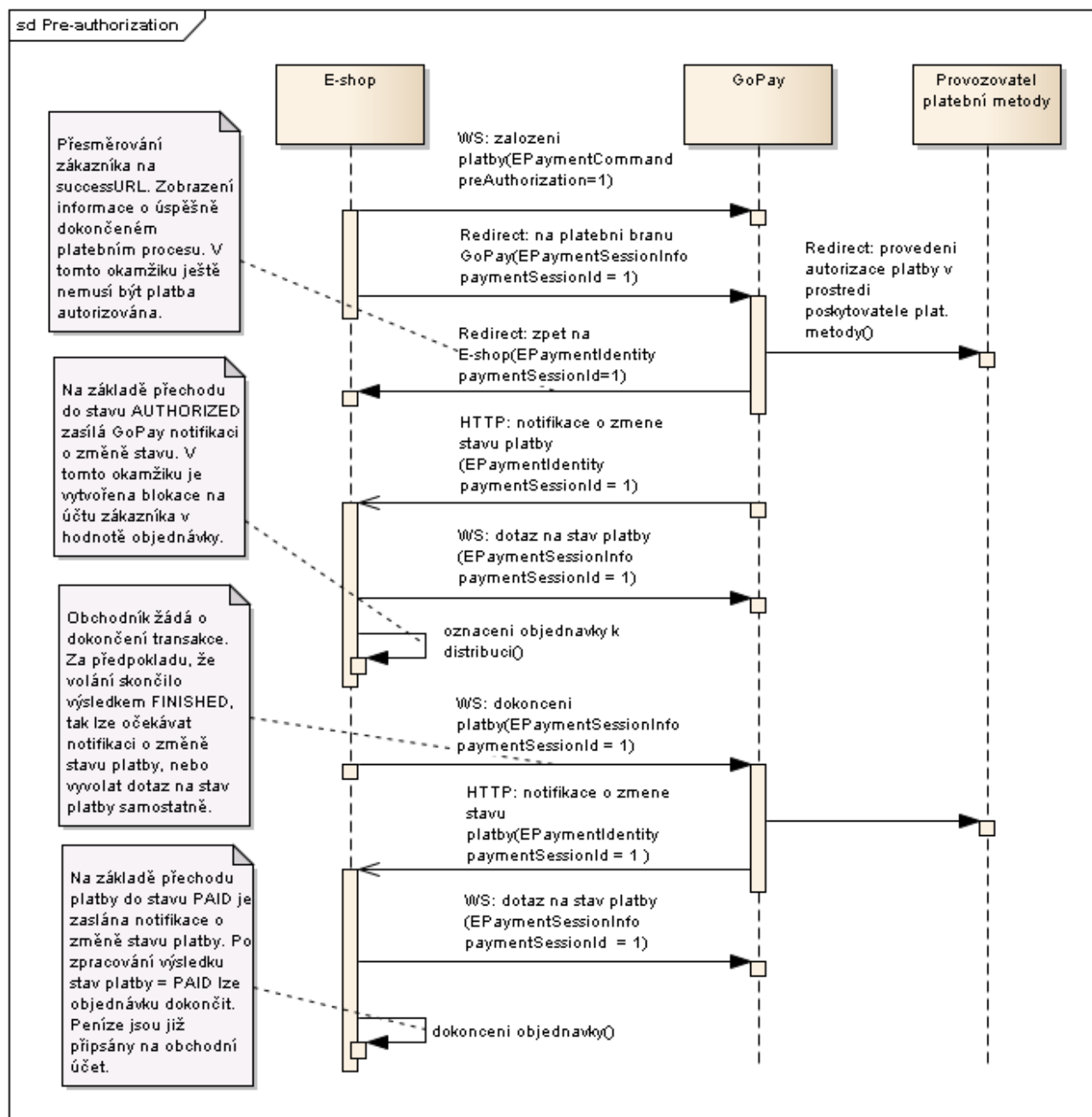
Uvolnění blokace na účtu zákazníka je provedena voláním webové služby 7.1.6 Zrušení před-autorizované platby. Implementaci zrušení před-autorizace je připravena pro jednotlivé programovací jazyky viz níže.

| Prostředí | implementace ověření stavu platby |
|---------------|--|
| PHP | viz 16.1, resp. zrušení před-autorizace je umístěno ve třídě <i>GoPaySoap</i> , funkce <i>voidAuthorization()</i> . |
| Java | viz 16.2, resp. zrušení před-autorizace je připraveno ve třídě <i>SimplePayment</i> , metoda <i>voidAuthorization</i> . |
| ASP.NET | viz 16.4, resp. zrušení před-autorizace je připraveno ve třídě <i>VoidAuthorization</i> , metoda <i>Void</i> . |
| Python | viz 16.3, resp. zrušení před-autorizace je připraveno ve skriptu <i>RecurrentPaymentGPE.py</i> , metoda <i>voidAuthorizedPayment</i> . |
| Ruby on Rails | viz 16.5, resp. zrušení před-autorizace je připraveno ve skriptu <i>RecurrentPaymentGPE.rb</i> , metoda <i>voidAuthorizedPayment</i> . |

Zrušení před-autorizace lze volat pouze na platbách, které jsou založeny jako před-autorizované a platbách ve stavu *AUTHORIZED*. Po dokončení platby proběhne přechod do stavu *CANCELED*.

5.4 Procesní diagramy

Procesní diagram průběhu před-autorizované platby je zobrazen na diagramu viz níže.



6 Opakovaná platba

Opakovaná platba je funkcionalita, která umožňuje přijímat platbu od zákazníka na pravidelné bázi. Zákazník je v okamžiku založení platby informován na platební bráně o jejích parametrech (částka, frekvence plateb, atd). Po úspěšném založení inicializační platby jsou prováděny automatizovaně s definovanou periodou nebo na vyžádání. Zákazník je informován formou emailu o každé dokončené platbě. Obchodník je informován formou notifikace o změně stavu platby. Platební metody podporující opakované platby jsou uvedeny viz 2 - Platební brána.

6.1 Založení platby

Princip fungování opakovaných plateb je založen na úplné integraci 4. Při založení platby je nutné nastavit vlastnosti specifické opakované platbě viz 13.1 - Platební příkaz (paymentCommand) (*recurrentPayment*, *recurrenceCycle*, *recurrencePeriod*, *recurrenceDateTo*). Implementace založení platby je připravena pro jednotlivé programovací jazyky viz níže.

| Prostředí | implementace ověření stavu platby |
|---------------|---|
| PHP | viz 16.1, resp. vytvoření platby je umístěno ve třídě GoPaySoap, funkce <i>createRecurrentPayment()</i> . |
| Java | viz 16.2, resp. vytvoření platby je připraveno ve třídě <i>RecurrentPayment</i> , metoda <i>createPayment</i> . |
| ASP.NET | viz 16.4, resp. vytvoření platby je připraveno ve třídě <i>RecurrentPayment</i> , metoda <i>Pay</i> . |
| Python | viz 16.3, resp. vytvoření platby je připraveno ve skriptu <i>RecurrentPaymentGPE.py</i> . |
| Ruby on Rails | viz 16.5, resp. vytvoření platby je připraveno ve skriptu <i>RecurrentPaymentGPE.rb</i> . |

Inicializační (první) platba je provedena okamžitě a zpracována viz 4 Úplná integrace, resp. zpracování notifikace o změně stavu platby.

6.2 Opakování platby

Následné platby jsou prováděny:

- automatizovaně - dle periody uvedené při zakládání platby *recurrenceCycle=DAY*, *WEEK*, *MONTH* v kombinaci s parametrem *recurrencePeriod=počet jednotek* (*recurrenceCycle=MONTH*, *recurrencePeriod=3* ~ opakování jednou za tři měsíce).
- na žádost - následné platby jsou prováděny na základě požadavku zaslaného prostřednictvím web-service *recurrenceCycle=ON_DEMAND*. U těchto plateb je možné s každou iterací volit částku a číslo objednávky.

Každá následná platba je identifikována *paymentSessionId* a *parentPaymentSessionId*. *ParentPaymentSessionId* odpovídá ID inicializační platby (rodičovská platba). Po provedení následné platby je E-shopu zaslána notifikace o změně stavu platby, kde

- *paymentSessionId* - ID nové platby
- *parentPaymentsessionId* - ID inicializační platby - na jejíž základě je následná platba prováděna.

Notifikace o změně stavu platby jsou doručovány viz 8.2 HTTP notifikace. Jde o proces, který informuje o změně stavu platby, tzn nelze na jehož základě provádět úhradu. Pro zjištění stavu platby je nutné dotázat viz 7.1.2 Stav platby. Implementace dotazu je připravena pro jednotlivé programovací jazyky viz níže.

| Prostředí | implementace ověření stavu platby, provedení opakované platby |
|---------------|---|
| PHP | viz 16.1, dotaz na stav platby je umístěn ve třídě <i>GoPaySoap</i> , funkce <i>isPaymentDone</i> . Provedení opakované platby (pouze pro <i>recurrenceCycle=ON_DEMAND</i>) je realizováno ve funkci <i>performRecurrentPayment</i> . |
| Java | viz 16.2, zpracování HTTP notifikace je popsáno ve třídě <i>RecurrentPaymentGPE</i> , metoda <i>notify</i> a provedení opakované platby v metodě <i>performRecurrence</i> . |
| ASP.NET | viz 16.4, dotaz na stav platby je umístěn ve třídě <i>GopayHelperWS</i> , funkce <i>IsPaymentDone</i> . Provedení opakované platby (pouze pro <i>recurrenceCycle=ON_DEMAND</i>) je realizováno ve třídě <i>RecurrentPayment</i> ve funkci <i>Pay</i> . |
| Python | viz 16.3, zpracování HTTP notifikace je popsáno ve skriptu <i>RecurrentPaymentGPE.py</i> , metoda <i>notify</i> a provedení opakované platby v metodě <i>performRecurrence</i> . |
| Ruby on Rails | viz 16.5, zpracování HTTP notifikace je popsáno ve skriptu <i>RecurrentPaymentGPE.rb</i> , metoda <i>notify</i> a provedení opakované platby v metodě <i>performRecurrence</i> . |

Opakované platby jsou podporovány na platebních metodách viz 2 Platební brána.

