

ІНТЕГРАЦІЙНИЙ ПРОТОКОЛ SSI ECR JSON

Цей документ є власністю ТОВ "Сервус Системз Інтегрейшн" (Servus Systems Integration LLC.) і інформація, що міститься в ньому, є конфіденційною. Без письмового дозволу правовласника інформація, що міститься в цьому документі, як загалом, так і частково не може бути відтворена, включена до складу іншого документа, передана третій особі, перероблена будь-яким чином і використана з метою, відмінною від тієї, для якої її було розроблено та надано; якщо не обумовлено інакше в угоді між "Сервус Системз Інтегрейшн" і третіми особами.

ВЕРСІЇ ДОКУМЕНТУ

Версія	Дата	Опис змін
1.0.0	04/07/2023	Створення документу
1.0.3	03/11/2023	Оновлення опису методів
1.0.3	11/12/2023	Оновлення опису методів
1.0.4	21/12/2023	Оновлення оформлення
1.0.5	11/01/2024	Внесено зміни в загальний опис
1.0.6	30/01/2024	Додано статус S07 до п.4.3.4
1.0.7	09/02/2024	Внесено уточнення до метода Interrupt
1.0.8	16/02/2024	Внесено уточнення до п.5.5.1
1.0.9	22/02/2024	Внесено виправлення до п.5.2.6
1.0.10	28/02/2024	Внесено уточнення до п.5.1 та додано два нових методи для друку, метод для отримання інфо про термінал. Порт по замовчуванню змінено на 3000.
1.1.0	21/03/2024	<p>В методі GetMerchantList змінені ключі: MerchantList -> merchantList, MerchantCount -> merchantCount; в GetLastReceipt змінений ключ: Receipt -> receipt.</p> <p>Додана секція "params" до наступних методів: GetStatus, GetLastReceipt, GetLastReport, GetMerchantList</p> <p>В метод GetLastReport доданий ключ isBatchEmpty який вказує на наявність/відсутність транзакцій</p>
1.1.1	08/04/2024	Внесено уточнення до п. 4.1, 4.3.2, 4.3.3, 5.1, 5.5.4, 5.5.6, виправлено помилки в блок-схемах в п.3
1.1.2	17/04/2024	Додано метод RestartPaymentApp
1.1.3	13/05/2024	Додано метод PartialVoid
1.1.4	30/05/2024	Додано параметр PAR в метод GetLastResult
1.1.5	22/06/2024	Оновлено ключі методів друку та помилок
1.1.6	31/07/2024	До методу PrintXML додано ключ "fontStyle" - файл шрифту, яким потрібно надрукувати текст
1.1.7	07/08/2024	До методу Purchase додано опціональний ключ discountedAmount
1.1.8	27/09/2024	Додано метод Restart
1.1.9	25/11/2024	Додано розділ про обробку реверсалів
1.2.0	11/12/2024	Уточнення по статусу S03

1.2.1	26/12/2024	<p>До методу GetLastReport Додано ключ "binName",</p> <p>До методу GetLastResult додано ключі "errorDetails", "transactionResult", "posEntryMode", "balance"</p> <p>До опису методу PingDevice внесено додатковий ключ "params"</p> <p>Видалено зайвий ключ authCode з методу "SalePreauth"</p> <p>Додана рекомендація щодо часу очікування відповіді на запити (п.1.1)</p> <p>Уточнено категорію для помилки E00, додано нові коди помилок: E19, E20, E21 (розділ ЗНАЧЕННЯ КЛЮЧА errorCode)</p> <p>Додано ключ "details" до таблиці "Обов'язкові ключі" та до п.5.1</p> <p>Додано ключ "params" до пунктів 5.1, 5.2.1.2, 5.2.11, 5.4.1, 5.4.2, 5.5.7, 5.5.8, 5.5.10</p> <p>Додано ключ "terminalId" до методу Cashback та опис цього ключа</p> <p>Додано ключ "discount" до методу GetLastResult</p>
1.2.2	30/12/2024	Розширено опис розділу "Фізичне підключення"
1.2.3	31/12/2024	<p>До таблиці «КЛЮЧІ ПАРАМЕТРІВ» додано опис ключів transactionResult, posEntryMode</p> <p>Уточнено назву методу GetLastReportReceipt в п.2.5. ДОДАТКОВІ ОПЕРАЦІЇ</p> <p>До табл. 4.3.2 КЛЮЧІ ПАРАМЕТРІВ додано опис ключа "details", "errorDetails"</p>
1.2.4	28/01/2025	Додано новий ключ та коментар до п. 5.2.3. ОПЕРАЦІЯ ПОВЕРНЕННЯ "REFUND"
1.2.5	30/01/2025	Додано нові ключі до операції поповнення DEPOSIT
1.2.6	31/01/2025	Додано новий ключ "transactionUid" до платіжних методів та опис ключа "step" для двох-прохідних операцій, уточнення прикладу LRC, зміни у оформленні, корекція назви ключа коду авторизації
1.2.7	27/02/2025	Корекція списку ключів відповіді по методу Balance
1.2.8	03/03/2025	Додано метод getResultByUid для отримання результату операції по зовнішньому ідентифікатору
1.2.9	06/03/2025	Додано статус S08 до табл. ЗНАЧЕННЯ КЛЮЧА STATUS, уточнено перелік статусів в розділі ОБРОБКА ІНШИХ МЕТОДІВ; додано ключ cashAmount та додаткові роз'яснення до методів GetLastResult, GetResultByUid; виправлено помилку в описі розрахунку Irc в розділі «ФОРМАТ ПОВІДОМЛЕНЬ»; додано додаткові роз'яснення до методу Restart; додано коментарі до ключів parEnabled, parValue в розділі «КЛЮЧІ ПАРАМЕТРІВ»; виправлено помилку в описі метода PrintXML: змінено назву ключа fontStyle на fontName та змінено місце його розташування в запиті
1.3.0	13/03/2025	Корекція помилок написання роздільних символів ключів та значень
1.3.1	26/03/2025	Додано метод readDiscountCard та getMerchantListDetailed , ключ merchantName
1.3.2	01/04/2025	Уточнення методу readDiscountCard та getMerchantListDetailed

1.3.3	21/03/2025	Розширено список дозволених статусів для методу Interrupt, ключ transactionUID замінено на transactionUid, уточнено назву методу getMerchantListDetailed у відповіді терміналу, замінено усі некоректні лапки у прикладах запитів та відповідей "" на коректні ""
1.3.4	12/05/2025	Додано шифрування ключа track2 за алгоритмом RSA/ECB/OAEPWithSHA-256AndMGF1Padding ключом довжиною не менш 2048 біт. Додано новий метод GetPinCode для отримання ПІН коду паливних або карт лояльності тощо. Додано метод GetOtpCode для введення одноразовий кодів, номеру телефону, коду документа з Дія та інших аналогічних цифрових ідентифікаторів для підтримки програм лояльності, паливних карт тощо. Додано відповідні ключі pinBlock та otpCode да методу GetLastResult
1.3.5	08/07/2025	Виправлено синтаксичні помилки у прикладах коду та внесені уточнення для методів GetPinCode та GetOtpCode . Виправлено помилку в описі розрахунку lrc для поля "receipt" в методі PrintXML . Розширено опис функціоналу методу PrintPdfUrl . Видалено зайвий ключ transCurrency з методу ReadDiscountCard . В таблицю ключів параметрів розширено опис для ключа track2 .
1.3.6	14/07/2025	Додано новий метод P2PCredit для операцій поповнення карток всіх емітентів (для SmartVista-Ukrcard)
1.3.7	28/07/2025	Розширено опис ключа pinPemFileName в методі GetPinCode
1.3.8	30/07/2025	Виправлено помилку в описі ключа pinBlock в методі GetPinCode
1.3.9	01/08/2025	Розширено опис методів GetLastReportReceipt , GetLastReceipt , PrintXML
1.4.0	10/08/2025	Уточнення щодо назв опціональних ключів у Deposit та p2pCredit методах
1.4.1	13/08/2025	Додано метод GetMultiPosBankListDetailed для мульти-еквайрингового функціоналу
1.4.2	17/10/2025	Додано опис використання шрифтів для методу printXML
1.4.3	16/01/2026	Додано опціональний ключ splitData для опису потрібного розщеплення платежу
1.4.4	21/01/2026	Додано опціональний ключ noFeed до методу printXml , корекція синтаксису splitData
1.4.5	14/02/2026	Додано роз'яснення по splitData , додано значення E22 для errorCode , яка відповідає за ідемпотентність методів при використанні transactionUid

Зміст

1.1. ФІЗИЧНЕ ПІДКЛЮЧЕННЯ	8
1.2. КОДУВАННЯ	8
1.3. ФОРМАТ ПОВІДОМЛЕНЬ.....	9
2.1. БЕЗГОТІВКОВІ ФІНАНСОВІ ОПЕРАЦІЇ	10
2.2. РЕВЕРСАЛ БЕЗГОТІВКОВИХ ФІНАНСОВИХ ОПЕРАЦІЙ	11
Таблиця стадій реверсалу.....	11
Параметри відправки реверсалу	12
2.3. ОТРИМАННЯ ЗВІТІВ	13
2.4. СЕРВІСНІ ОПЕРАЦІЇ.....	13
2.5. ДОДАТКОВІ ОПЕРАЦІЇ.....	13
3. СХЕМА ВЗАЄМОДІЇ.....	15
3.1. ОБРОБКА БЕЗГОТІВКОВИХ ОПЕРАЦІЙ	15
3.2. 2-Х КРОКОВА ОБРОБКА БЕЗГОТІВКОВИХ ОПЕРАЦІЙ	16
3.3. ОБРОБКА ІНШИХ МЕТОДІВ	17
4.1. ФОРМАТ ЗНАЧЕНЬ ДАНИХ	20
4.2. ТИПИ КЛЮЧІВ.....	20
4.3. ОПИС КЛЮЧІВ	21
4.3.1 ЗАГАЛЬНІ КЛЮЧІ	21
4.3.2 КЛЮЧІ ПАРАМЕТРІВ.....	22
4.3.3. ЗНАЧЕННЯ КЛЮЧА errorCode.....	29
4.3.4. ЗНАЧЕННЯ КЛЮЧА status	31
4.3.5. Основні КЛЮЧІ запитів-відповідей	31
5.1 ВІДПОВІДЬ НА НЕКОРЕКТНИЙ ЗАПИТ	33
5.2. БЕЗГОТІВКОВІ ФІНАНСОВІ ОПЕРАЦІЇ	34
5.2.1. ОПЕРАЦІЯ ОПЛАТИ "PURCHASE"	34
5.2.2. СКАСУВАННЯ ОПЕРАЦІЇ "VOID"	35
5.2.3. ОПЕРАЦІЯ ПОВЕРНЕННЯ "REFUND"	36
5.2.4. ОПЛАТА З ВИДАЧЕЮ ГОТІВКИ "CASHBACK"	37
5.2.5. ПРЕДАВТОРИЗАЦІЯ "PREAUTHORIZATION"	38
5.2.6. ЗАВЕРШЕННЯ ПРЕДАВТОРИЗАЦІЇ "SALEPREAUTH"	39
5.2.7. СКАСУВАННЯ ПРЕДАВТОРИЗАЦІЇ "VOIDPREAUTH"	39