6.3 Zrušení opakování platby

V případě nutnosti lze opakování platby ukončit voláním WS viz 7.1.4 Zrušení opakování platby, nebo viz implementace.

| Prostředí | implementace ověření stavu platby |
|-----------|--|
| PHP | viz 16.1, resp. zrušení opakování je umístěno ve třídě <i>GoPaySoap</i> , funkce <i>voidRecurrentPayment()</i> . |
| Java | viz 16.2, zrušení opakování platby je popsáno ve třídě <i>RecurrentPayment</i> , metoda <i>voidRecurrence</i> . |
| ASP.NET | viz 16.4, zrušení opakování platby je popsáno ve třídě <i>VoidRecurrentPayment</i> , |

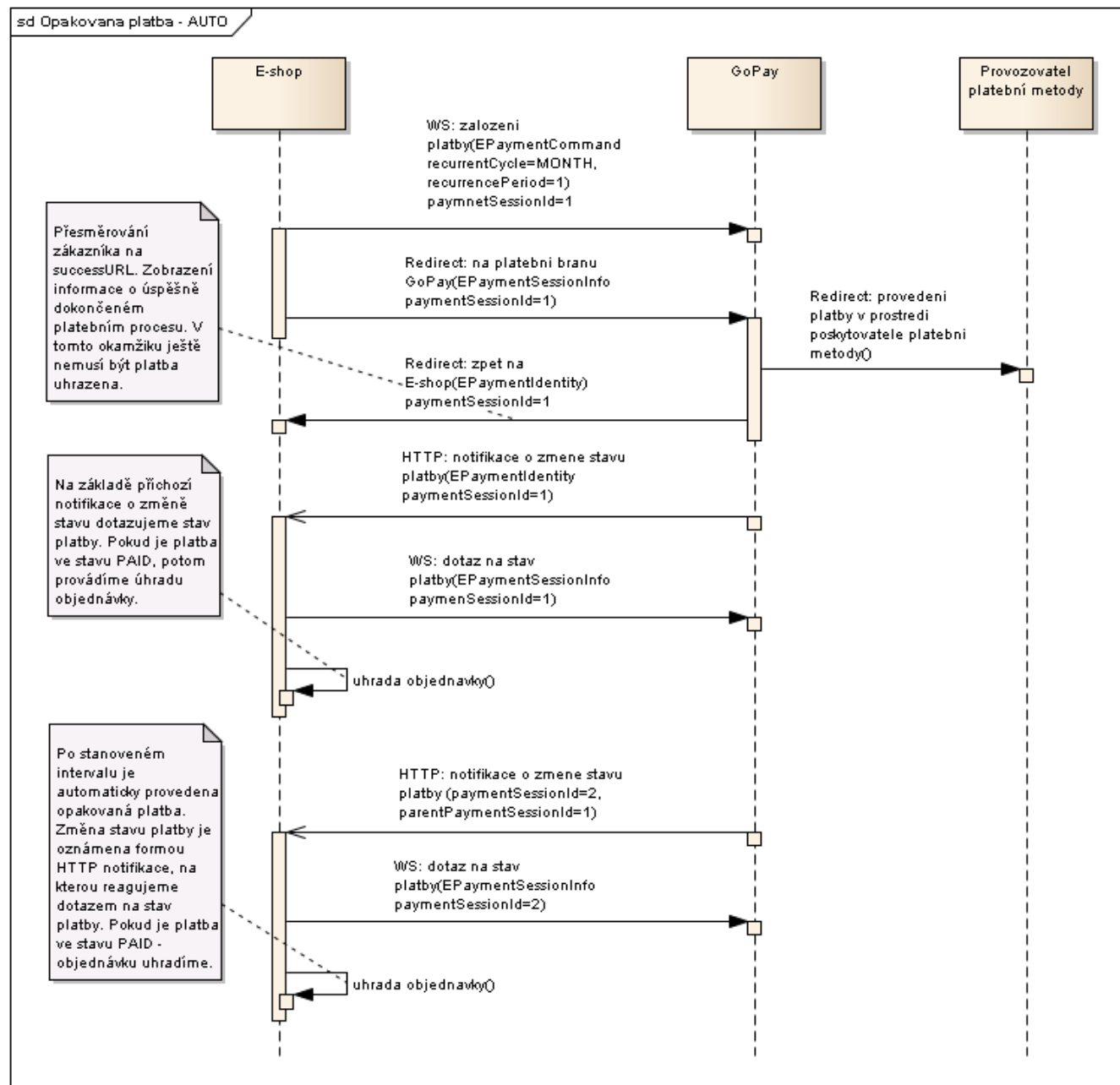


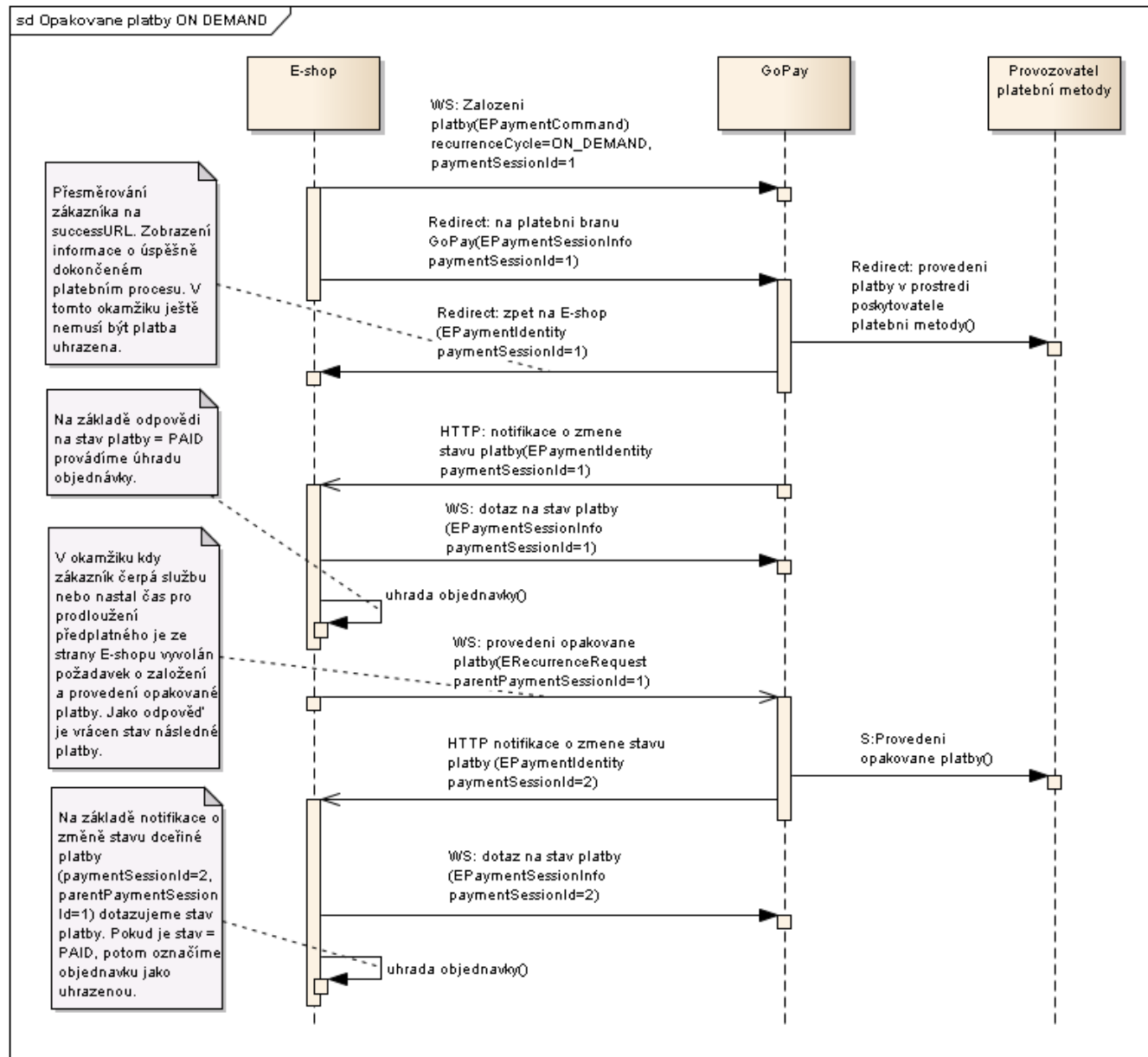
| | |
|---------------|--|
| | metoda <i>Void</i> . |
| Python | viz 16.3 , zrušení opakování platby je popsáno ve skriptu <i>RecurrentPaymentGPE.py</i> , metoda <i>voidRecurrence</i> . |
| Ruby on Rails | viz 16.5, <i>zrušení opakování platby je popsáno ve skriptu RecurrentPaymentGPE.</i> , metoda <i>voidRecurrence</i> . |

6.4 Procesní diagram

Celkový procesní diagram opakovaných plateb s automatickým opakováním je zobrazen níže.

Celkový procesní diagram opakovaných plateb s opakováním na požádání (on demand) je zobrazen níže.





7 GoPay služby

GoPay služby nabízí funkcionality pro přímé volání ze strany E-shopu. Funkcionality jsou dostupné pomocí GoPayWS(Web services), více viz 7.2 - GoPayWS.

7.1 Služby

7.1.1 Vytvoření platby

Vytvoření platby je použito jako součást úplné integrace. Při vytváření platby jsou předávány systému GoPay informace o platbě 13.1 - Platební příkaz (paymentCommand). Výsledek vytvoření platby 13.5 - Stav platby (paymentStatus) obsahuje parametry platby a identifikaci platby na straně GoPay (částka, název zboží, variabilní symbol, paymentSessionId, apod.).

7.1.2 Stav platby

Volání stav platby je použito jako součást úplné integrace pro kontrolu potvrzení, zrušení, vypršení platby apod. Parametrem volání je platební session 13.3 - Platební session (paymentSession). Volání vrací stav platby 13.5 - Stav platby (paymentStatus), který obsahuje parametry platby.

Stav platby je určen pro kontrolu údajů proti objednávce uložené na straně E-shopu (částka, název zboží, variabilní symbol, paymentSessionId, správný podpis apod.).

7.1.3 Provedení opakované platby

Provedení opakované platby slouží k založení následné platby na základě rodičovské platby. Funkcionalita je parametrizována požadavkem na opakování 13.5 Stav platby (paymentStatus). Za *parentPaymentSessionId* je nutné zadávat ID rodičovské platby.

Volání vrací informační element 13.5 Stav platby (paymentStatus), který popisuje aktuální stav následné platby. Standardně je po tomto volání ve stavu CREATED. Změna stavu platby bude předávána standardně formou notifikací o změně stavu platby.

7.1.4 Zrušení opakování platby

Zrušení opakované platby slouží pro ukončení provádění dalších opakovaných plateb. Funkcionalita je parametrizována platební session 13.3 - Platební session (paymentSession). Za *paymentSessionId* je nutné zadávat ID inicializační platby.

Volání vrací informační element 13.9 - Výsledek volání (paymentResult). V případě návratové hodnoty *Result*:

- *ACCEPTED* - pro zjištění stavu nutné dotaz opakovat s určitým časovým odstupem



- FINISHED -opakování platby je zrušeno
- FAILED - opakování platby nelze zrušit (nutné kontaktovat podporu GoPay)

7.1.5 Provedení před-autorizované platby

Před-autorizovaná platba zajišťuje blokaci peněžních prostředků na účtu zákazníka pro provedení transakce s určitým zpožděním (max 7 dnů). Dokončení před-autorizované platby provede převod prostředků z účtu zákazníka na obchodní účet v rámci GoPay.

Volání pro dokončení platby je parametrizováno platební session 13.3 - Platební session (paymentSession). Návratovou hodnotou je element 13.9 - Výsledek volání (paymentResult). V případě *Result*:

- *ACCEPTED* - dokončení před-autorizace zařazeno do fronty.
- FINISHED - transakce dokončena (současně budete informováni notifikací o změně platby - přechod do PAID)
- FAILED - dokončení transakce není možné (autorizace vypršela)

7.1.6 Zrušení před-autorizované platby

Před-autorizovaná platba blokuje peněžní prostředky na účtu zákazníka pro provedení transakce s určitým zpožděním. Tato služba provede zrušení blokace a uvolnění prostředků.

Volání pro dokončení platby je parametrizováno platební session 13.3 - Platební session (paymentSession). Návratovou hodnotou je element 13.9 - Výsledek volání (paymentResult). V případě *Result*:

- *ACCEPTED* - zrušení autorizace zařazeno do fronty
- FINISHED - blokace zrušena (současně budete informováni notifikací o změně platby - přechod do CANCELED)
- FAILED - dokončení transakce není možné (autorizace vypršela)

7.1.7 Založení uživatele

Funkcionalita umožňuje před-vytvořit uživatele GoPay peněženky (viz 11 - Založení uživatele). Parametrem volání je element založení platby 13.7 - Založení uživatele (buyerCreate). Volání vrací element výsledek založení uživatele 13.8 - Výsledek vytvoření uživatele (buyerCreateResult).

Element s výsledkem obsahuje vlastnost *result* a *resultDescription*. Result vyjadřuje, zda požadavek byl správně zpracován. Pokud nebyl správně zpracován potom situaci interpretujte jako systémovou chybu GoPay. Pokud byl správně zpracován potom jsou podstatné informace ve vlastnosti *resultDescription*.

7.2 GoPayWS

GoPayWS je webová služba, která implementuje funkcionality 7.1.1 - Vytvoření platby, 7.1.2 - Stav platby, 7.1.7 -Založení uživatele.

7.2.1 Definice WS

Definice webové služby je k dispozici na URL (viz níže).

| Prostředí | URL |
|-----------|---|
| Test | http://testgw.gopay.cz/axis/EPaymentServiceV2 |
| Produkce | https://gate.gopay.cz/axis/EPaymentServiceV2 |

7.3 Popis volání

Níže jsou uvedeny názvy metod, jejich parametry a návratové hodnoty.

Vytvoření platby

```
EPaymentStatus createPayment (EPaymentCommand paymentCommand) ;
```

- *EPaymentCommand* jehož vlastnosti odpovídají 13.1.
- *EPaymentStatus* jehož vlastnosti odpovídají 13.5

Opakování platby platby

```
EPaymentStatus createRecurrentPa (ERecurrencePayment recurrencePayment) ;
```

- *ERecurrencePayment* jehož vlastnosti odpovídají 13.2.
- *EPaymentStatus* jehož vlastnosti odpovídají 13.5

Stav platby

```
EPaymentStatus paymentStatus (EPaymentSessionInfo sessionInfo) ;
```

- *EPaymentSessionInfo* jehož vlastnosti odpovídají 13.3
- *EPaymentStatus* jehož vlastnosti odpovídají 13.5

Zrušení opakované platby

```
EPaymentResult voidRecurrentPayment(EPaymentSessionInfo sessionInfo);
```

- *EPaymentSessionInfo* jehož vlastnosti odpovídají 13.3
- *EPaymentResult* jehož vlastnosti odpovídají 13.9

Založení uživatele

```
EBuyerCreateResult createBuyer(EBuyerCreate buyerCreate);
```

- *EBuyerCreate* jehož vlastnosti odpovídají 13.7
- *EBuyerCreateResult* jehož vlastnosti odpovídají 13.8.

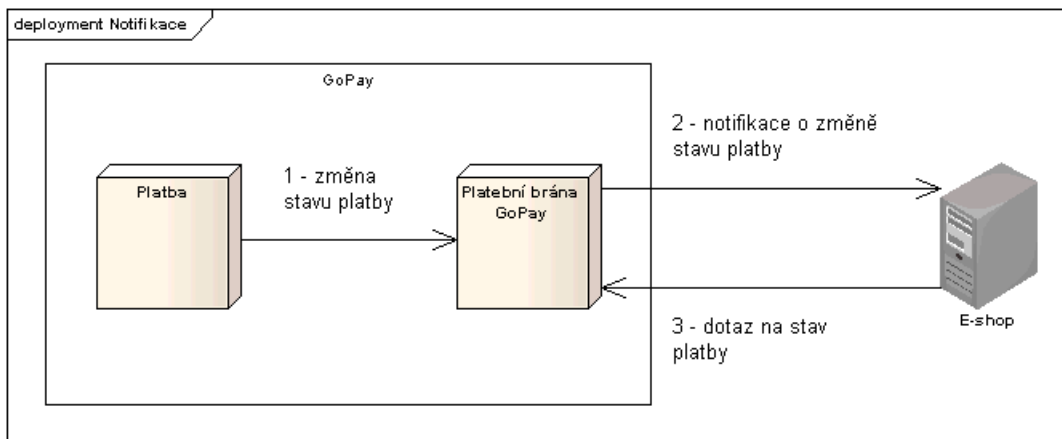
7.3.1 Použití WS

V tabulce (viz níže) naleznete odkazy na případy použití GoPayWS v jednotlivých programových prostředích s využitím dodávaného API.

| Prostředí | implementace WS klientské části |
|---------------|---|
| PHP | viz 16.1, užití WS naleznete ve třídě GopaySoap. |
| Java | viz 16.2, užití WS naleznete ve třídě SimplePayment. |
| ASP.NET | viz 16.4, užití WS naleznete ve třídě GopayHelperWS. |
| Python | viz 16.3, užití WS naleznete ve skriptu SimplePayment.py. |
| Ruby on Rails | viz 16.5, užití WS naleznete ve skriptu SimplePayment.rb. |

8 GoPay notifikace

Funkcionalita GoPay notifikace je určena pro předání informace o změně stavu platby. Existuje několik metod jak tuto notifikaci doručit HTTP, XML-RPC, Email, SMS.



Průběh notifikace:

1. Nastala změna stavu platby (viz 13.11 - Stav platby).
2. GoPay zasílá notifikaci, která obsahuje informace odpovídající elementu 13.4 - Identita platby (paymentIdentity). Pokud E-shop notifikaci nezpracuje, potom je volání opakováno. V případě notifikace E-mailem je notifikace považována za doručenu v okamžiku odeslání E-mailové zprávy.
3. E-shop provádí ověření platby pomocí GoPay služeb (viz 7.1.2 - Stav platby). Ověření platby je před jejím dokončením na straně E-shopu povinné.

8.1 Doručení notifikace

Notifikace je doručena podle níže uvedených pravidel:

| Typ notifikace | |
|----------------|--|
| HTTP | HTTP notifikace je považována za doručenu v okamžiku přijetí HTTP se status kódem 200. Pokud se nedaří notifikaci doručit, potom je doručení opakováno: <ul style="list-style-type: none"> do doby 1 hodina od provedení platby s periodou 10 minut od 1 hodiny od provedení s periodou 1 hodina |
| XML-RPC | XML-RPC notifikace je považována za doručenu pokud služba vrátí <i>notificationResult</i> = 1. Pokud se nedaří notifikaci doručit, potom je doručení opakováno: <ul style="list-style-type: none"> do doby 1 hodina od provedení platby s periodou 10 minut od 1 hodiny od provedení s periodou 1 hodina |
| E-mail | E-mail notifikace je považována za doručenu v okamžiku odeslání emailu |
| SMS | SMS notifikace je považována za doručenu v okamžiku odeslání SMS |

8.2 HTTP notifikace

Notifikace je realizována HTTP/GET požadavkem (viz níže). URL pro notifikaci je definováno v rámci registrace E-shopu. Při notifikaci se předávají parametry odpovídající identitě platby 13.4 - Identita platby (paymentIdentity).

Notifikace je považována za doručenu na základě odpovědi s HTTP status kódem 200.

Upozornění: HTTP notifikace nepodporují omezení přístupu formou HTTP autentizace.

Příklad notifikace:

HTTP notifikace

<http://www.eshop.cz/gopay/notification.action?paymentSessionId=3000011014&targetGold=8911180113&orderNumber=8911180113-367716&encryptedSignature=6e7734dec9f84e73a0f47330d79cf868a6476dce6c2a29f3d1803982c36c73ff05e4b0502c26f2f11420ea850d744202&p1=1234&p2=abcd&p3=5678&p4=ab12>

8.3 XML-RPC notifikace

Notifikace je realizována XML-RPC voláním, formát viz níže. URL umístění XML-RPC služby je definováno v rámci registrace E-shopu. Notifikace předává parametry odpovídající identitě platby 13.4 - Identita platby (paymentIdentity).

Notifikace je považována za doručenou na základě odpovědi s nastavenou vlastností *notificationResult = 1*.