5.2.9. ПОПОВНЕННЯ РАХУНКУ "P2PCREDIT"	40
5.2.8. ВИДАЧА ГОТІВКИ "CASH"	41
5.2.9. ПОПОВНЕННЯ РАХУНКУ "DEPOSIT"	42
5.2.10. ЗАПИТ БАЛАНСУ "BALANCE"	42
5.2.11. ЕХО ТЕСТ "ECHO"	43
5.2.12. ЧАСТКОВЕ СКАСУВАННЯ ОПЕРАЦІЇ "PARTIALVOID"	44
5.2.13. ЗЧИТУВАННЯ ДИСКОНТНОЇ АБО ПАЛИВНОЇ КАРТКИ "READDISCOUNTCARD"	45
5.2.14. ЗЧИТУВАННЯ ПІН КОДУ ДИСКОНТНОЇ АБО ПАЛИВНОЇ КАРТКИ "GETPINCODE"	46
5.2.15. ЗЧИТУВАННЯ цифрового коду "GETOTPCODE"	47
5.3. ЗВІТИ	50
5.3.1. Z-ЗВІТ	50
5.3.2. X-ЗВІТ	50
5.4. СЕРВІСНІ ОПЕРАЦІЇ	51
5.4.1. ПОТОЧНИЙ СТАТУС ТЕРМІНАЛА "GETSTATUS"	51
5.4.2. ПЕРЕРИВАННЯ ПОТОЧНОЇ ОПЕРАЦІЇ "INTERRUPT"	51
5.4.3. ПЕРЕЗАВАНТАЖЕННЯ ТЕРМІНАЛУ "RESTART"	52
5.5. ДОДАТКОВІ ОПЕРАЦІЇ	53
5.5.1. ОТРИМАННЯ РЕЗУЛЬТАТУ ОСТАННЬОЇ ОПЕРАЦІЇ "GETLASTRESULT"	53
5.5.2. ОТРИМАННЯ РЕЗУЛЬТАТУ ОПЕРАЦІЇ по зовнішньому ідентифікатору "GETRESULTBYUID"	54
5.5.2. ОТРИМАННЯ ЧЕКУ ОСТАННЬОЇ ТРАНЗАКЦІЇ "GETLASTRECEIPT"	56
5.5.3. ПЕРЕВІРКА ЗВ'ЯЗКУ З ТЕРМІНАЛОМ "PINGDEVICE"	56
5.5.4. ОТРИМАТИ РЕЗУЛЬТАТИ ПІДСУМКІВ ОСТАННЬОГО X-ЗВІТУ ЧИ ЗВІРКИ "GETLASTREPORT"	57
5.5.5. ПЕРЕЛІК ВСІХ МЕРЧАНТІВ НА ТЕРМІНАЛІ "GETMERCHANTLIST"	59
5.5.6. ПЕРЕЛІК ВСІХ МЕРЧАНТІВ НА ТЕРМІНАЛІ "GETMERCHANTLISTDETAILED"	59
5.5.7. ОТРИМАТИ ОСТАННІЙ ЧЕК X- АБО Z-ЗВІТУ "GETLASTREPORTRECEIPT"	61
5.5.8. ДРУК ФАЙЛА В XML-ФОРМАТІ "PRINTXML"	62
5.5.9. ДРУК PDF-ФАЙЛА ПО URL-ЛІНКУ "PRINTPDFURL"	64
5.5.10. ОТРИМАННЯ ДАНИХ ПО ТЕРМІНАЛУ "GETTERMINALINFO"	64
5.5.11. ПЕРЕЗАВАНТАЖЕННЯ ПЛАТІЖНОГО ЗАСТОСУНКУ "RESTARTPAYMENTAPP"	68
6.1. РОЗРАХУНОК LRC	71

ЗАГАЛЬНИЙ ОПИС

Цей документ призначений для використання інтеграторами рішень і кінцевими користувачами, які впроваджують у свої рішення функції безготівкових платежів, та описує програмний інтерфейс між POS-терміналом та програмним забезпеченням електронного касового апарату (ECR) або ПЗ іншої системи управління торгівлею тощо, який дозволяє виконувати безготівкові платіжні операції шляхом виклику методів платіжного додатку SSI SmartPOS_EMV. Інтеграційний інтерфейс використовує JSON нотацію для обміну командами та інформацією між додатками.

Інтеграційний протокол є незалежним від платформи та операційної системи, не потребує додаткових компонент та працює з фізичними інтерфейсами підключення POS-терміналу по TCP-IP або HTTP сокету, а також кабельним з'єднанням по USB до зовнішньої системи/зовнішнього або внутрішнього (розміщеного на самому POS-терміналі) додатку.

1. БАЗОВІ ХАРАКТЕРИСТИКИ ІНТЕГРАЦІЙНОГО ПРОТОКОЛУ

1.1. ФІЗИЧНЕ ПІДКЛЮЧЕННЯ

Підключення по фізичному протоколу **TCP/IP** до POS-терміналу виконується адресацією запитів на IP адресу POS-терміналу (xxx.xxx.xxx.xxx) або localhost (127.0. 0.1) для додатків на тому ж самому POS-терміналі, порт підключення **3000** за замовчуванням. Також підключення можна зробити по протоколу http, наприклад з вікна WEB браузера, на IP адресу POS-терміналу (xxx.xxx.xxx.xxx) або localhost (127.0. 0.1) для додатків на тому ж самому POS-терміналі, порт підключення **3001** за замовчуванням (+1 від номеру порту TCP/IP підключення). Порт підключення до інтеграційного сервісу за необхідністю може бути змінений шляхом звернення до вашої фінансової організації, або до сервісної організації. Якщо на POS-терміналі присутні декілька платіжних додатків різних фінансових організацій, потрібно налаштувати кожний платіжний додаток на свій індивідуальний порт.

Для підключення по фізичному протоколу **USB** до POS-терміналу шляхом з'єднання USB Type-C порту POS-терміналу з зовнішньою системою необхідно встановити на ній або драйвер порту, або драйвер POS-терміналу в залежності від операційної системи.

Підтримка того чи іншого інтерфейсу підключення залежить від конкретної моделі терміналу.

Слід враховувати, що відповіді від терміналу на запити можуть формуватися до 15 секунд.

Пауза між отриманням відповіді на поточний запит та надсиланням наступного запиту повинна становити щонайменше 0.25 секунд.

1.2. КОДУВАННЯ

Усі не-**ASCII** символи, зокрема кирилиця, передаються в кодуванні **UTF-8**.

Протокол використовує JSON нотацію для запитів/відповідей. Повідомлення, якими обмінюються POS-термінал і зовнішня система, мають у своєму складі обов'язкові та опціональні ключі в залежності від типу операції. Обов'язкові ключі мають бути присутніми в повідомленні завжди, опціональні ключі можуть як міститись в повідомленні, так і бути відсутніми. У випадку, якщо передається ключ, опис якого відсутній в даному документі – його треба вважати опціональним і не обробляти.

1.3. ФОРМАТ ПОВІДОМЛЕНЬ

```

<STX>    02 66 01
<HEADER> 00 4A
<Data>   7B 22 6D 65 74 68 6F 64 22 3A 22           {"method": "
          50 69 6E 67 44 65 76 69 63 65 22 2C 22 65 72 72 PingDevice", "err
          6F 72 22 3A 66 61 6C 73 65 2C 22 65 72 72 6F 72 or": false, "error
          43 6F 64 65 22 3A 22 22 2C 22 65 72 72 6F 72 44 Code": "", "errorD
          65 73 63 72 69 70 74 69 6F 6E 22 3A 22 22 7D  escription": ""}
<LRC>    53
  
```

Цей формат повідомлень використовується при інтеграції з касою через tcp/ip сокет або через USB/Serial підключення.

- <STX> початок повідомлення (0x02)(0x66)(0x01) 0x66 - ідентифікатор протоколу, 0x01 - версія протоколу
- <HEADER> довжина повідомлення 2 байти (спочатку старший байт, потім молодший, приклад у Binary hex format: 16, байт: 00 10). ⚡ **Довжина рахується тільки для розділу DATA.**
- <Data> дані JSON повідомлення. ⚡ **Розмір секції даних до 64К.**
- <LRC> контрольна сума повідомлення, довжина 1 байт, LRC байт генерується застосуванням операції логічного ВИКЛЮЧАЮЧЕ_АБО до всіх байтів поля DATA (не враховуються поля STX та HEADER)

2. ПЕРЕЛІК ОПЕРАЦІЙ

2.1. БЕЗГОТІВКОВІ ФІНАНСОВІ ОПЕРАЦІЇ

Назва	Метод
Оплата/Продаж	Purchase
Скасування операції	Void
Повне або часткове повернення	Refund
Оплата/продаж з видачею готівки	CashBack
Предавторизація	Preauthorization
Завершення предавторизації	SalePreauth
Скасування предавторизації	VoidPreauth
Видача готівки	Cash
Поповнення рахунку	Deposit
Запит балансу	Balance
Перевірка зв'язку з процесинговим центром еквайрингової інституції	Echo
Часткове скасування операції	PartialVoid
Поповнення картки переказом P2P	P2pCredit

2.2. РЕВЕРСАЛ БЕЗГОТІВКОВИХ ФІНАНСОВИХ ОПЕРАЦІЙ

Реверсал або реверс – це передбачена протоколом ISO8583 по обміну фінансових повідомлень автоматична відміна фінансових операцій, яку виконує термінал задля збереження фінансової цілісності фінансових операцій між процесінгом та ПОС терміналом у деяких випадках, які описано нижче.

ТАБЛИЦЯ СТАДІЙ РЕВЕРСАЛУ

Стадія фінансової операції	Результат	Дія
Встановлення зв'язку з процесінгом	Позитивний Негативний	Операція помічається як «потенційний» реверсал Операція відміняється як неуспішна
Відправка даних по операції	Негативний: - Time out - Socket Exception	Операція відміняється як неуспішна Формується та планується до відправки реверсал поточної операції
Відправка даних по операції	Негативний: - Time out - Socket Exception - Message format error	Операція відміняється як неуспішна Формується та планується до відправки реверсал поточної операції
EMV online result	Негативний: - Помилка при обробці EMV ядром відповіді процесінгу	Операція відміняється як неуспішна Формується та планується до відправки реверсал поточної операції
Отримання підпису як метода верифікації	Негативний: - Відсутність підпису	Операція відміняється як неуспішна Формується та планується до відправки реверсал поточної операції
Отримання відповіді від процесінгу	Негативний:	

Стадія фінансової операції	Результат	Дія
	- код відмови від процесінгу Позитивний	Операція відміняється як неуспішна Операція успішна, помітка «потенційний» реверсал скасовується
Відсутність паперу	Негативний: - папір відсутній та закінчився таймаут очікування паперу	За умови встановленого у конфігурації терміналу правила вимагати друк чеку, формується та планується до відправки реверсал поточної операції. За умови встановлення у конфігурації терміналу правила не вимагати друк чеку (ігнорувати відсутність паперу), операція успішна, помітка «потенційний» реверсал скасовується

ПАРАМЕТРИ ВІДПРАВКИ РЕВЕРСАЛУ

Термінал може бути налаштовано банком на 2 способу відправки реверсалу:

1) *Відправка реверсалу при наступній операції*

При такому налаштуванні реверсал буде відправлятися виключно з наступною операцією, та виконання наступної операції буде заблоковано до тих пір, поки реверсал не буде успішно прийнято процесінговою системою

Зауваження: для процесінгу УПЦ подвійний неуспішний код реверсалу приведе до видалення реверсалу терміналом, але на інших процесінгах термінал очікує успішного підтвердження реверсалу.

2) *Відправка реверсалу з певною періодичністю від часу оригінальної операції*

При такому налаштуванні у системі управління терміналами можливо налаштувати часовий інтервал, який буде використано терміналом для спроби відправки реверсалу без очікування наступної операції, наприклад, кожні 60 секунд від часу оригінальної операції.

Зауваження: у випадку коли жодна зі спроб відправити реверсал по розкладу не буде мати успіху, проведення наступної операції також буде заблоковано до тих пір, поки реверсал не буде успішно прийнято процесінговою системою.

2.3. ОТРИМАННЯ ЗВІТІВ

Назва	Метод
Z звіт	ZReport
X звіт	XReport

2.4. СЕРВІСНІ ОПЕРАЦІЇ

Назва	Метод
Поточний статус терміналу	GetStatus
Переривання поточної операції (до вводу PIN коду або до зчитування платіжного засобу)	Interrupt
Перезавантаження терміналу	Restart
Перезавантаження платіжного додатку	RestartPaymentApp

2.5. ДОДАТКОВІ ОПЕРАЦІЇ

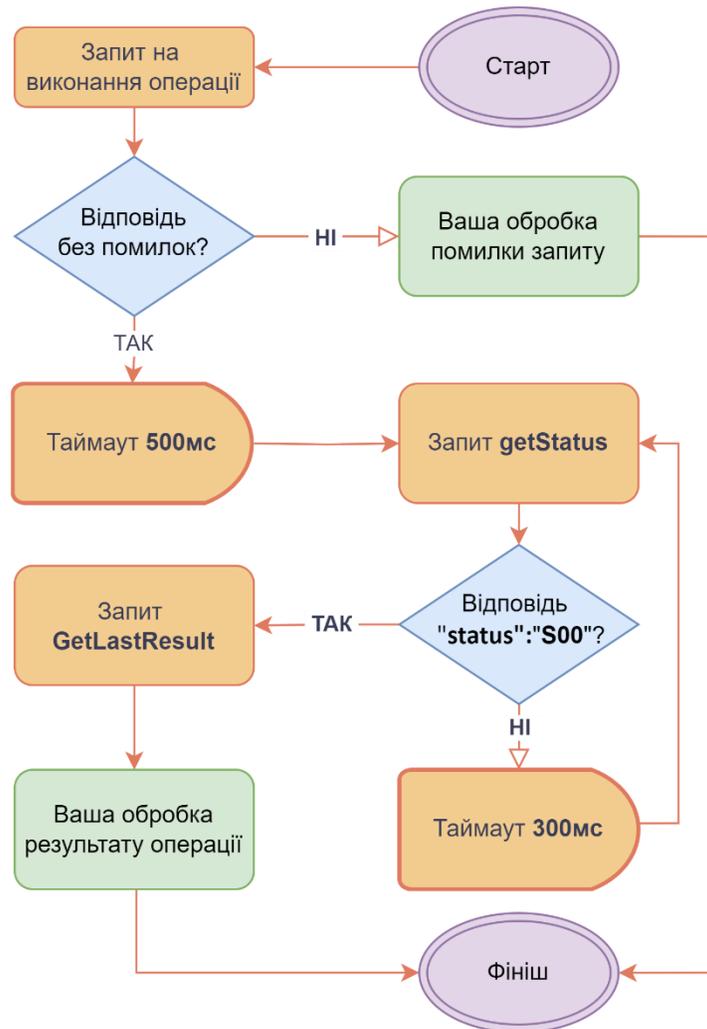
Назва	Метод
Отримати результат останньої транзакції	GetLastResult
Отримати чек останньої транзакції	GetLastReceipt
Перевірка зв'язку з терміналом	PingDevice
Отримання цифрового ідентифікатору (одноразового коду, номеру телефона тощо)	getOtpCode

Отримання ПІН коду для паливних та інших типів карт з використанням публічного RSA ключу	getPinCode
Отримати результати підсумків останнього X-звіта чи звірки	GetLastReport
Перелік всіх merchantID на терміналі	GetMerchantList
Перелік всіх торговців на терміналі з назвою	GetMerchantListDetailed
Копія останнього X- або Z-звіту	GetLastReportReceipt
Друк довільного файлу в xml-форматі	PrintXml
Друк довільного змісту файлу у Pdf-форматі який розташований за вказаним лінком	PrintPdfUrl
Отримання інформації щодо терміналу та по встановленому програмному забезпеченню на терміналі	GetTerminalInfo

3. СХЕМА ВЗАЄМОДІЇ

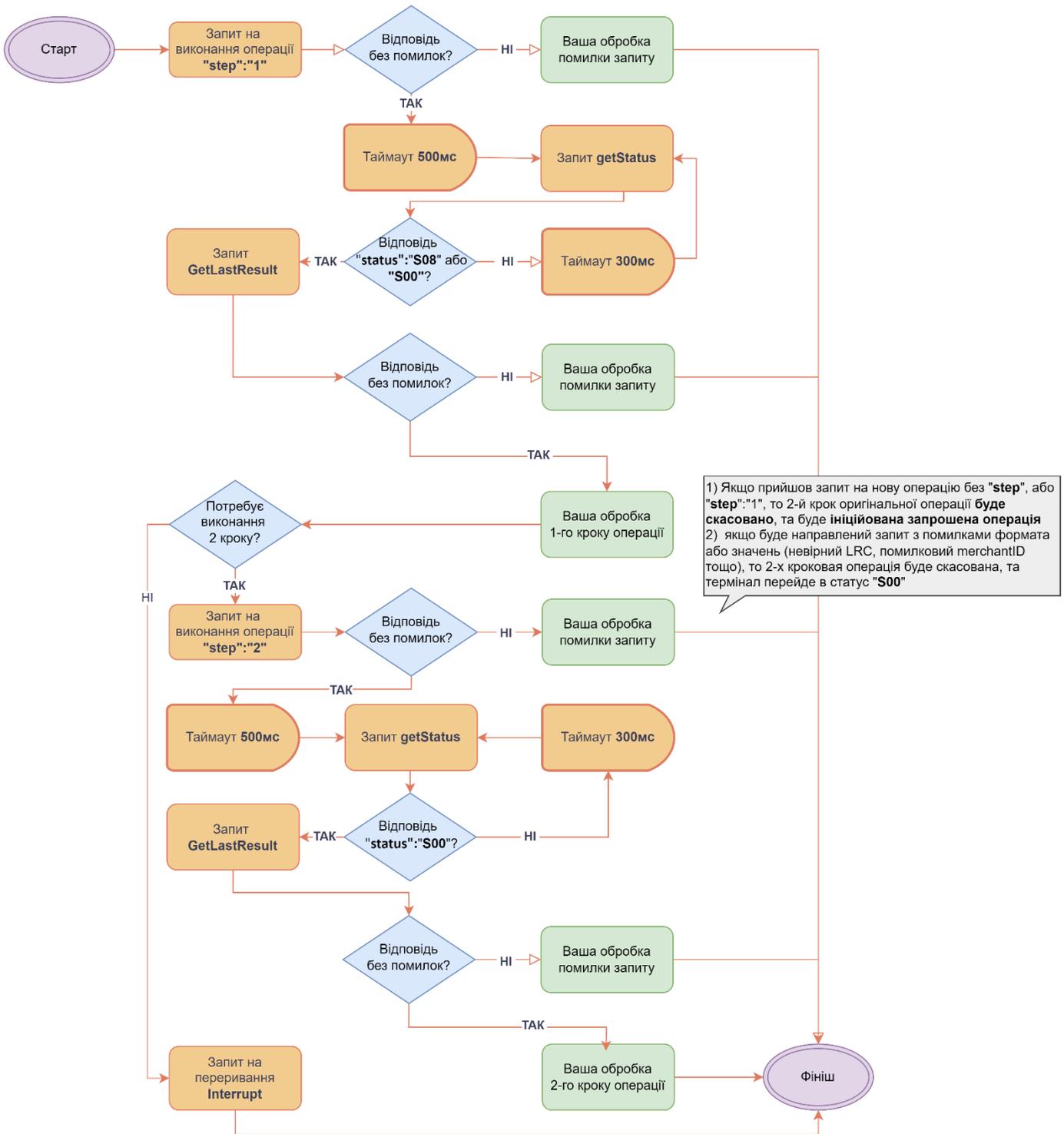
3.1. ОБРОБКА БЕЗГОТІВКОВИХ ОПЕРАЦІЙ

Методи з розділу «**Безготівкові фінансові операції**» складаються із запиту з боку каси/системи та підтвердження отримання команди з боку POS-терміналу. Результат операції зберігається в POS-терміналі і передається на касу за допомогою методу [GetLastResult](#). Контроль за проходженням операції відбувається за допомогою методу [GetStatus](#). Результат операції повертається у методі [GetLastResult](#). Загальна блок-схема виконання операцій з розділу «Безготівкові фінансові операції» виглядає так:



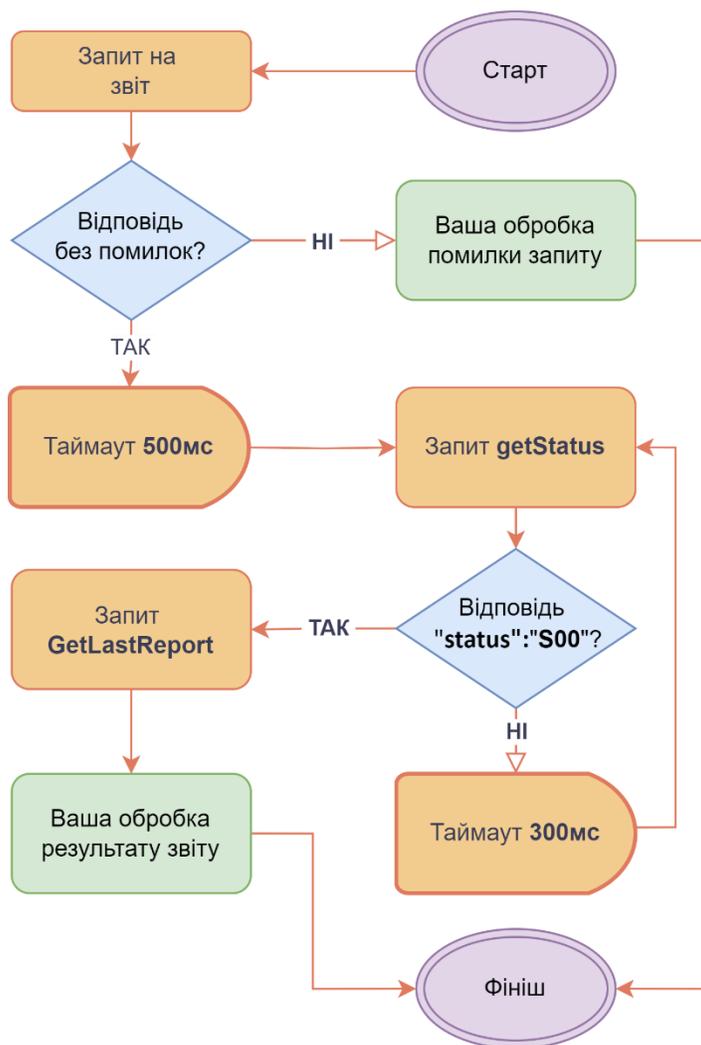
3.2. 2-Х КРОКОВА ОБРОБКА БЕЗГОТІВКОВИХ ОПЕРАЦІЙ

Методи з розділу «**Безготівкові фінансові операції**» можуть також виконуватися у двох кроковій схемі з використанням ключа "step" у запиті зі значенням 1 та 2 для відповідного кроку, коли платіжний термінал спочатку читає та виконує відповідну верифікацію платіжного засобу (введення ПІН коду тощо), після чого повертає номер платіжного засобу до касової системи (**9 перших цифр номеру карти та 1 останню цифру**), та надає можливість завершити цю операцію з іншою сумою операції, яка може відрізнятися від суми, яку було надано на першому кроці, наприклад, нарахування додаткової знижки по номеру кобрендінгової карти, або перерахунок комісії по карті партнерського банку тощо. Результат операції на кожному кроці зберігається в POS-терміналі і повертається на касу за допомогою методу [GetLastResult](#), перервати виконання 2-го кроку можливо методом [Interrupt](#) або новим запитом на іншу операцію. Загальна блок-схема 2-х крокового виконання операції виглядає так:



3.3. ОБРОБКА ІНШИХ МЕТОДІВ

Методи з розділу «Отримання звітів» працюють аналогічно методам з розділу «Безготівкові фінансові операції» за винятком того, що замість метода [GetLastResult](#) використовується метод [GetLastReport](#). Загальна блок-схема виконання методів з розділу «Отримання звітів» виглядає так:



Методи з розділу «Сервісні операції» виконуються в момент виконання операцій з розділу «Безготівкові фінансові операції» або «Отримання звітів» та складаються із запиту з боку каси та відповіді з боку POS-терміналу. Виклик метода [GetLastResult](#) для цих методів не передбачено, результат виконання повертається відразу.