Popis XML-RPC volání:

| xml-rpc notifikace |
|---|
| <pre>Map<String, Object> notifyPayment(String paymentSessionId, String variableSymbol, String eshopGold, String buyerGold, String encryptedSignature)</pre> |

Odpověď je předávána formou mapy (<struct>) viz níže. Odpověď obsahuje následující vlastnosti.

| Vlastnost | Popis |
|-------------------------|---|
| notificationResult | 1 – OK, -1 Failed. Notifikace je považována za doručenou pokud nabývá hodnoty 1. (Povinný parametr) |
| notificationDescription | Doprovodný text (Nepovinný parametr) |
| universalId | ID pod kterým je platba evidována na straně E-shopu. (Nepovinný parametr) |

Formát XML-RPC volání:

| | URL |
|-----------------------|---|
| Notifikace | https://www.gopay.cz/download/notification/notify_request.xml |
| Odpověď na notifikaci | https://www.gopay.cz/download/notification/notify_response.xml |

8.4 Email notifikace

Notifikace je zasílána formou emailové zprávy. Formát (viz níže). Notifikační emailová adresa je nastavena v průběhu integrace E-shopu.

Formát E-mailu

Notifikační E-mail - platba provedena

*** Na tento email neodpovídejte. Email má pouze informační charakter. ***

Vážený obchodníku,

byla úspěšně provedena platba za objednávku:

ID objednávky: \$orderId

Popis objednávky: \$orderName

Částka: \$amount Kč

Obchodní místo: \$shopName

Zvolená platební metoda: \${paymentMethod}

Čas/Datum vytvoření objednávky: \$dateCreated

Stav objednávky: ZAPLACENO/YPRŠELA/ZRUŠENA

ID platby: \$paymentSessionId

\$note

Více informací o této platbě naleznete v GoPay Monitoru nebo kontaktujte naši zákaznickou podporu na emailu podpora@gopay.cz.

Děkujeme za využívání našich služeb.

S pozdravem

Tým GoPay



8.5 SMS notifikace

Notifikace je zasílána formou SMS zprávy. Formát viz níže. Notifikační telefonní číslo je nastaveno v průběhu integrace E-shopu. Notifikace se používá pouze pro informování o provedení platby. Zrušení nebo vypršení platby notifikováno není.

Formát SMS

Notifikační SMS - platba provedena

Zprava pro prodejce EVC:{0}. Na Vas GoPay obchodni ucet byla pripsana platba ve vysí:{1}Kc. E-shop
GoID:{2}, VS:{3}, Poznamka:{4}



9 Vyúčtování

GoPay poskytuje dokument o vyúčtování v XML formátu. Dokument je k dispozici ke stažení z GoPay Monitoru (Obchodní účet > Přehled vyúčtování).

Popis XML struktury:

| | URL |
|--------------------|---|
| Schéma | https://www.gopay.cz/download/clearing/clearing.xsd |
| Příklad vyúčtování | https://www.gopay.cz/download/clearing/clearing_example.xml |

10 Výpis pohybů z obchodního účtu

Výpis pohybů z obchodního účtu lze získat v prostředí GoPay monitoru, nebo prostřednictvím webové služby. V rámci GoPay monitoru je možné vytvořit výpis ve formátu XLS nebo CSV. Webová služba generuje výpis ve formátu CSV, kódování CP1250.

Příklad užití webové služby pro stažení výpisu je připraven pro následující programové prostředí:

| Prostředí | implementace GoPay HTTP klientské části |
|---------------|---|
| PHP | viz 16.1, užití GoPay HTTP naleznete ve třídě GoPayHTTP funkce getAccountStatement. |
| Java | viz 16.2, užití GoPay HTTP naleznete ve třídě GetAccountStatement. |
| ASP.NET | viz 16.4, užití WS naleznete ve třídě AvailablePaymentMethods. |
| Python | viz 16.3, užití GoPay HTTP naleznete ve skriptu GetAccountStatement.py. |
| Ruby on Rails | viz 16.5, užití GoPay HTTP naleznete ve skriptu GetAccountStatement.rb. |

URL pro získání výpisu pohybů na obchodním účtu je zobrazen níže.

| Prostředí | URL |
|-----------|---|
| Test | https://testgw.gopay.cz/gw/services/get-account-statement |
| Produkce | https://gate.gopay.cz/gw/services/get-account-statement |

Parametry volání odpovídají 13.6 - Výpis z účtu (eStatementRequest).

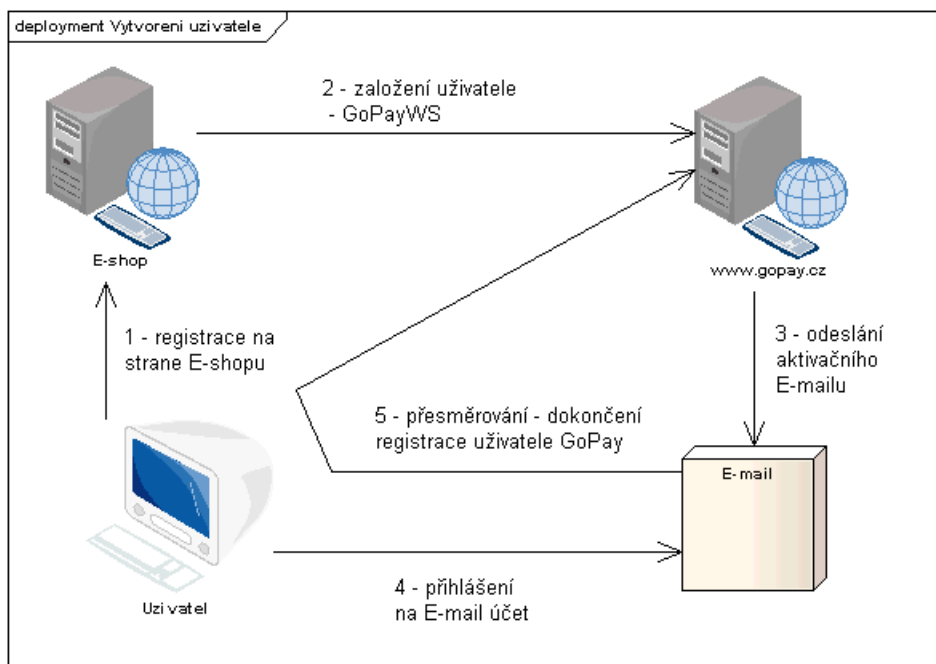
Výsledkem volání je soubor dat ve formátu CSV. V případě nevalidních vstupů jsou vrácena data s popisem chyby.

Popis hodnot, jak jdou v CSV za sebou:

- Identifikátor pohybu v systému GoPay
- Datum provedení pohybu
- Typ pohybu – např. Platba na e-shop, vyúčtování, poplatek, atd.
- Původce platby – např. Eplatby, Mojeplatba, GoPay, atd.
- Identifikace objednávky v rámci E-shopu. Alfanumerické označení objednávky max. 128 znaků.
- Částka
- Stav účtu před provedením pohybu
- Stav účtu po provedení pohybu
- Měna, ve které transakce proběhla
- Identifikátor pohybu, ke kterému se daný pohyb vztahuje – např. u poplatku udává Id pohybu vyúčtování, ze kterého je poplatek vypočítán.

11 Založení uživatele

GoPay umožňuje aplikaci obchodníka před-vytvořit uživatele GoPay. Tímto zajišťuje uživateli jednotný přístup k aplikaci obchodníka a GoPay peněžence na úrovni uživatelského jména.



Tato služba viz výše uvedený diagram zajišťuje pouze před-vytvoření uživatelského profilu. Uživatel musí na základě obdrženého aktivačního E-mailu registraci dokončit. Registrace je spojena se zadáním osobních údajů a souhlasem s obchodními podmínkami GoPay.

Podmínky pro úspěšné vytvoření uživatele jsou následující:

- Předávané uživatelské jméno musí být unikátní v rámci GoPay,
- povolené znaky pro uživatelské jméno [A-z,a-z,0-9,!,',_],
- min délka uživatelského jména 6 znaků,
- předávaný E-mail účet musí být unikátní v rámci GoPay.

Funkcionalita je dostupná pomocí služeb GoPay (viz 7.1.7 - Založení uživatele).

12 Testování plateb

Pro vývojové účely je připravena testovací instance platební brány GoPay. Testovací platební brána je provozována na jiném URL, nežli je produkční, což je nutné brát v potaz během realizace. V integračním manuálu jsou URL pro přesměrování na platební bránu, vytváření plateb nebo webových služeb uvedeny v odpovídajících kapitolách.

Údaje pro konfiguraci testovacího rozhraní jsou předány obchodníkovi na základě vyplněného RF (Registračního formuláře), který získá při komunikaci s obchodním oddělením GoPay (viz 3 - Postup integrace).

Veškeré operace na testovacím prostředí platební brány GoPay jsou prováděny s fiktivními penězi. A je nutné, aby obchodník zabránil svým zákazníkům přístup na testovací rozhraní GoPay platební brány.

Upozornění: přístup do testovací GoPay peněženky a testovacího GoPay Monitoru je omezen na množinu IP adres. Používané IP adresy je nutné předat před zahájením integrace.

12.1 GoPay platební brána

Testovací platební brána umožňuje provedení platby všemi dostupnými platebními metodami. Pro tyto účely je sestaveno webové rozhraní, které simuluje chování dodavatele platební metody. Resp. co nejméně kopíruje komunikaci s ním, ale webové rozhraní jako takové je zcela odlišné.

Pro provedení platby prostřednictvím on-line metod (viz 2 - Platební brána) je umožněno provést platbu přímo prostřednictvím průvodce platby. Průvodce obsahuje instrukce pro provedení nebo zrušení platby.

Pro offline platby získá zákazník na platební bráně pouze podklady pro provedení platby. Z tohoto důvodu je nutné provést úhradu z jiného prostředí (viz 12.1.2 - Platební metoda superCASH).

Přístup na testovací platební bránu a webové služby není omezen IP adresou.