Методи з розділу «Додаткові операції» виконуються в момент, коли термінал знаходиться в стані **Idle** (Status = S00 або S08) та складаються із запиту з боку каси та відповіді з боку ПО терміналу.

Виклик метода [GetLastResult](#) для цих методів не передбачено, результат виконання повертається відразу.

4. ОПИС ФОРМАТА ДАНИХ ТА КЛЮЧІВ

4.1. ФОРМАТ ЗНАЧЕНЬ ДАНИХ

Формат	Опис
a	Текстове поле, може містити тільки букви
n	Текстове поле, може містити тільки цифри
an	Текстове поле, може містити букви і цифри
ans	Текстове поле, може містити будь-які значення, кожен елемент даних поля займає 1 байт
array	Поле може містити масив даних
lst	Поле може містити список даних
boolean	Бінарне значення true/false

4.2. ТИПИ КЛЮЧІВ

Тип	Опис
M	Mandatory - обов'язкове
O	Optional - опціональне

4.3. ОПИС КЛЮЧІВ

4.3.1 ЗАГАЛЬНІ КЛЮЧІ

Ключ	Опис	Формат	Приклад
method	Назва методу	an	Purchase
step	Індикатор кроку для 2-х прохідних операцій: "1" або "2"	n	1
params	Набір параметрів. Список даних, складається з списку ключів.	lst	
receipt	Вигляд чека залежить від моделі терміналу: чек в електронному вигляді Чек в форматі JSON	ans dim	
status	Поточний статус терміналу	an	S01
error	Статус помилки	boolean	true/false
errorCode	Повідомлення про помилку. Можливі значення ключа і опис надано в п. 4.3.3.	ans	E00
errorDescription	Для методу GetLastResult - дублювання опису помилки, який надрукований на чеку. Для інших методів - опис помилки згідно з 4.3.3. Значення ключа errorCode	ans	General error

4.3.2 КЛЮЧІ ПАРАМЕТРІВ

Тей чи інший ключ може використовуватися в залежності від конкретного методу.

Ключ	Опис	Формат	Приклад
originalTrnName	Назва операції, по якій формується відповідь	a	SalePreauth
time	Час операції	ans	15:12:34
date	Дата операції	ans	07/04/2023
pan	Номер платіжного засобу	ans	4143*****9008
expiryDate	Строк дії платіжного засобу (ММYY)	ans	"0326"
cardholderName	Ім'я тримача платіжного засобу	ans	TEST CARD
firstName	Ім'я особи, як виконує платіжну операцію	ans	"Сергій"
familyName	Прізвище особи, як виконує платіжну операцію	ans	"Козак"
transAmount	Сума операції з урахуванням суми знижки*, "2450" є сумою з урахуванням копійок 24.50	n(12)	2450
endAmount	Остаточна сума операції при завершенні предавторизації*	n(12)	27000
cashAmount	Сума до видачі*	n(12)	700
discount	Сума знижки. Не використовується в поточній версії протоколу. Див. ключ transAmount *	n(12)	500
transCurrency	Код валюти	n(3)	980

Ключ	Опис	Формат	Приклад
splitData	Опис необхідного розщеплення суми операції, де ідентифікатор SubMerchantID надається оператором розщеплення	ans	SubMerchantId1=4000,SubMerchant2=3000
merchantId	ID мерчанта, яку надає юрособі еквайрингова інституція, в режимах мульти-мерчанти (декілька торговців на одному фізичному терміналі) в терміналі існує декілька різних значень	ans	SSI00002
merchantName	Назва торговця, з якою його заведено у систему управління терміналами	ans	ФОП Ковбасюк
terminalId	Ідентифікатор терміналу у еквайрингової інституції, в режимах мульти-мерчанти, мульти-термінал (декілька торговців або логічних терміналів одного торговця на одному фізичному терміналі) в терміналі існує декілька різних значень	ans	SSI00001
invoiceNum	Номер чеку в поточній зміні у POS-терміналі	n	000456
authCode	Унікальний код авторизації, який надає процесинговий центр еквайрингової інституції	an	86429E
rrn	Унікальний номер операції, який надає процесинговий центр еквайрингової інституції	an	987612344098
responseCode	Код відповіді еквайрингової інституції (код відповіді може бути як 2 символи, так і 3 символи)	an	00
track2	Track2 платіжного засобу. Може повертатися у зашифрованому вигляді алгоритмом RSA/ECB/OAEPWithSHA-	ans	4143*****9008=26092011881002000001

Ключ	Опис	Формат	Приклад
	256AndMGF1Padding публічним ключем довжиною не менш 2048 біт. Для шифрування публічний RSA ключ має бути наданий компанії SSI для використання у профілі терміналу. При відсутності в терміналі публічного RSA ключа, track2 передається в маскованому вигляді.		
bankName	Офіційна назва банку/еквайрингової компанії, яка опрацьовує цю безготівкову операцію через платіжний додаток на POS-терміналі	ans	BANK
binName	Назва BIN діапазону платіжного засобу (для використання у програмах лояльності)	an	VISA
binDiscount	Знижка/комісія налаштована для BIN діапазону (у відсотках). <i>Не використовується в поточній версії протоколу</i>	ans	10
discountedAmount	Розрахована зовнішньою системою сума з урахуванням знижки на певний діапазон карт, який заведено у профілі терміналу як діапазон зі знижкою binDiscount *	n(12)	2250
transactionUid	Унікальний ідентифікатор операції, який передає зовнішня система при ініціації такої операції, так який повертається без змін у результаті виконання операції	ans	"5467-76765"
pinBlock	Блок зашифрованих даних, який містить в собі введений ПІН код, шифрування здійснюється публічним ключем RSA, який надається інтегратором для використання у методі GetPinCode	ans	"6565F48BC54735"

Ключ	Опис	Формат	Приклад
<code>otpCode</code>	Цифровий ідентифікатор (код підтвердження, номер телефону, код документу Дія тощо), який введено користувачем на терміналі по заданому у методі <code>getOtpCode</code> шаблону.	ans	"342746"
<code>isBatchEmpty</code>	Значення true вказує на відсутність операцій в журналі транзакцій	boolean	false/true
<code>isDetailReport</code>	При значенні true сформується по транзакційний детальний звіт. По замовчуванню дорівнює false та формується звіт тільки з підсумками.	boolean	false/true
<code>transactions</code>	Масив детальних даних по транзакціям.	array	див. <code>GetLastReport</code>
<code>totalSaleCount</code>	Підсумок по кількості операцій продажу у поточному журналі операцій	n	5
<code>totalSaleAmount</code>	Підсумок по сумі операцій продажу у поточному журналі операцій*	n(12)	000000080000
<code>totalFinAuthCount</code>	Підсумок кількості операцій предавторизацій у поточному журналі операцій*	n	4
<code>totalFinAuthAmount</code>	Підсумок по кількості сум операцій предавторизацій*	n(12)	000000043000
<code>totalRefundCount</code>	Підсумок по кількості операцій повернення	n	5
<code>totalRefundAmount</code>	Підсумок по кількості сумі операцій повернення*	n(12)	000000065000
<code>totalCashCount</code>	Підсумок по кількості операцій видачі готівки	n	4

Ключ	Опис	Формат	Приклад
totalCashAmount	Підсумок по сумі операцій видачі готівки*	n(12)	000000970000
totalPaymentCount	Підсумок по кількості операцій поповнення	n	5
totalPaymentAmount	Підсумок по сумі операцій поповнення*	n(12)	000000950000
totalVoidCount	Підсумок по кількості операцій відміни	n	6
totalVoidAmount	Підсумок по сумі операцій відміни*	n(12)	000000230000
merchantCount	Кількість мерчантів в поточному профілі термінала	n	2
merchantList	Перелік MerchantID в поточному профілі термінала	an	"000000064003122", "600078380000000"
url	Рядок, який має містити URL PDF-файлу для друку	ans	"https://docs.google.com/document/d/sample.pdf"
lrc	контрольна сума ключа «receipt» для методу PrintXml (детальний опис див. в п.5.5.7)		
receipt (для методу PrintXml)	Детальний опис див. в п.5.5.7		
parEnabled	Значення true вказує на можливість роботи з параметром PAR. Використовується в терміналах X990, які працюють за протоколом UPC	boolean	false/true
parValue	Значення параметру PAR. Використовується в терміналах X990, які працюють за протоколом UPC	ans	Q1HJZ28RKA1EBL470G9XYG90R5D 3E
transactionResult	Результат виконання операції	ans	FAILED, TRY_AGAIN, TRY_AGAIN_PHONE, CARD_ERROR,

Ключ	Опис	Формат	Приклад
			MESSAGE_ERROR, NO_CONNECTION, CONNECTION_ERROR, INTERNAL_ERROR, EMV_ERROR, CONNECTION_TIMEOUT, INSUFFICIENT_BALANCE, CARD_BLOCKED, CANCELLED, UNSUPPORTED_CARD, DECLINED_ONLINE, APPROVED_ONLINE, DECLINED_OFFLINE, CARD_DECLINED_ONLINE, APPROVED_OFFLINE, KEY_ERROR, BATCH_UPLOAD_NEEDED, SIGNATURE_VERIFICATION_ERROR, COMPLETED, APPROVED, IN_PROGRESS, FORCED_DECLINED_OFFLINE, CANCELLED_BEFORE_START, PIN_ERROR, BATCH_CLOSE_NEEDED, OK, PRINTER_ERROR
posEntryMode	Інтерфейс, за яким було оброблено платіжний засіб.	ans	CONTACT_EMV, CONTACT_FALLBACK, CONTACTLESS, MAGSTRIPE, MANUAL
details	Додаткова розшифровка помилки в наборі даних	ans	
errorDetails	Додатковий опис помилки значення ключа, в якому невірно вказане значення (опціонально при E09 помилці)	ans	
Для терміналів з операційною системою Android			
securityDriver	Версія драйверц безпеки (для Android OS)	ans	"X990-V1.0.12(202111221449)"