12.1.1 Platební metoda GoPay

Test platby z GoPay peněženky je umožněn pomocí testovacího prostředí brány GoPay. K testu je připraven uživatel s virtuálním kreditem.

| | přístupové údaje |
|--------------|------------------|
| Go přezdívka | test01 |
| Heslo | Test0001 |

12.1.2 Platební metoda superCASH

Test platby superCASH je realizován z testovacího prostředí platební brány GoPay. Při volbě platba superCASH je vygenerováno fiktivní číslo superCASH. Pro úhradu zadejte toto číslo do rozhraní viz níže a proveďte platbu dle uvedených instrukcí. Během 2 minut bude Vašemu E-shopu doručena HTTP notifikace o provedení platby.

| | URL |
|---------------------------|---|
| simulace platby superCASH | https://testgw.gopay.cz/gp-gateways/supercash/gateway.action |

12.1.3 Opakovaná platba

Test opakované platby je realizován z testovacího prostředí založením opakované platby viz 6 Opakovaná platba. První platba je zpracována okamžitě a platby následné lze uměle vyvolat prostřednictvím rozhraní viz níže.

| | URL |
|--|---|
| simulace platby provedení opakované platby | https://testgw.gopay.cz/gp-gateways/recurrent/gateway.action |

12.1.4 Platební metoda PremiumSMS

Platba prostřednictvím PremiumSMS vyžaduje zadání kódu GoKupónu v odpovídající hodnotě. Fiktivní GoKupón je možné vytvořit na webovém rozhraní (viz níže). Na provozní platební bráně jsou kódy zasílány na mobilní telefon, z něhož byla odeslána Premium SMS.

| | URL |
|--------------------------------|---|
| vytvoření virtuálního SMS kódu | http://testgw.gopay.cz/gp-gateways/creditsms/gateway.action?gopayId=1 |



12.2 GoPay Monitor

Prostředí GoPay monitoru je dostupné i na testovací instanci platební brány. GoPay Monitor je nástrojem obchodníka pro ovládání obchodního účtu. V GoPay monitoru můžete např. ověřit princip fungování vyúčtování obchodního účtu, atd.

| Prostředí | URL | IP omezení |
|---------------|---|------------|
| GoPay monitor | https://test.gopay.cz/gopay-monitor/prihlaseni | ANO |

13 Komunikační elementy

13.1 Platební příkaz (*paymentCommand*)

Níže popsané parametry jsou používány pro sestavení platebního tlačítka a v rámci úplné integrace jsou použity ve volání pro vytvoření platby (viz 7.1.1 - Vytvoření platby).

| Název parametru | Popis parametru | Povinný |
|-----------------------|--|----------------|
| successURL | Návratové URL volané po úspěšném dokončení platebního procesu. Formát odpovídající RFC 2396. | ANO |
| failedURL | Návratové URL volané po zrušení platby. Formát odpovídající RFC 2396. | ANO |
| productName | Popis zboží/služby (max. 256 znaků). | ANO |
| targetGold | Identifikace příjemce platby. Unikátní označení E-shopu/GoPay uživatele v rámci GoPay. Celočíselný identifikátor – definován během integrace E-shopu. | ANO |
| orderNumber | Identifikace objednávky v rámci E-shopu. Alfanumerické označení objednávky (max. 128 znaků). | ANO |
| totalPrice | Cena zboží/služby v haléřích. Celočíselné vyjádření ceny. | ANO |
| currency | Měna, ve které je hrazena cena za zboží/službu. Formát měny se řídí mezinárodním měnovým standardem ISO 4217 viz 13.15. | ANO |
| paymentChannels | Výčet platebních metod, které jsou zobrazeny na platební bráně. Kódy platebních metod najdete ve sekci 13.10. Jednotlivé kódy jsou odděleny “;”. Jde o volitelný parametr, pokud není uveden, zobrazí se všechny platební metody. Zobrazit lze pouze platební metody, které jsou pro daný E-shop aktivovány. | NE |
| defaultPaymentChannel | Textový řetězec označující před-vybranou platební metodu, nabývá hodnot viz 13.10. | NE |
| preAuthorization | Pro založení před-autorizované platby nastavte 1. Může nabývat hodnot 0 (NE) nebo 1(ANO). Dokončení platby (zboží je na skladě) je nutné zavolat GoPay službu 7.1.5 - Provedení před-autorizované platby. | NE |
| recurrentPayment | Pro založení opakované platby nastavte 1. Může nabývat hodnot 0 (NE) nebo 1(ANO). Četnost opakování je popsáno následujícími parametry. | NE |
| recurrenceCycle | Nabývá hodnot [DAY, WEEK, MONTH], které nastavují základní časovou jednotku opakování. Pro opakování od ČS a.s. lze | ANO - pokud je |

| | | |
|--------------------------|--|----------------------------|
| | použít pouze hodnotu DAY. | recurrent Payment= 1 |
| recurrencePeriod | Společně s <i>recurrenceCycle</i> definuje periodu opakované platby. Např. při konfiguraci DAY,5 bude platba prováděna každý 5-tý den. | |
| recurrenceDateTo | Definuje datum, do něhož budou prováděny opakované platby. Jedná se textový řetězec ve formátu yyyy-MM-dd. | |
| encryptedSignature | Podpis platebního příkazu. Sestavení podpisu viz 13.17 a sestavení řetězce pro podpis viz níže. | ANO |
| customerData.firstName | Jméno zákazníka (max. 128 znaků). | NE |
| customerData.lastName | Příjmení zákazníka (max. 128 znaků). | NE |
| customerData.email | Emailová adresa zákazníka (max. 128 znaků). | ANO |
| customerData.phoneNumber | Telefonní kontakt na zákazníka. | NE |
| customerData.street | Bydliště zákazníka - ulice (max. 256 znaků) | NE |
| customerData.city | Bydliště zákazníka - město (max. 128 znaků) | NE |
| customerData.postalCode | Bydliště zákazníka – poštovní směrovací číslo. | NE |
| customerData.countryCode | Bydliště zákazníka – kód země. Kódy země jsou uvedeny v číselníku 13.14. | NE |
| p1 - p4 | Volitelný parametr (max. 128 znaků). Parametry jsou vráceny v nezměněné podobě jako součást 13.11 a 13.4. | NE |
| lang | Explicitní nastavení jazykové mutace platební brány viz 13.16. | NE |

Řetězec pro podpis

Šifrovaný podpis je sestaven z vlastností platebního příkazu. Podpis je šifrován tajným klíčem označeným jako *secureKey*, který je předáván v rámci integrace.

Řetězec pro sestavení podpisu

```
targetGold+'|'+productName+'|'+totalPrice+'|'+currency+'|'+orderNumber+'|'+failedURL+'|'+successURL+'|'+preAuthorization+'|'+recurrentPayment+'|'+recurrenceDateTo+'|'+recurrenceCycle+'|'+recurrencePeriod+'|'+paymentChannels+'|'+secureKey
```

13.2 Opakování platby (*recurrenceRequest*)

Element opakování platby je použit při požadavku na vytvoření a provedení následné platby (viz 6.2 Opakování platby).

| Název parametru | Popis parametru | Povinný |
|------------------------|---|---------|
| targetGold | Identifikace příjemce platby. Unikátní označení E-shopu/GoPay uživatele v rámci GoPay. Celočíselný identifikátor – definován během integrace E-shopu. | ANO |
| parentPaymentSessionId | Identifikace rodičovské platby. Celočíselný identifikátor - předáván při vytvoření rodičovské platby. | ANO |
| orderNumber | Identifikace objednávky v rámci E-shopu. Alfanumerické označení objednávky (max. 128 znaků). | ANO |
| totalPrice | Cena zboží/služby v haléřích. Celočíselné vyjádření ceny. | ANO |
| encryptedSignature | Podpis platebního příkazu. Sestavení podpisu viz 13.17 a sestavení řetězce pro podpis viz níže. | ANO |

Řetězec pro podpis

Šifrovaný podpis je sestaven z vlastností žádosti o opakování platby. Podpis je šifrován tajným klíčem označeným jako *secureKey*, který je předáván v rámci integrace.

Řetězec pro sestavení podpisu

```
parentPaymentSessionId+'|'+targetGold+'|'+orderNumber+'|'+totalPrice+'|'+secureKey
```

13.3 Platební session (paymentSession)

Platební session je využita při úplné integraci v rámci přesměrování na platební bránu GoPay. Dále se používá při volání GoPay služby stav platby (viz 13.5 - Stav platby (paymentStatus)), atd..

| Název parametru | Popis parametru | Povinný |
|--------------------|---|---------|
| paymentSessionId | Identifikace platby v systému GoPay. Celé číslo. | ANO |
| targetGold | Identifikace E-shopu/GoPay uživatele v rámci GoPay. Celočíslný identifikátor – definován během integrace E-shopu. | ANO |
| encryptedSignature | Podpis platebního příkazu. Sestavení podpisu viz 13.17 a sestavení řetězce pro podpis viz níže. | ANO |

Řetězec pro podpis

Šifrovaný podpis je sestaven z vlastností Platebního session. Podpis je šifrován tajným klíčem označeným jako *secureKey*, který je předáván v rámci integrace E-shopu.

Řetězec pro sestavení podpisu

```
targetGold+'|'+paymentSessionId+'|'+secureKey
```

13.4 Identita platby (paymentIdentity)

Identita platby se používá při úplné integraci jako součást parametrů přesměrování z platební brány GoPay zpět na E-shop po provedení/zrušení platby.

| Název parametru | Popis parametru | Povinný |
|------------------------|--|---------|
| paymentSessionId | Identifikace platby v systému GoPay. Celé číslo. | ANO |
| targetGold | Identifikace E-shopu v rámci GoPay. Celočíselný identifikátor – definován během integrace E-shopu. | ANO |
| orderNumber | Identifikace objednávky v rámci E-shopu. Alfanumerické označení objednávky max. 128 znaků. | ANO |
| parentPaymentSessionId | Identifikace rodičovské platby. Používané u opakovaných plateb viz 6-Opakovaná platba. | NE |
| encryptedSignature | Podpis platebního příkazu. Sestavení podpisu viz 13.17 a sestavení řetězce pro podpis viz níže. | ANO |
| p1 - p4 | Volitelný parametr (max. 128 znaků). | NE |

Řetězec pro podpis

Šifrovaný podpis je sestaven z vlastností Identity platby. Podpis je šifrován tajným klíčem označeným jako *secureKey*, který je předáván v rámci integrace E-shopu.

Řetězec pro sestavení podpisu

```
targetGold+'|'+paymentSessionId+'|'+parentPaymentSessionId+'|'+orderNumber+'|'+secureKey
```


13.5 Stav platby (*paymentStatus*)

Obsahuje informace o stavu platby. Skládá se z následujících parametrů.

| Název parametru | Popis parametru | Povinný |
|------------------------|---|---------|
| paymentSessionId | Identifikace platby v systému GoPay. Celé číslo. | ANO |
| parentPaymentSessionId | Identifikace rodičovské platby. Používáno pro opakované platby. | NE |
| targetGold | Identifikace E-shopu/GoPay uživatele v rámci GoPay. Celočíslný identifikátor – definován během integrace E-shopu. | ANO |
| productName | Popis zboží/služby. Předáný při založení Platby. | ANO |
| orderNumber | Identifikace objednávky v rámci E-shopu. Alfamerické označení objednávky max. 128 znaků. | ANO |
| totalPrice | Cena zboží/služby v haléřích. Celočíslné vyjádření ceny. Předáný při založení Platby. | ANO |
| currency | Určuje měnu platby. Formát měny odpovídá ISO 4217 viz 13.15. | ANO |
| sessionState | Stav platby – kódy reprezentující stav platby jsou popsány v 13.11. | ANO |
| sessionSubState | Doplňující stav platby – kódy reprezentují specifické situace viz 13.12. | NE |
| sessionSubStateDesc | Textový popis doplňující informace. | NE |
| result | CALL_COMPLETED – volání WS proběhlo bezchybně, CALL_FAILED – volání WS proběhlo s chybou (např. neodpovídající podpis). | ANO |
| resultDescription | Popis chybové situace při CALL_FAILED. | NE |
| preAuthorization | Informuje, zda byla platba založena jako předautorizovaná platba. | NE |
| recurrentPayment | Informuje, zda byla platba založena jako opakovaná platba. | NE |
| paymentChannel | Určuje platební metodu, kterou zákazník vybral pro úhradu platby hodnot viz 13.10. | ANO |
| encryptedSignature | Podpis platebního příkazu. Sestavení podpisu viz 13.17 a sestavení řetězce pro podpis viz níže. | ANO |
| p1 - p4 | Volitelný parametr (max. 128 znaků). | NE |



Řetězec pro podpis

Šifrovaný podpis je sestaven z vlastností stavu platby. Podpis je šifrován tajným klíčem označeným jako `secureKey`, který je předáván v rámci integrace E-shopu.

Řetězec pro sestavení podpisu

```
targetGold+'|'+productName+'|'+totalPrice+'|'+currency+'|'+orderNumber+'|'+recurrentPayment+'|'+parentPaymentSessionId+'|'+preAuthorization+'|'+result+'|'+sessionState+'|'+sessionSubState+'|'+paymentChannel+'|'+secureKey
```

13.6 Výpis z účtu (eStatementRequest)

Element je vstupním parametrem pro výpis pohybů na obchodním účtu 10 - Výpis pohybů z obchodního účtu.

| Název parametru | Popis parametru | Povinný |
|--------------------|---|---------|
| dateFrom | Datum, od kterého je výpis generován (datum včetně). Datum je řetězec ve formátu YYYY-MM-DD. | ANO |
| dateTo | Datum, do kterého je výpis generován (datum včetně). Období je včetně tohoto dne. Datum je řetězec ve formátu YYYY-MM-DD. | ANO |
| targetGold | Identifikace E-shopu v rámci GoPay. Celočíselný identifikátor – definován během integrace E-shopu. | ANO |
| encryptedSignature | Podpis platebního příkazu. Sestavení podpisu viz 13.17 a sestavení řetězce pro podpis viz níže. | ANO |

Řetězec pro podpis

Šifrovaný podpis je sestaven z vlastností požadavku výpisu pohybů na obchodním účtu. Podpis je šifrován tajným klíčem označeným jako *secureKey*, který je předáván v rámci integrace E-shopu.

Řetězec pro sestavení podpisu

```
dateFrom+'|'+dateTo+'|'+targetGold+'|'+secureKey
```

13.7 Založení uživatele (*buyerCreate*)

Element je vstupním parametrem pro založení uživatele GoPay 7.1.7 - Založení uživatele.

| Název parametru | Popis parametru | Povinný |
|--------------------|---|---------|
| buyerUsername | Uživatelské jméno – unikátní označení uživatele GoPay. Sestaveno ze znaků ['A-Z', 'a-z', '.', '_', ':'], max 128 znaků. | ANO |
| buyerEmail | E-mail kontakt na uživatele – unikátní E-mail GoPay. Musí splňovat formát E-mailu, max 128 znaků. | ANO |
| eshopGold | Identifikace E-shopu v rámci GoPay. Celočíselný identifikátor – definován během integrace E-shopu. | ANO |
| encryptedSignature | Podpis platebního příkazu. Sestavení podpisu viz 13.17 a sestavení řetězce pro podpis viz níže. | ANO |

Řetězec pro podpis

Šifrovaný podpis je sestaven z vlastností založení uživatele. Podpis je šifrován tajným klíčem označeným jako *secureKey*, který je předáván v rámci integrace E-shopu.

Řetězec pro sestavení podpisu

```
eshopGold+'|'+buyerUsername+'|'+buyerEmail+'|'+secureKey
```

13.8 Výsledek vytvoření uživatele (buyerCreateResult)

Obsahuje informaci o výsledku volání 7.1.7 - Založení uživatele. Obsahuje následující informace:

| Název parametru | Popis parametru | Povinný |
|--------------------|---|---------|
| buyerUsername | Uživatelské jméno GoPay | ANO |
| eshopGold | Identifikace E-shopu v rámci GoPay. Celočíselný identifikátor – definován během integrace E-shopu. | ANO |
| buyerGold | Identifikace uživatele GoPay | ANO |
| result | CALL_COMPLETED – volání proběhlo bezchybně, CALL_FAILED – volání proběhlo s chybou (např. neodpovídající podpis). | ANO |
| resultDescription | Popis výsledku: INTERNAL_ERROR, INVALID_INPUT, BUYER_CREATED, EMAIL_NOT_UNIQUE, USERNAME_NOT_UNIQUE, INVALID_EMAIL, INVALID_USERNAME. | ANO |
| encryptedSignature | Podpis platebního příkazu. Sestavení podpisu viz 13.17 a sestavení řetězce pro podpis viz níže. | ANO |

Řetězec pro podpis

Šifrovaný podpis je sestaven z výše uvedených vlastností. Podpis je šifrován tajným klíčem označeným jako *secureKey*, který je předáván v rámci integrace E-shopu.

Řetězec pro sestavení podpisu

```
eshopGold+'|'+buyerUsername+'|'+buyerGold+'|'+result+'|'+resultDescription+'|'+secureKey
```

13.9 Výsledek volání (*paymentResult*)

Obsahuje informaci o výsledku volání 7.1.4 - Zrušení opakování platby, 7.1.5 - Provedení před-autorizované platby. Obsahuje následující informace:

| Název parametru | Popis parametru | Povinný |
|--------------------|---|---------|
| paymentSessionId | Uživatelské jméno GoPay | ANO |
| result | ACCEPTED - požadavek přijat (zařazen do fronty), FINISHED - požadavek úspěšně zpracován, FAILED - požadavek nelze provést | ANO |
| resultDescription | Textový popis výsledku | ANO |
| encryptedSignature | Podpis platebního příkazu. Sestavení podpisu viz 13.17 a sestavení řetězce pro podpis viz níže. | ANO |

Řetězec pro podpis

Šifrovaný podpis je sestaven z výše uvedených vlastností. Podpis je šifrován tajným klíčem označeným jako *secureKey*, který je předáván v rámci integrace E-shopu.