Ключ	Опис	Формат	Приклад
"currentApp": "name"	Назва активного платіжного додатку (для Android OS)	ans	"MonoBank"
"currentApp": "packageName"	Назва пакету платіжного додатку в Андроїд системі (для Android OS)	ans	"com.paydustry.banking.monobank"
"currentApp": "versionName"	Номер версії платіжного додатку (для Android OS)	ans	"1.2.0.010"
"thirdPartyApps": "name"	Назва додатку (для Android OS)	ans	"checkbox Verifone"
"thirdPartyApps": "packageName"	Назва пакету в Андроїд системі (для Android OS)	ans	"ua.in.checkbox.verifone"
"thirdPartyApps": "versionName"	Номер версії додатку (для Android OS)	ans	"0.6.49"
"serviceApps": "name"	Сервісні додатки виробника (для Android OS)	ans	"VFSYSTEMSERVICE"
"serviceApps": "packageName"	Назва сервісного додатку (для Android OS)	ans	"com.vfi.smartpos.system_service"
"serviceApps": "versionName"	Версія сервісного додатку (для Android OS)	ans	"1.11.2"
"paymentApps": "name"	Платіжний додаток (для Android OS)	ans	"MonoBank"
"paymentApps": "packageName"	Назва платіжного додатку (для Android OS)	ans	"com.paydustry.banking.monobank"
"paymentApps": "versionName"	Версія платіжного додатку (для Android OS)	ans	"1.2.0.010"
"externalRegisterApps": "name"	Назва Інтеграційного додатку, при використанні SSI Json протоколу для інтеграції не використовується (для Android OS)	ans	"1.0.92"

Ключ	Опис	Формат	Приклад
Для терміналів з операційною системою Linux			
boot_version	Boot версія (для Linux OS)	ans	"01.00_00_000019"
devcfg_version	Версія конфігурації терміналу (для Linux OS)	ans	"SP880.3652010.014"
os_version	Версія OS Linux (для Linux OS)	ans	"9.0.3.0_20231017"
napi_api_version	Версія API (для Linux OS)	ans	"NAPI_API_V2.7.0"
pci_fw_version	Версія firmware (для Linux OS)	ans	"D_01.00_9000300"
pos_sn	Серійний номер терміналу (для Linux OS)	ans	"B9AA00176476"
noFeed	Флаг для вимикання прокрутки паперу після друку	boolean	true

* ⚡ всі суми вказуються у мінімальних грошових одиницях (у копійках) без будь-яких роздільників. У всіх інших варіантах заповнення суми, у відповіді буде надходити помилка значення суми.

4.3.3. ЗНАЧЕННЯ КЛЮЧА ERRORCODE

Код	Ключ errorDescription	Опис	Категорія
E00	General error	Загальна помилка	Запит
E01	Protocol version is not supported by the terminal	Версія протоколу не підтримується терміналом	Запит
E02	Checksum error	Помилка контрольної суми	Запит
E03	Json message format error	Помилка формату JSON повідомлення	Запит
E04	Required fields are missing	Відсутні обов'язкові поля	Запит
E05	Unknown method	Невідомий метод	Запит

Код	Ключ <code>errorDescription</code>	Опис	Категорія
E06	Terminal Busy	Термінал зайнятий іншою операцією, повторіть запит пізніше.	Запит
E07	Merchant Id is not found	Не знайдено торговця з вказаним Merchant ID	Запит
E08	Interrupt prohibited	Переривання заборонено	Запит
E09	Format error	Помилка в форматі даних	Запит
E10	Connection error	Помилка зв'язку	Операція
E11	Verification error (pin, signature)	Помилка верифікації	Операція
E12	Transaction canceled	Транзакцію скасовано	Операція
E13	MAC Error - MAC	Помилка макування [ME]	Операція
E14	Key error	Помилка/відсутність ключів	Операція
E15	Need z report	Необхідно виконати звірку	Операція
E16	Card error	Помилка зчитування карти	Операція
E17	EMV error	Помилка EMV	Операція
E18	Printer is out of paper or inoperational	В принтері немає паперу або принтер не працює	Запит
E19	Transaction amount out of limit	Сума транзакції поза лімітом	Запит
E20	No last transactions found	Відсутні останні операції	Операція
E21	General error in transaction	Загальна помилка в операції	Операція
E22	Transaction with such transactionUid is already processing	Операція з таким transactionUid вже опрацьовується	Операція

4.3.4. ЗНАЧЕННЯ КЛЮЧА STATUS

Код	Опис
S00	Термінал в режимі очікування
S01	Виконання транзакції, термінал зайнятий поточною операцією або зайнятий
S02	Термінал очікує вводу платіжного засобу
S03	Термінал очікує вводу ПІН коду або введення підпису на екрані терміналу
S04	Зв'язок з банком
S05	Друк чеку
S06	Термінал вимагає виконати Z-звіт
S07	Термінал вимагає витягнути картку з картоприймача
S08	Термінал в режимі очікування на запит другого кроку двохкрокової транзакції або будь-якої іншої команди

4.3.5. ОСНОВНІ КЛЮЧІ ЗАПИТІВ-ВІДПОВІДЕЙ

Ключ	Запит	Відповідь	Коментар
<code>method</code>	М	М	Назва методу
<code>step</code>	О	О	Індикатор багатопрохідної операції, якщо присутній ("1" – для першого кроку багатопрохідної операції, "2" – для наступного кроку тощо)
<code>params</code>	О	О	У запиті: обов'язковий для усіх операцій з розділів «Безготівкові фінансові операції» або «Отримання звітів». Для сервісних та додаткових операцій - залежить від конкретної операції. Див. в описі операції. У відповіді: обов'язковий для успішних операцій <code>GetLastResult</code> , для неуспішних транзакцій (не плутати з некоректними запитом) <code>GetLastResult</code> . Не передається для усіх інших відповідей.
<code>error</code>		М	Наявність помилки (true/false)
<code>errorCode</code>		М	Код помилки
<code>errorDescription</code>		М	Опис помилки
<code>details</code>		О	Додаткова розшифровка помилки

5. ОПИС ОПЕРАЦІЙ

В прикладах запитів та відповідей всі ключі, виділені жирним шрифтом - обов'язкові.

5.1 ВІДПОВІДЬ НА НЕКОРЕКТНИЙ ЗАПИТ

Відповідь на запит	<pre>{ "method": "Purchase", "error": true, "errorCode": "E01", "errorDescription": "Версія протоколу не підтримується терміналом", "params":{"details":{"actual: ee, expected: 66"}} }</pre>
--------------------	---

Ця відповідь формується на запити, в котрих присутні помилки як в форматі повідомлення, так і в змісті.

⚡ Якщо не змогли визначити метод, у полі **"method"** буде значення **"Unknown"**

⚡ У випадку некоректного запиту, буде відправлено відповідь з одним з кодів E00-E99 що відносяться до категорії "Запит" (див. табл. 4.3.3)

⚡ Кожен коректний виклик методу з розділу «Безготівкові фінансові операції», при якому термінал повернув **error = false**, перезапише буфер **GetLastResult** (див. приклад в п. 5.5.1.1). Кожен виклик методу з розділу «Безготівкові фінансові операції», при якому термінал повернув **error = true**, залишить буфер **GetLastResult** без змін.

⚡ Кожен коректний виклик методу з розділу "Отримання звітів", при якому термінал повернув **error = false**, перезапише буфер **GetLastReport** (див. п. 5.5.4). Кожен виклик методу з розділу «Отримання звітів», при якому термінал повернув **error = true**, залишить буфер **GetLastReport** без змін.

5.2. БЕЗГОТІВКОВІ ФІНАНСОВІ ОПЕРАЦІЇ

5.2.1. ОПЕРАЦІЯ ОПЛАТИ "PURCHASE"

5.2.1.1. ЗАПИТ НА ОПЕРАЦІЮ ОПЛАТА "PURCHASE"

Результат виконання операції **Purchase** можна отримати за допомогою **GetLastResult** після повернення терміналу в статус S00 (див. п.5.4.1)

<i>Запит</i>	<pre>{ "method": "Purchase", "step": "1", "params": { "transAmount": "24500", "transCurrency": "980", "merchantId": "000000060007176" "discountedAmount": "22300", // опціональний ключ, якщо розрахунок знижки на певні діапазони БІН номерів карт здійснюється на стороні зовнішньої системи "transactionUid": "2i89y0787300", // опціональний ключ, ідентифікатор операції на стороні зовнішньої системи "splitData": { // опціональний ключ, який містить правила розщеплення суми операцій між декількома отримувачами платежів, для формату та значень ключів звертайтеся до вашого оператора розщеплення [{"merchantId": "merchanID1", "amount": "5000"}, {"merchantId": "merchanID2", "amount": "70000"}] } } }</pre>
--------------	---

5.2.1.2 ВІДПОВІДЬ У ВИПАДКУ КОРЕКТНОГО ЗАПИТУ

Відповідь	<pre>{ "method": "Purchase", "error": false, "errorCode": "", "errorDescription": "" "params": {} }</pre>
-----------	--

5.2.1.3 ВІДПОВІДЬ У ВИПАДКУ НЕКОРЕКТНОГО ЗАПИТУ

Відповідь у випадку НЕкоректного запиту та в інших випадках, які перелічені в п. 4.3.3. як значення ключа `errorCode`, описана в пункті 5.1.

5.2.2. СКАСУВАННЯ ОПЕРАЦІЇ "VOID"

Результат виконання операції **Void** можна отримати за допомогою **GetLastResult** після повернення терміналу в статус **S00** (див. п.5.4.1)

Запит	<pre>{ "method": "Void", "params": { "invoiceNum": "000456", "transactionUid": "2i89y0787300", // опціональний ключ, ідентифікатор операції на стороні зовнішньої системи "splitData": { // опціональний ключ, який містить правила розщеплення суми операцій між декількома отримувачами платежів, для формату та значень ключів звертайтеся до вашого оператора розщеплення [{"merchantId": "merchanID1", "amount": "5000"}, {"merchantId": "merchanID2", "amount": "70000"}] } } }</pre>
-------	---

```

    }
  }
}

```

Відповідь у випадку коректного та некоректного запиту аналогічна пунктам 5.2.1.2 та 5.2.1.3.

5.2.3. ОПЕРАЦІЯ ПОВЕРНЕННЯ "REFUND"

Результат виконання операції **Refund** можна отримати за допомогою **GetLastResult** після повернення терміналу в статус **S00** (див. п.5.4.1)

Запит	<pre> { "method": "Refund", "params": { "transAmount": "24500", "transCurrency": "980", "rrn": "987612344098", "authCode": "123456", "_comment": "Обов'язково використання або ключа \"rrn\", або \"authCode\", або обох цих ключів в залежності від налаштувань профілю терміналу" "merchantId": "000000060007176", "transactionUid": "2i89y0787300", // опціональний ключ, ідентифікатор операції на стороні зовнішньої системи "splitData": { // опціональний ключ, який містить правила розщеплення суми операцій між декількома отримувачами платежів, для формату та значень ключів звертайтеся до вашого оператора розщеплення [{"merchantId":"merchanID1", "amount":"5000"}, {"merchantId":"merchanID2", "amount":"70000"}] } } } </pre>
-------	--

Відповідь у випадку коректного та некоректного запиту аналогічна пунктам 5.2.1.2 та 5.2.1.3.

5.2.4. ОПЛАТА З ВИДАЧЕЮ ГОТІВКИ "CASHBACK"

Результат виконання операції **CashBack** можна отримати за допомогою **GetLastResult** після повернення терміналу в статус S00 (див. п.5.4.1)

Запит	<pre>{ "method": "CashBack", "step": "1", "params": { "transAmount": "24500", "cashAmount": "700", "transCurrency": "980", "merchantId": "000000060007176", "terminalId": "SSI00001", "transactionUid": "2i89y0787300", // опціональний ключ, ідентифікатор операції на стороні зовнішньої системи "splitData": { // опціональний ключ, який містить правила розщеплення суми операцій між декількома отримувачами платежів, для формату та значень ключів звертайтеся до вашого оператора розщеплення [{"merchantId": "merchanID1", "amount": "5000"}, {"merchantId": "merchanID2", "amount": "70000"}] } } }</pre>
-------	--

Відповідь у випадку коректного та некоректного запиту аналогічна пунктам 5.2.1.2 та 5.2.1.3. Ключ "terminalId" передається у випадку коли на терміналі під одним "merchantId" налаштовано більше одного "terminalId".

5.2.5. ПРЕДАВТОРИЗАЦІЯ "PREAUTHORIZATION"

Результат виконання операції **Preauthorization** можна отримати за допомогою **GetLastResult** після повернення терміналу в статус S00 (див. п.5.4.1)

Запит	<pre>{ "method": "Preauthorization", "step": "1", "params": { "transAmount": "24500", "transCurrency": "980", "merchantId": "000000060007176", "transactionUid": "2i89y0787300", // опціональний ключ, ідентифікатор операції на стороні зовнішньої системи "splitData": { // опціональний ключ, який містить правила розщеплення суми операцій між декількома отримувачами платежів, для формату та значень ключів звертайтеся до вашого оператора розщеплення [{"merchantId": "merchanID1", "amount": "5000"}, {"merchantId": "merchanID2", "amount": "70000"}] } } }</pre>
-------	---

Відповідь у випадку коректного та некоректного запиту аналогічна пунктам 5.2.1.2 та 5.2.1.3.

5.2.6. ЗАВЕРШЕННЯ ПРЕДАВТОРИЗАЦІЇ "SALEPREAUTH"

Результат виконання операції **SalePreauth** можна отримати за допомогою **GetLastResult** після повернення терміналу в статус **S00** (див. п.5.4.1)

Запит	<pre>{ "method": "SalePreauth", "params": { "transAmount": "24500", "endAmount": "27000", "transCurrency": "980", "rrn": "987612344098", "merchantId": "000000060007176", "transactionUid": "2i89y0787300", // опціональний ключ, ідентифікатор операції на стороні зовнішньої системи "splitData": { // опціональний ключ, який містить правила розщеплення суми операцій між декількома отримувачами платежів, для формату та значень ключів звертайтеся до вашого оператора розщеплення [{"merchantId":"merchanID1", "amount":"5000"}, {"merchantId":"merchanID2", "amount":"70000"}] } } }</pre>
-------	--

Відповідь у випадку коректного та некоректного запиту аналогічна пунктам 5.2.1.2 та 5.2.1.3.

5.2.7. СКАСУВАННЯ ПРЕДАВТОРИЗАЦІЇ "VOIDPREAUTH"

Результат виконання операції **VoidPreauth** можна отримати за допомогою **GetLastResult** після повернення терміналу в статус **S00** (див. п.5.4.1)

Запит	<pre> { "method": "VoidPreauth", "params": { "invoiceNum": "000456" "transactionUid": "2i89y0787300" // опціональний ключ, ідентифікатор операції на стороні зовнішньої системи } } </pre>
-------	--

Відповідь у випадку коректного та некоректного запиту аналогічна пунктам 5.2.1.2 та 5.2.1.3.

5.2.9. ПОПОВНЕННЯ РАХУНКУ "P2PCREDIT"

Результат виконання операції **P2pCredit** можна отримати за допомогою **GetLastResult** після повернення терміналу в статус S00 (див. п.5.4.1). Операція використовується при поповненні балансу карт інших емітентів виключно для ПЦ Ukrcard та або зчитує реквізити картки для поповнення через карткові інтерфейси ПОС терміналу або ініціює операцію поповнення по значенням із опціональних ключів методу, якщо дані картки було отримано без використання ПОС терміналу зовнішньою системою.

Запит	<pre> { "method": "P2pCredit", "step": "1", "params": { "transAmount": "24500", "transCurrency": "980", "merchantId": "000000060007176", "pan": "xxxxxxxxxxxx", // опціональний ключ, номер картки для проведення операції без зчитування картки на терміналі "expireDate": "xxxx", // опціональний ключ, строк дії картки для проведення операції без зчитування картки на терміналі "clientFirstName": "xxxxxxxxxxxx", // опціональний ключ, ім'я власника картки або особи, яка виконує ініціює поповнення для передачі до процесінгової системи "clientFamilyName": "xxxxxxxxxxxx", // опціональний ключ, прізвище власника картки або </pre>
-------	---

	<p>особи, яка виконує ініціює поповнення для передачі до процесінгової системи</p> <pre> "transactionUid": "2i89y0787300" // опціональний ключ, ідентифікатор операції на стороні зовнішньої системи "splitData": { // опціональний ключ, який містить правила розщеплення суми операцій між декількома отримувачами платежів, для формату та значень ключів звертайтеся до вашого оператора розщеплення [{"merchantId":"merchanID1", "amount":"5000"}, {"merchantId":"merchanID2", "amount":"70000"}] } } </pre>
--	---

Відповідь у випадку коректного та некоректного запиту аналогічна пунктам 5.2.1.2 та 5.2.1.3.

5.2.8. ВИДАЧА ГОТІВКИ "CASH"

Результат виконання операції **Cash** можна отримати за допомогою **GetLastResult** після повернення терміналу в статус S00 (див. п.5.4.1)

<i>Запит</i>	<pre> { "method": "Cash", "step": "1", "params": { "transAmount": "24500", "transCurrency": "980", "merchantId": "000000060007176" "transactionUid": "2i89y0787300" // опціональний ключ, ідентифікатор операції на стороні зовнішньої системи } } </pre>
--------------	---

Відповідь у випадку коректного та некоректного запиту аналогічна пунктам 5.2.1.2 та 5.2.1.3.

5.2.9. ПОПОВНЕННЯ РАХУНКУ "DEPOSIT"

Результат виконання операції **Deposit** можна отримати за допомогою **GetLastResult** після повернення терміналу в статус S00 (див. п.5.4.1)

Запит	<pre>{ "method": "Deposit", "step": "1", "params": { "transAmount": "24500", "transCurrency": "980", "merchantId": "000000060007176", "pan": "xxxxxxxxxxxxxxxx", // опціональний ключ, номер картки для проведення операції без зчитування картки на терміналі "expireDate": "xxxx", // опціональний ключ, строк дії картки для проведення операції без зчитування картки на терміналі "clientFirstName": "xxxxxxxxxxxxxxxx", // опціональний ключ, ім'я власника картки або особи, яка виконує ініціює поповнення для передачі до процесінгової системи "clientFamilyName": "xxxxxxxxxxxxxxxx", // опціональний ключ, прізвище власника картки або особи, яка виконує ініціює поповнення для передачі до процесінгової системи "transactionUid": "2i89y0787300" // опціональний ключ, ідентифікатор операції на стороні зовнішньої системи } }</pre>
-------	---

Відповідь у випадку коректного та некоректного запиту аналогічна пунктам 5.2.1.2 та 5.2.1.3.

5.2.10. ЗАПИТ БАЛАНСУ "BALANCE"

Результат виконання операції **Balance** можна отримати за допомогою **GetLastResult** після повернення терміналу в статус S00 (див. п.5.4.1)

⚡ При операції "Balance" каса не отримує суму балансу та чек.

<i>Запит</i>	<pre>{ "method": "Balance", "params": { "transCurrency": "980", "merchantId": "000000060007176" "transactionId": "2i89y0787300" // опціональний ключ, ідентифікатор операції на стороні зовнішньої системи } }</pre>
--------------	--

Відповідь у випадку коректного та некоректного запиту аналогічна пунктам 5.2.1.2 та 5.2.1.3.

5.2.11. EXO ТЕСТ "ECHO"

Результат виконання операції **Echo** можна отримати за допомогою **GetLastResult** після повернення терміналу в статус S00 (див. п.5.4.1)

<i>Запит</i>	<pre>{ "method": "Echo", "params": { "merchantId": "000000060007176" } }</pre>
--------------	--

<i>Відповідь</i>	<pre>{ "method": "Echo", "error": false, "errorCode": "" }</pre>
------------------	--

	<pre>"errorDescription":"" "params":{} }</pre>
--	--

Відповідь у випадку некоректного запиту аналогічна пункту 5.2.1.3.

5.2.12. ЧАСТКОВЕ СКАСУВАННЯ ОПЕРАЦІЇ "PARTIALVOID"

Результат виконання операції **PartialVoid** можна отримати за допомогою **GetLastResult** після повернення терміналу в статус **S00** (див. п.5.4.1)

<i>Запит</i>	<pre>{ "method": "PartialVoid", "params": { "invoiceNum": "000456", "transAmount": "24500", "transactionUid": "2i89y0787300", // опціональний ключ, ідентифікатор операції на стороні зовнішньої системи "splitData": { // опціональний ключ, який містить правила розщеплення суми операцій між декількома отримувачами платежів, для формату та значень ключів звертайтеся до вашого оператора розщеплення [{"merchantId":"merchanID1", "amount":"5000"}, {"merchantId":"merchanID2", "amount":"70000"}] } } }</pre>
--------------	--

Відповідь у випадку коректного та некоректного запиту аналогічна пунктам 5.2.1.2 та 5.2.1.3.

 В полі **transAmount** передається виключно сума на яку повинна бути зменшена сума оригінальної транзакції.

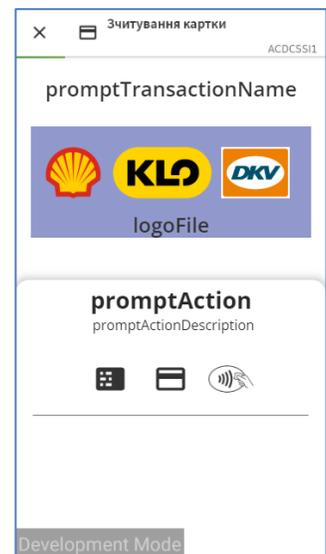
5.2.13. ЗЧИТУВАННЯ ДИСКОНТНОЇ АБО ПАЛИВНОЇ КАРТКИ "READDISCOUNTCARD"

Результат виконання операції **ReadDiscountCard** можна отримати за допомогою **GetLastResult** після повернення терміналу в статус S00 (див. п.5.4.1). Маскування номеру картки (**pan**) та найменування діапазону картки (**binName**) буде здійснюватися згідно налаштувань відповідного діапазону карт/токенів, які встановлені для терміналу у системі управління терміналами банком-еквайером. Додаткові опціональні поля можуть також модифікувати стандартні підказки користувача на екрані зчитування та відображати додатковий логотип, який повинен бути завантажений до терміналу як static content через систему управління терміналами.

```

Запит
{
  "method": "ReadDiscountCard",
  "params":
  {
    "promptTransactionName": "Дисконтна/паливна картка",
    "promptAction": "ОЧІКУЄМО КАРТКУ",
    "promptActionDescription": "Вставте, проведіть або доторкніться вашою картою",
    "logoFile": "Fuelcard.jpg"
  }
}

```



```

Відповідь
{
  "method": "ReadDiscountCard",
  "error": false,
  "errorCode": "",
  "errorDescription": "",
  "params": {}
}

```

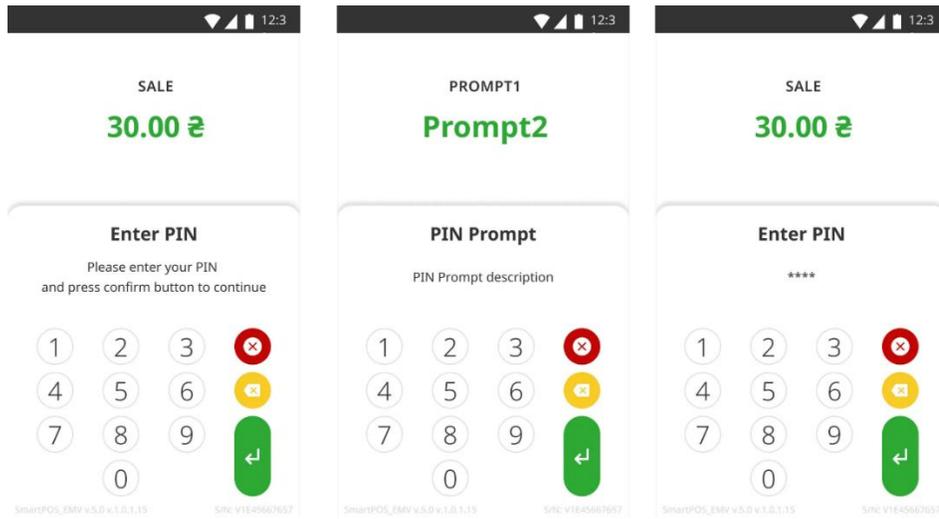
```
}
}
```

Відповідь у випадку некоректного запиту аналогічна пункту 5.2.1.3.