Řetězec pro sestavení podpisu

```
paymentSessionId+'|'+result+'|'+secureKey
```

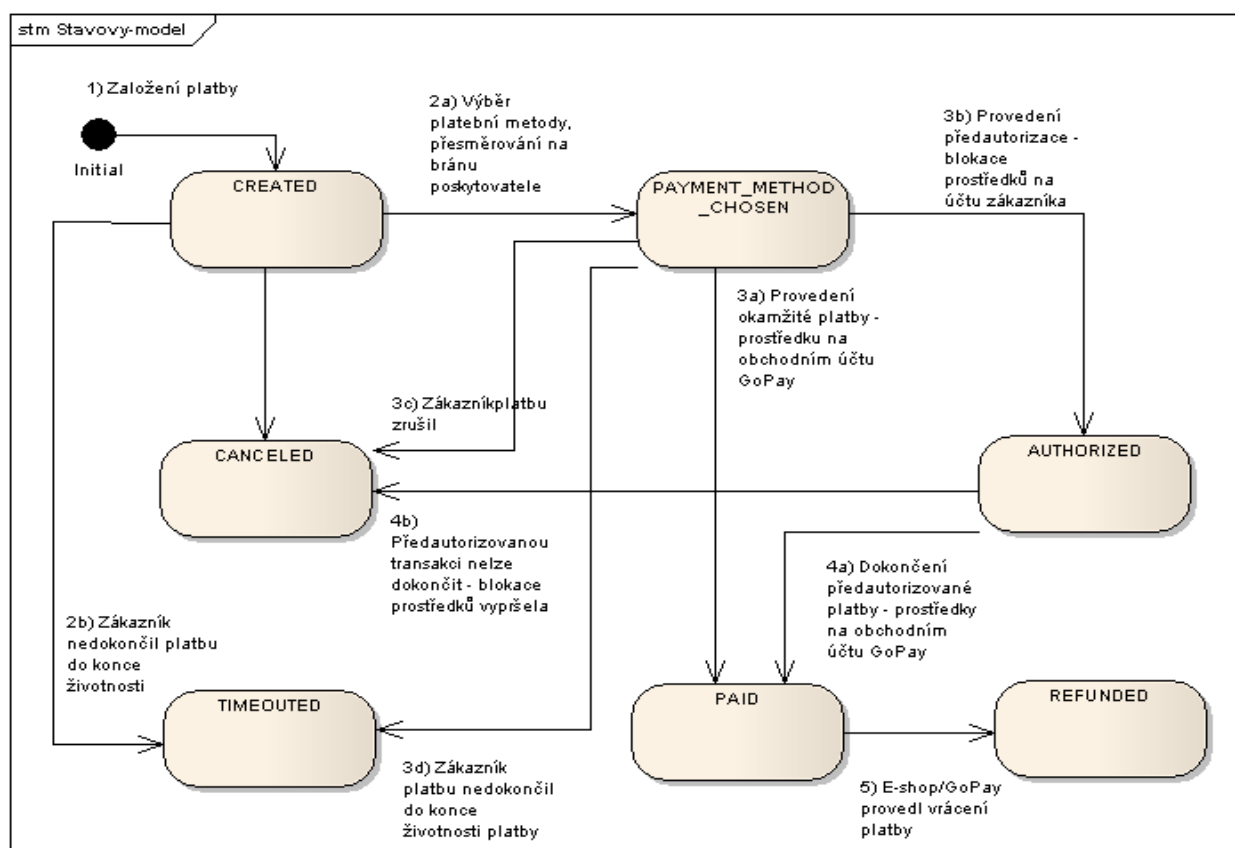
13.10 Kódy platebních metod

| Kód platebního kanálu | Popis parametru | Platba realizována prostřednictvím |
|-----------------------|---|---|
| cz_cs_c | Platba prostřednictvím GoPay peněženky- platební karty MasterCard, VISA | Česká spořitelna, a.s. - E-commerce 3-D Secure |
| eu_gp_u | Platba prostřednictvím GoPay peněženky- platební karty MasterCard, VISA | UniCredit Bank - Global payments |
| | | |
| SUPERCASH | SuperCASH | Terminál České pošty, Sazka a.s. |
| | | |
| eu_pr_sms | Premium SMS | Mobilní telefon - Premium SMS |
| cz_mp | Mobilní platba - M-platba | Mobilní telefon - platební brána operátora |
| | | |
| cz_kb | Platba KB – Mojeplatba - platební tlačítko | Internetové bankovníctví Komerční banky a.s. |
| cz_rb | Platba RB – ePlatby - platební tlačítko | Internetové bankovníctví Raiffeisenbank a.s. |
| cz_mb | Platba mBank – mPeníze - platební tlačítko | Internetové bankovníctví MBank |
| cz_fb | Platba Fio Banky – platební tlačítko | Internetové bankovníctví Fio banky |
| sk_uni | Platba UniCredit Bank - uniplatba - platební tlačítko | Internetové bankovníctví UniCredit Bank a.s. |
| sk_sp | Platba SLSP - sporopay - platební tlačítko | Internetové bankovníctví Slovenská sporiteľňa, a. s. |
| | | |
| eu_bank | Bankovní převod | Běžný bankovní převod – GoPay sestavuje instrukce pro provedení platby. |
| | | |
| eu_gp_w | GoPay peněženka | Elektronická peněženka. |

13.11 Stav platby

Platba na straně GoPay prochází od vytvoření níže popsanými stavy.

| Kód | Popis stavu |
|-----------------------|--|
| CREATED | Platba vytvořena. |
| PAYMENT_METHOD_CHOSEN | Zákazník provedl výběr platební metody. V případě online platebních metod byl zároveň přesměrován na platební bránu poskytovatele platební metody. |
| CANCELED | Platba byla zrušena na platební bráně GoPay nebo z platební brány poskytovatele platební metody. |
| TIMEOUTED | Životnost platby vypršela. |
| PAID | Platba byla provedena. Peněžní prostředky byly připsány na obchodní účet. |
| AUTHORIZED | Autorizace platby byla úspěšně dokončena. Prostředky lze pomocí 7.1.5 převést na obchodní účet v rámci GoPay. |
| REFUNDED | Platba byla vrácena zákazníkovi. |



13.12 Doplnující stavy

Doplňující stavy slouží k úplnějšímu popisu situace. Tyto stavy by měli sloužit pouze k vizualizaci stavu platby. Není vhodné na jejich základě měnit stav objednávky. S přibývajícimi platebními metodami a funkcionalitami budou stavy doplňovány.

| Kód | Popis stavu |
|------------|---|
| 101 | K úhradě vybrána online platební metoda. |
| 102 | K úhradě vybrána offline platební metoda. Zákazník obdržel informační email s instrukcemi pro provedení platby. |

13.13 Doba životnosti platby

Doba životnosti platby definuje interval během něhož může dojít k provedení platby, nebo jejímu zrušení. Po ukončení životnosti platby je provedena změna stavu platby na TIMEOUTED.

Při opětovném přesměrování zákazníka na platební bránu GoPay neumožňujeme opakovaný výběr platební metody bez ohledu na její životnost.

V případě, že je platba provedena po limitu životnosti platby, potom je vrácena odesilatel.

V níže uvedené tabulce jsou uvedeny doby životnosti pro jednotlivé platební metody. Zároveň, pro orientaci, uvádíme průměrné doby potřebné pro provedení platby.

| Platební metoda | Životnost platby | Průměrná doba provedení platby |
|-----------------|---|---|
| SUPERCASH | Není definována. SuperCASH čárový kód lze uhradit kdykoliv. | 24 hodin |
| eu_pr_sms | 1 hodina od vytvoření platby | 120s |
| cz_mp | 1 hodina od vytvoření platby | 180s |
| cz_kb | 14 dnů | 140s |
| cz_rb | 14 dnů | 180s |
| cz_mb | 14 dnů | 100s |
| cz_fb | 14 dnů | 180s |
| sk_uni | 14 dnů | 180s |
| sk_sp | 14 dnů | 180s |
| eu_bank | 14 dnů | 1hodina v rámci stejné banky, 1-2 dny ostatní |
| eu_gp_w | 1 hodina od vytvoření platby | 30s |
| eu_gp_u | 1 hodina od vytvoření platby | 180s |

13.14 Kód země

Kódy země jsou 3 znakové a odpovídají standardu ISO 3166-1 alpha-3. Příklad několika zemí je uveden níže. GoPay API obsahují zdrojové kódy s číselníkem zemí.

| Kód země | Název země |
|----------|-----------------|
| CZE | Česká republika |
| SVK | Slovensko |
| SWE | Švédsko |
| SWI | Švýcarsko |



13.15 Číselník měn

Číselník měn podle ISO 4217.

| Kód měny | Označení měny | Lokalita používání měny |
|----------|---------------|-------------------------|
| CZK | Česká koruna | Česká republika |
| EUR | Euro | Vybrané evropské státy |

13.16 Kód jazyka

Kód jazyka pro nastavení jazykové mutace platební brány.

| Kód jazyka | Jazyk |
|------------|------------|
| CS | Čeština |
| EN | Angličtina |

13.17 Podpis komunikačních elementů

Šifrovaný podpis je vytvořen následujícím postupem.

- řetězec pro podpis – sestavení řetězce je definováno viz jednotlivé komunikační elementy
- HASH SHA-1 – `hashedInput = hash(řetězec pro podpis)`;
- bin2HEX – `toCipherInput = bin2hex(hashedInput)`;
- TripleDES/ECB/NoPadding cipher – `cipheredInput = tripleDES(toCipherInput, secureKey)`;
- `encryptedSignature = bin2Hex(cipheredInput)`;

Proměnná *encryptedSignature* obsahuje podpis, který je součástí informačních elementů viz výše.

13.17.1 Vzorové implementace

V tabulce (viz níže) naleznete odkazy na sestavení podpisu informačních elementů v jednotlivých Ruby on Rails programových prostředích.

| Prostředí | sestavení podpisu |
|---------------|--|
| PHP | viz 16.1, pro sestavení podpisu informačního elementu jsou připraveny funkcionality ve třídě <i>GopayHelper</i> . Funkce jsou pojmenovány s použitím konvence viz níže a jsou doplněny o komentáře popisující jejich užití. např: <i>GopayHelper::concatPaymentCommand()</i> - sestavení řetězce pro podpis <i>GopayHelper::hash()</i> - sestavení hash <i>GopayHelper::encrypt()</i> - šifrování |
| Java | viz 16.2, pro sestavení podpisu informačního elementu jsou připraveny funkcionality ve třídě <i>GoPayHelper</i> např: <i>GoPayHelper.signEPaymentCommand(EPaymentCommand c, String key)</i> . |
| ASP.NET | viz 16.4, pro sestavení podpisu informačního elementu jsou připraveny funkcionality ve třídě <i>GopayHelper</i> . Funkce jsou pojmenovány s použitím konvence viz níže a jsou doplněny o komentáře popisující jejich užití. např: <i>GopayHelper.ConcatPaymentCommand()</i> - sestavení řetězce pro podpis <i>GopayHelper.Hash()</i> - sestavení hash <i>GopayHelper.Encrypt()</i> - šifrování |
| Python | viz 16.3, pro sestavení podpisu informačního elementu jsou připraveny funkcionality ve skriptu <i>GoPayHelper.py</i> např: <i>encrypt, decrypt, hash, sign</i> . |
| Ruby on Rails | viz 16.5, pro sestavení podpisu informačního elementu jsou připraveny funkcionality ve skriptu <i>GoPayHelper.rb</i> např: <i>encrypt, decrypt, hash, sign</i> . |

14 Vzorové implementace

GoPay připravilo pro jednotlivé programovací jazyky vzorové implementace E-shopu. Jde o velmi jednoduché prostředí, kde je demonstrováno vytvoření platby, zpracování přesměrování zpět na E-shop a zpracování notifikace o změně stavu platby.

| Prostředí | sestavení podpisu |
|---------------|---|
| PHP | viz 16.1 resp. gopay-php_2.3.zip |
| Java | viz 16.2 pro prostředí Java není připraven kompletní E-shop, ale v rámci třídy <i>SimplePayment</i> jsou k dispozici veškeré funkcionality potřebné pro komunikaci s GoPay platební bránou. |
| ASP.NET | viz 16.4 resp. gopay-asp-net_v_2_0.zip |
| Python | viz 16.3 pro prostředí Python není připraven kompletní E-shop, ale v rámci třídy <i>SimplePayment</i> jsou k dispozici veškeré funkcionality potřebné pro komunikaci s GoPay platební bránou. |
| Ruby on Rails | viz 16.5, pro prostředí Ruby on Rails není připraven kompletní E-shop, ale v rámci třídy <i>SimplePayment</i> jsou k dispozici veškeré funkcionality potřebné pro komunikaci s GoPay platební bránou. |



15 Kontaktní údaje

Technické informace, integrace

integrace@gopay.cz

M | +420 724 668 345

Podpora, provozní otázky

podpora@gopay.cz

M | +420 387 685 160

Smluvní podpora

podnikej@gopay.cz

16 Přílohy

16.1 PHP

Následující skripty naleznete v příloze *gopay-php-api_2.3.zip*.