5.2.14. ЗЧИТУВАННЯ ПІН КОДУ ДИСКОНТНОЇ АБО ПАЛИВНОЇ КАРТКИ "GETPINCODE"

Результат виконання операції **GetPinCode** можна отримати за допомогою **GetLastResult** після повернення терміналу в статус S00 (див. п.5.4.1). Введений ПІН код повертається в ключі **pinBlock**, його значення зашифроване з використанням публічного RSA ключа, який зберігається на терміналі як static content, ключ має мати .pem формат та довжину не менше 2048 біт, файлове ім'я ключа вказується у ключі **pinPemFileName**, файл ключа попередньо повинен бути завантажений на термінал з використанням системи управління терміналами. Шифрування здійснюється за алгоритмом RSA/ECB/OAEPWithSHA-256AndMGF1Padding. Додаткові опціональні ключі можуть також модифікувати стандартні підказки та значення для користувача на екрані вводу ПІН коду.

<p><i>Запит</i></p>	<pre>{ "method": "GetPinCode", "params": { promptHeader1:"ПІДТВЕРДЖЕННЯ СПИСАННЯ БОНУСІВ", // при відсутності ключа буде передана пуста строка promptHeader2:"356", // при відсутності ключа буде передана пуста строка promptEntry:"ВВЕДІТЬ ПІН КОД", // при відсутності ключа буде передано "Введіть онлайн ПІН код" promptEntryDescription:"Введіть ваш ПІН код та натисніть зелену кнопку", // при відсутності ключа буде передана пуста строка pinMinLength:"4", // значення за замовчунням при відсутності ключа = "4". Мінімальне значення = 4 pinMaxLength:"6", // значення за замовчунням при відсутності ключа = "12" pinPemFileName:"ITBI.pem" // Для терміналів Topwise максимальна довжина цього ключа дорівнює 12 символів (8 символів - ім'я файлу + 1 символ "." + 3 символа - розширення файлу. } }</pre>
---------------------	--



Відповідь

```
{
  "method": "GetPinCode",
  "error": false,
  "errorCode": "",
  "errorDescription": "",
  "params": {}
}
```

Відповідь у випадку некоректного запиту аналогічна пункту 5.2.1.3.

5.2.15. ЗЧИТУВАННЯ ЦИФРОВОГО КОДУ "GETOTPCODE"

Результат виконання операції [GetOtpCode](#) можна отримати за допомогою [GetLastResult](#) після повернення терміналу в статус S00 (див. п.5.4.1). Введений ПІН код повертається у ключі [otpCode](#).

Маска вводу задається шаблоном, який може містити символи:

- 'X' — позиція для вводу цифри;
- '-', '' (пробіл) — роздільники;
- '380', '+380', '+XXX' — масочні цифрові префікси на початку шаблону.

Важливо: Символи 'X', пробіли та дефіси не передаються у значенні *otpCode*, тобто роздільники автоматично фільтруються. Але якщо маска починається з цифрового префікса (наприклад, 380 або +380), ці символи враховуються у довжині і будуть включені у значення *otpCode*.

Поведінка маски:

Якщо маска містить менше 'X', ніж максимальна довжина, маска все одно буде відобразитись у повному вигляді, і користувач зможе ввести більше цифр, ніж передбачено шаблоном.

Якщо маска не задана (порожня), за замовчуванням використовується маска відповідно до максимальної довжини — наприклад, якщо `maxLength = 3`, то маска буде `XXX`.

Довжина маски (кількість 'X') повинна бути у межах від 1 до 20.

Приклади допустимих масок:

+380 XX XXX XX XX

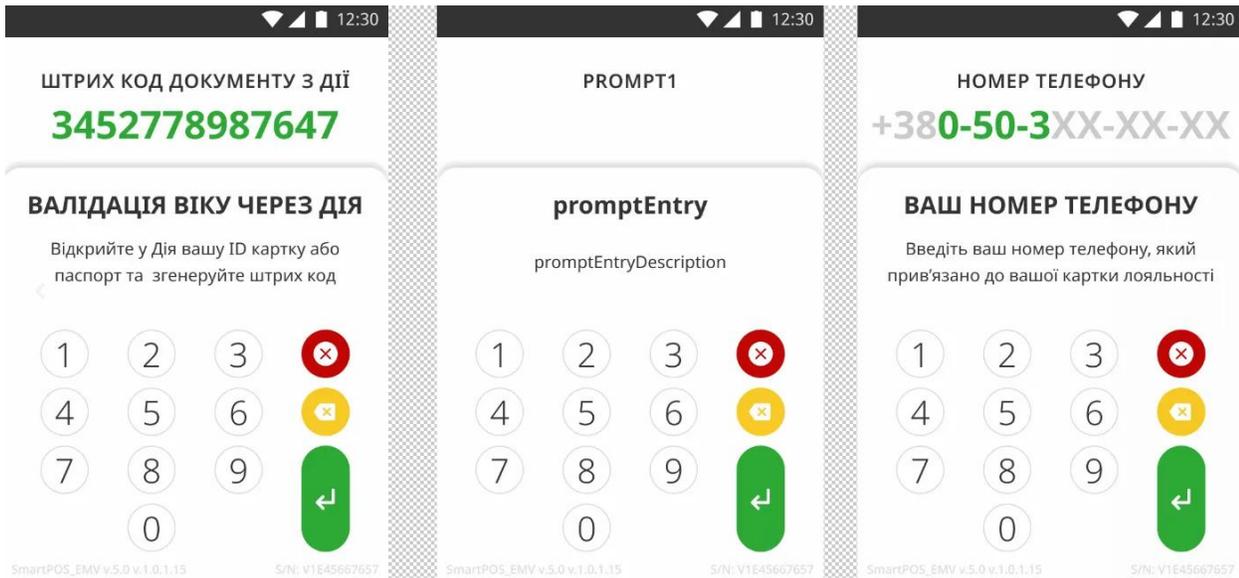
+XXX-XX-XXX-XX-XX

XX-XX

XXXX XXXX XXXX XXXX XXX

Додаткові опціональні ключі (`promptHeader1` і т.д.) можуть також модифікувати стандартні підказки та значення для користувача на екрані вводу цифрового коду.

Запит	<pre> { "method": "GetOtpCode", "params": { "promptHeader1": "КАРТА ЛОЯЛЬНОСТІ", // при відсутності ключа буде передана пуста строка "maskHeader2": "+380-XX-XXX-XX-XX", // при відсутності ключа або при пустом його значенні, маска не буде застосовуватись (буде відображено поле для вводу кода без роздільників) "promptEntry": "ВВЕДІТЬ НОМЕР ТЕЛЕФОНУ", // при відсутності ключа буде передана пуста строка "promptEntryDescription": "Введіть ваш номер телефону для пошука у базі лояльності", // при відсутності ключа буде передана пуста строка "otpMinLength": "9", // мінімально дозволена довжина коду. Дозволено вказувати від 1. При відсутності ключа, значення по замовчуванню буде "1" "otpMaxLength": "9" // максимально дозволена довжина коду. Дозволено вказувати до 20. При відсутності ключа, значення по замовчуванню буде "20" } } </pre>
-------	---



Відповідь	<pre> { "method": "getOtpCode", "error": false, "errorCode": "", "errorDescription": "", "params": {} } </pre>
-----------	--

Відповідь у випадку некоректного запиту аналогічна пункту 5.2.1.3.

5.3. ЗВІТИ

5.3.1. Z-ЗВІТ

Результат виконання операції **ZReport** можна отримати за допомогою **GetLastReport** після повернення термінала в статус S00 (див. п.5.4.1)

<i>Запит</i>	<pre>{ "method": "ZReport", "params": { "merchantId": "000000060007176", "isDetailReport": true } }</pre>
--------------	---

Відповідь у випадку коректного та некоректного запиту аналогічна пунктам 5.2.1.2 та 5.2.1.3.

5.3.2. X-ЗВІТ

Результат виконання операції **XReport** можна отримати за допомогою **GetLastReport** після повернення термінала в статус S00 (див. п.5.4.1)

<i>Запит</i>	<pre>{ "method": "XReport", "params": { "merchantId": "000000060007176", "isDetailReport": true } }</pre>
--------------	---

Відповідь у випадку коректного та некоректного запиту аналогічна пунктам 5.2.1.2 та 5.2.1.3.

5.4. СЕРВІСНІ ОПЕРАЦІЇ

Всі поля описані в цьому розділі є обов'язковими для відповідних методів.

5.4.1. ПОТОЧНИЙ СТАТУС ТЕРМІНАЛА "GETSTATUS"

Запит	<pre>{ "method": "GetStatus" }</pre>
-------	--

Відповідь	<pre>{ "method": "GetStatus", "error": false, "errorCode": "", "errorDescription": "", "status": "S00", "params": {} }</pre>
-----------	--

Відповідь у випадку некоректного запиту аналогічна пункту 5.2.1.3.

5.4.2. ПЕРЕРИВАННЯ ПОТОЧНОЇ ОПЕРАЦІЇ "INTERRUPT"

Використовуючи даний метод можна зупинити виконання операції, яка ініційована з каси, тільки на тих стадіях її виконання, коли **GetStatus** повертає коди **S02**, **S03**, **S08**. У всіх інших стадіях виконання операції POS-терміналом даний метод не перериває проходження операції.

Запит	<pre>{ "method": "Interrupt" }</pre>
-------	--

<i>Відповідь</i>	<pre>{ "method": "Interrupt", "error":false, "errorCode":""," "errorDescription":""," "params":{} }</pre>
------------------	---

Відповідь у випадку некоректного запиту аналогічна пункту 5.2.1.3.

5.4.3. ПЕРЕЗАВАНТАЖЕННЯ ТЕРМІНАЛУ "RESTART"

Метод перезавантажує термінал незалежно від того, в якому статусі він знаходиться.

<i>Запит</i>	<pre>{ "method": "Restart" }</pre>
<i>Відповідь</i>	<pre>{ "method": "Restart", "error":false, "errorCode":""," "errorDescription":""," "status": "S04" // Статус терміналу, при якому він отримав запит на Restart }</pre>

Відповідь у випадку некоректного запиту аналогічна пункту 5.2.1.3.

5.5. ДОДАТКОВІ ОПЕРАЦІЇ

5.5.1. ОТРИМАННЯ РЕЗУЛЬТАТУ ОСТАННЬОЇ ОПЕРАЦІЇ "GETLASTRESULT"

Даний метод повертає результат останньої успішної чи неуспішної операції з розділу «Безготівкові фінансові операції». Працює в форматі запит-відповідь. Формат запиту завжди однаковий, відповідь залежить від попереднього запиту. Зберігається і після перезавантаження термінала.