| Skript | Složka | Popis |
|-------------------------------------|---------------|--|
| GoPayConfig gopay_config.php | api/ | Konfigurační třída definuje, zda API komunikuje s testovacím nebo produkčním prostředím. |
| GoPayHelper gopay_helper.php | api/ | Pomocné funkce pro <ul style="list-style-type: none"> • sestavování řetězců pro podpis komunikačních elementů • šifrování-dešifrování • kontrolu podpisů elementů pro zpracování • vytváření platebního formuláře/odkazu Předpokladem je verze PHP 5.1.2 a vyšší, modul mcrypt, mhash. |
| GoPaySoap gopay_soap.php | api/ | Pomocné funkce pro komunikaci pomocí GoPayWS - modul soap. Předpokladem je verze PHP 5.1.2 a vyšší, modul soap, modul open_ssl a nast. vlastnost allow_url_fopen = On. |
| CountryCode country_code.php | api/ | Pomocná třída - obsahuje číselník validních kódů zemí. |
| PaymentMethod payment_method.php | api/ | Pomocná třída - popisuje platební metodu (název, logo, kód). List platebních metod lze získat pomocí GoPayWS - GoPaySoap::paymentMethodList(). |
| Vzorové užití API | example/* | Vzorové užití API <ul style="list-style-type: none"> • pomocné třídy order.php, config.php • vytvoření platby, kontrola stavu platby |
| config.php | example/ | Konfigurace eshopu <ul style="list-style-type: none"> • nastavení EshopGoID, secureKey • URL eshopu • TEST/Provoz GoPay |
| order.php | example/ | Pomocná třída pro přístup k objednávce, resp. jejím parametrům. |
| payment.php | example/soap/ | Vzorová implementace vytvoření platby formou úplné integrace. |
| callback.php | example/soap/ | Vzorová implementace návratu na E-shop po provedení/neprovedení platby. |
| notify.php | example/soap/ | Vzorová implementace zpracování HTTP notifikace. |



Vzorový příklad E-shopu *gopay-php_2.3.zip*.

| Složka | Popis |
|---|--|
| api/ | GoPay PHP API |
| example/config.php example/order.php | Konfigurace E-shopu, podpůrná třída pro popis objednávky |
| example/soap | Skripty pro vytvoření platby, kontrolu stavu platby, atd |
| example/view_pages/ | Prezentační vrstva |
| example/images | Obrázky |
| /* | Podpůrné skripty/objekty |

Aktuální PHP API podporuje funkcionality před-autorizace a opakovaných plateb. Veškeré chybové stavy jsou aktuálně předávány pomocí PHP Exceptions.

16.2 Java API

Balíček s Java API má následující strukturu. Aktuální verze Java API gp-java-api-2.3.zip.

| Složka | Popis |
|---------------------|---|
| src/* | Zdrojové kódy Java API. |
| javadoc/* | Dokumentace zdrojových kódů |
| dep/* | Zavislosti API na externích knihovnách. |
| gp-java-api-2.3.jar | Java API - verze 2.3 |

Struktura Java API

| Třída | Package | Popis |
|------------------------|------------------------|---|
| * | cz.gopay.api.v2 | Podpůrné třídy |
| GoPayHelper | cz.gopay.api.v2.helper | Pomocní třída pro: <ul style="list-style-type: none"> • sestavování řetězců pro podpis komunikačních elementů • šifrování/dešifrování • kontrola podpisu komunikačních elementů |
| CryptoHelper | cz.gopay.api.v2.helper | Pomocná třída pro šifrování/dešifrování. |
| AxisEPaymentProviderV2 | cz.gopay.api.v2.axis | Rozhraní definuje funkcionality webové služby GoPay. |
| * | cz.gopay.api.v2.axis | Implementace klientské části WS. |
| Vzorové užití API | | |
| SimplePayment | cz.gopay.example | Vzorová implementace <ul style="list-style-type: none"> - vytvoření platby - list platebních metod - sestavení přesměrování na platební bránu - zpracování návratu na platební bránu - zpracování HTTP notifikace |
| RecurrentPayment | cz.gopay.example | Vzorová implementace <ul style="list-style-type: none"> - vytvoření opakované platby - sestavení přesměrování - zpracování návratu na E-shop - zpracování HTTP notifikace - zpracování HTTP notifikace následné platby |

16.3 Python API

Balíček s Python API má následující strukturu. Aktuální verze Python API gopay-python_2.3.zip.

| Složka | Popis |
|-----------|---|
| gopay/api | GoPay Python API |
| example | Skripty pro vytvoření platby, kontrolu stavu platby |

Struktura Python API

| Třída | Složka | Popis |
|---|---------------------------------|--|
| PaymentStatus | api/GopayHelper.py | Informace o stavu platby |
| PaymentResult | api/GopayHelper.py | Informační element, pro informování o stavu zpracování požadavku. |
| PaymentConstants | api/GopayHelper.py | Definice URL adres a popisů stavů plateb. |
| GopayException | api/GopayHelper.py | Definice chybových stavů. |
| GopayHelper | api/GopayHelper.py | Pomocná třída pro platbu v systému GoPay <ul style="list-style-type: none"> • sestavování řetězců pro podpis komunikačních elementů • šifrování/dešifrování řetězců • verifikace podpisu informačních řetězců |
| CryptoHelper | api/CryptoHelper.py | Pomocná třída pro šifrování/dešifrování. |
| CountryCode | api/CountryCode.py | Definice kódů jednotlivých zemí. |
| VerifiedHTTPSConnection VerifiedHTTPSHandler VerifiedHTTPSTransport | api/SSLHelper.py | Třídy pro ověřování SSL certifikátu. |
| Vzorové užití API | | |
| SimplePayment | example/SimplePayment.py | Vzorová implementace <ul style="list-style-type: none"> - vytvoření platby - list platebních metod - sestavení přesměrování na platební bránu - zpracování návratu na platební bránu |
| RecurrentPaymentGPE | example/RecurrentPaymentGPE.py | Vzorová implementace <ul style="list-style-type: none"> - vytvoření opakované platby platební metodou Global Payments |
| RecurrentPaymentCSAS | example/RecurrentPaymentCSAS.py | Vzorová implementace <ul style="list-style-type: none"> - vytvoření opakované platby platební metodou 3D secure České spořitelny |
| RefundPayment | example/RefundPayment.py | Vzorová implementace <ul style="list-style-type: none"> - založení platby |



| | | |
|---------------------|--------------------------------|---|
| | | - její castecna refundace |
| GetAccountStatement | example/GetAccountStatement.py | Vzorová implementace - stazeni vypisu pohybu na uctu |

16.4 ASP.NET

Následující kód naleznete v příloze gopay-asp-net_v_2_0.zip.

| Složka/soubor | Popis |
|---------------|--|
| GoPay | Zdrojový kód GoPay.dll – ve formě projektu VisualStudio. |
| GoPayEshop | Z gopay-asp-net-2.0.zip zdrojový kód vzorového E-shopu – ve formě projektu VisualStudio. |
| GoPay.dll | Knihovna pro komunikaci s GoPay. |

| Třída | Namespace | Popis |
|---------------|------------------|--|
| GopayHelper | GoPay.api | Pomocné funkce pro <ul style="list-style-type: none"> • sestavování řetězců pro podpis komunikačního elementu • šifrování-dešifrování • kontrolu vytvořené platby |
| GopayHelperWS | GoPay.api | Pomocné funkce pro komunikaci se systémem GoPayWS. |
| GopayWS | GoPay.api | Stub pro komunikaci s GoPayWS |
| GopayConfig | GoPay.api | Konfigurační třída definuje, zda API komunikuje s testovacím nebo produkčním prostředím. |
| Payment | GopayEshop.gopay | Vzorová implementace vytvoření platby formou úplné integrace. |
| Callback | GopayEshop.gopay | Vzorová implementace návratu na E-shop po provedení/neprovedení platby. |
| Notify | GopayEshop.gopay | Vzorová implementace zpracování HTTP notifikace. |
| Config | GopayEshop.gopay | Konfigurace eshopu <ul style="list-style-type: none"> • nastavení EshopGoID, secureKey • URL eshopu |

16.5 Ruby on Rails API

Balíček s Ruby on Rails API má následující strukturu. Aktuální verze Ruby on Rails API gopay-python_2.4.zip.

| Složka | Popis |
|-----------|---|
| gopay/api | GoPay Ruby on Rails API |
| example | Skripty pro vytvoření platby, kontrolu stavu platby |

Struktura Ruby on Rails API

| Třída | Složka | Popis |
|--|------------------------------------|--|
| GopayHelper | api/GopayHelper.rb | Pomocná třída pro platbu v systému GoPay <ul style="list-style-type: none"> • sestavování řetězců pro podpis komunikačních elementů • šifrování/dešifrování řetězců • verifikace podpisu informačních řetězců |
| CryptoHelper | api/CryptoHelper.rb | Pomocná třída pro šifrování/dešifrování. |
| CountryCode | api/CountryCode.rb | Definice kódů jednotlivých zemí. |
| VerifiedHTTPSConnection VerifiedHTTPSHandler VerifiedHTTPSTransport | api/GopayCommunicator. rb | Třídy pro ověřování SSL certifikátu. |
| EremoteData EcustomerData EpaymentCommand EpaymentStatus EpaymentSessionInfo EpaymentIdentity EpaymentResult ErecurrenceRequest ERefundRequest | Api/Types.rb | Pomocné třídy |
| Vzorové užití API | | |
| SimplePayment | example/SimplePayment. rb | Vzorová implementace <ul style="list-style-type: none"> - vytvoření platby - sestavení přesměrování na platební bránu - zpracování návratu na platební bránu - zpracování http notifikace |
| RecurrentPaymentGPE | example/RecurrentPaym entGPE.rb | Vzorová implementace <ul style="list-style-type: none"> - vytvoření opakované platby platební metodou Global Payments |



| | | |
|----------------------|---------------------------------|--|
| RecurrentPaymentCSAS | example/RecurrentPaymentCSAS.rb | Vzorová implementace - vytvoření opakované platby platební metodou 3D secure České spořitelny |
| RefundPayment | example/RefundPayment.rb | Vzorová implementace - založení platby - její částečná refundace |
| GetAccountStatement | example/GetAccountStatement.rb | Vzorová implementace - stážení výpisu pohybu na účtu |