Запит	<pre>{ "method": "GetLastResult" }</pre>
-------	--

⚡ Якщо дані відсутні, то в полях придуть пусті/нульові значення

⚡ Для транзакції БАЛАНС transAmount завжди повертається 0

Відповідь на виконану транзакцію	<pre>{ "method": "GetLastResult", "error": false, "errorCode": "", "errorDescription": "", "params": { "originalTrnName": "Purchase", "transAmount": "24500", // Загальна схвалена сума. Для метода CashBack – складається з // transAmount + cashAmount "cashAmount": "700", // Передається, якщо оригінальна транзакція - CashBack "discount": "500", "authCode": "123456", "date": "08/11/2023", "time": "11:34:14", } }</pre>
----------------------------------	---

```
"invoiceNum": "999999",
"bankName": "Bank Acquirer",
"merchantId": "MERCHANT1",
"pan": "4725XXXXXXXX1627",
"responseCode": "0000",
"rrn": "99999999999999",
"transactionResult": "APPROVED-ONLINE",
"terminalId": "T0000001",
"posEntryMode": "CONTACTLESS",
"binName": "VISA" (береться з налаштувань таблиці діапазонів карт в TMC),
"track2": "4402722218390184=26092011881002000001",
"cardHolderName": "MR CARDHOLDER",
"parEnabled": true,
"parValue": "7",
"errorDetails": "",
"transactionUid": "2i89y0787300"
}
}
```

5.5.2. ОТРИМАННЯ РЕЗУЛЬТАТУ ОПЕРАЦІЇ ПО ЗОВНІШНЬОМУ ІДЕНТИФІКАТОРУ "GETRESULTBYUID"

Даний метод повертає результат успішної операції з розділу «Безготівкові фінансові операції» по зовнішньому ідентифікатору. Працює в форматі запит-відповідь. Формат запиту завжди однаковий, відповідь залежить від попереднього запиту. Зберігається і після перезавантаження терміналу.

Запит	<pre>{ "method": "GetResultByUid", "params" { "transactionUid": "2i89y0787300" } }</pre>
-------	--

⚡ Якщо дані відсутні або не знайдено операції з запитаним transactionID, то в полях придуть пусті/нульові значення

<p>Відповідь на виконану транзакцію</p>	<pre>{ "method": "GeResultByUid", "error": false, "errorCode": "", "errorDescription": "", "params": { "originalTrnName": "Purchase", "transAmount": "24500", // Загальна схвалена сума. Для метода CashBack – складається з // transAmount + cashAmount "cashAmount": "700", // Передається, якщо оригінальна транзакція - CashBack "discount": "500", "authCode": "123456", "date": "08/11/2023", "time": "11:34:14", "invoiceNum": "999999", "bankName": "Bank Acquirer", "merchantId": "MERCHANT1", "pan": "4725XXXXXXXX1627", "responseCode": "0000", "rrn": "99999999999999", "transactionResult": "APPROVED-ONLINE", "terminalId": "T0000001", "posEntryMode": "CONTACTLESS", "binName": "VISA" (береться з налаштувань таблиці діапазонів карт в TMC), "track2": "4402722218390184=26092011881002000001", "cardHolderName": "MR CARDHOLDER", "parEnabled": true, "parValue": "7", "errorDetails": "" } }</pre>
---	--

	<pre> "transactionUid":"2i89y0787300" } } </pre>
--	--

5.5.2. ОТРИМАННЯ ЧЕКУ ОСТАННЬОЇ ТРАНЗАКЦІЇ "GETLASTRECEIPT"

Даний метод повертає чек тій операції з розділу «Безготівкові фінансові операції», яка була виконана останньою.

<i>Запит</i>	<pre> { "method": "GetLastReceipt" } </pre>
--------------	---

<i>Відповідь</i>	<pre> { "method": "GetLastReceipt", "error":false, "errorCode": "", "errorDescription": "", "params": { "receipt": "Receipt text" } } </pre>
------------------	--

Для терміналів з ОС Android: У ключі "receipt" дані передаються у форматі XML. Якщо в цих даних присутні зображення (наприклад, логотип банку в тегу <DynamicImageItem>), то вони закодовані у шістнадцятковому (hexadecimal) форматі.

Одним із способів декодування таких зображень є використання онлайн-інструментів. Наприклад:

<https://capitalizemytitle.com/hex-to-image-converter/>

Відповідь у випадку некоректного запиту аналогічна пункту 5.2.1.3.

5.5.3. ПЕРЕВІРКА ЗВ'ЯЗКУ З ТЕРМІНАЛОМ "PINGDEVICE"

Метод можна використовувати, коли термінал знаходиться у будь-якому статусі (Sxx).

<i>Запит</i>	<pre>{ "method": "PingDevice" }</pre>
--------------	---

<i>Відповідь</i>	<pre>{ "method": "PingDevice", "error":false, "errorCode": "", "errorDescription": "", "params": {} }</pre>
------------------	---

Відповідь у випадку некоректного запиту аналогічна пункту 5.2.1.3.

5.5.4. ОТРИМАТИ РЕЗУЛЬТАТИ ПІДСУМКІВ ОСТАННЬОГО X-ЗВІТУ ЧИ ЗВІРКИ "GETLASTREPORT"

Метод повертає результати підсумків по тому методу (**XReport** або **ZReport**), який був викликаний останнім. В підсумках наводиться сумарна кількість ("штук") операцій та підсумкова сума по кожній з наведених операцій: оплата, завершення розрахунку (преавторизації), повернення, внесення готівки, поповнення, відміна.

<i>Запит</i>	<pre>{ "method": "GetLastReport" }</pre>
--------------	--

<i>Відповідь</i>	<pre>{ "method": "GetLastReport", "error":false, "errorCode": "", "errorDescription": "", "params": { } }</pre>
------------------	---

```
"transactions": [  
  {  
    "originalTrnName": "Purchase",  
    "transAmount": "24500",  
    "authCode": "123456",  
    "date": "08/11/2023",  
    "time": "11:34:14",  
    "invoiceNum": "999999",  
    "pan": "4725XXXXXXXX1627",  
    "responseCode": "0000",  
    "rrn": "99999999999999",  
    "binName": "VISA"  
  }  
],  
"responseCode": "",  
"totalSaleCount": "2",  
"totalSaleAmount": "123",  
"totalFinAuthCount": "0",  
"totalFinAuthAmount": "0",  
"totalRefundCount": "0",  
"totalRefundAmount": "0",  
"totalCashCount": "1",  
"totalCashAmount": "23",  
"totalPaymentCount": "1",  
"totalPaymentAmount": "11",  
"totalVoidCount": "0",  
"totalVoidAmount": "0",  
"isBatchEmpty": true,  
"batchNumber": "999"  
}
```

Якщо звірка не пройшла, або виникла якась помилка (окрім кода відповіді 95) - поля з `totalSaleCount` по `totalVoidAmount` заповнюються нулями, а `error` = true. Якщо пуста звірка успішна, то поля з `totalSaleCount` по `totalVoidAmount` заповнюються також нулями, а `error` = false. Масив даних `transactions` заповнюється тільки коли ключ `isDetailReport` = true.

Відповідь у випадку некоректного запиту аналогічна пункту 5.2.1.3.

5.5.5. ПЕРЕЛІК ВСІХ МЕРЧАНТІВ НА ТЕРМІНАЛІ "GETMERCHANTLIST"

<i>Запит</i>	<pre>{ "method": "GetMerchantList" }</pre>
--------------	--

<i>Відповідь</i>	<pre>{ "method": "GetMerchantList", "error": false, "errorCode": "", "errorDescription": "", "params": { "merchantCount": 2, "merchantList": ["mer1", "mer2"] } }</pre>
------------------	---

Відповідь у випадку некоректного запиту аналогічна пункту 5.2.1.3.

5.5.6. ПЕРЕЛІК ВСІХ МЕРЧАНТІВ НА ТЕРМІНАЛІ "GETMERCHANTLISTDETAILED"

<i>Запит</i>	<pre>{ "method": "GetMerchantListDetailed" }</pre>
--------------	--

<i>Відповідь</i>	<pre>{ "method": "GetMerchantListDetailed", "error": false, "errorCode": "", "errorDescription": "", "params": { "merchantCount": 2, "merchantList": [{"merchantId": "xxxx", "terminalId": "yyyy", "merchantName": "zzzz"}, {"merchantId": "xxxx", "terminalId": "yyyy", "merchantName": "zzzz"}] } }</pre>
------------------	---

Відповідь у випадку некоректного запиту аналогічна пункту 5.2.1.3.

5.5.7. ПЕРЕЛІК ВСІХ МЕРЧАНТІВ НА ТЕРМІНАЛІ "GETMERCHANTLISTDETAILED"

<i>Запит</i>	<pre>{ "method": "GetMerchantListDetailed" }</pre>
--------------	--

<i>Відповідь</i>	<pre>{ "method": "GetMerchantListDetailed", "error": false, "errorCode": "",</pre>
------------------	--

```

"errorDescription": "",
"params":
{
  "merchantCount": 2,
  "merchantList": [
    {"merchantId": "xxxxx", "terminalId": "yyyyy", "merchantName": "zzzzz"},
    {"merchantId": "xxxxx", "terminalId": "yyyyy", "merchantName": "zzzzz"}
  ]
}

```

Відповідь у випадку некоректного запиту аналогічна пункту 5.2.1.3.

5.5.8. ОТРИМАТИ ОСТАННІЙ ЧЕК X- АБО Z-ЗВІТУ "GETLASTREPORTRECEIPT"

Даний метод повертає останній чек звіту по тому методу (**XReport** або **ZReport**), який був викликаний останнім.

<i>Запит</i>	<pre> { "method": "GetLastReportReceipt" } </pre>
--------------	---

<i>Відповідь</i>	<pre> { "method": "GetLastReportReceipt", "error": false, "errorCode": "", "errorDescription": "", "params": { "receipt": "Receipt text" } } </pre>
------------------	---

Для терміналів з ОС Android: У ключі "receipt" дані передаються у форматі XML. Якщо в цих даних присутні зображення (наприклад, логотип банку в тегу <DynamicImageItem>), то вони закодовані у шістнадцятковому (hexadecimal) форматі.

Одним із способів декодування таких зображень є використання онлайн-інструментів. Наприклад:
<https://capitalizemytitle.com/hex-to-image-converter/>

Відповідь у випадку некоректного запиту аналогічна пункту 5.2.1.3.

5.5.9. ДРУК ФАЙЛА В XML-ФОРМАТІ "PRINTXML"

Даний метод дозволяє друкувати XML-файл з довільним вмістом. Найкраща якість при друку зображень шириною 384 pixels. Максимальна довжина запита 64 кБ.

Обратите внимание, а терминале доступны для использования следующие шрифты, в случае необходимости в моноширинному шрифте мы рекомендуем использовать

⚡ Тільки для Android терміналів.

<i>Запит</i>	<pre>{ "method": "PrintXml", "params": { "lrc": "26", "receipt": "<receipt fontName=\"RobotoMono-Regular.ttf\" ><text align=\"center\" size=\"normal\" weight=\"normal\" darkness=\"normal\">printtest</text><image>B64 Image</image></receipt>", "noFeed":false // опціональний параметр задля вимкнення протяжки паперу } }</pre>
--------------	---

<i>Відповідь</i>	<pre>{ "method": "PrintXml", "error":false, "errorCode": "", "errorDescription": "" }</pre>
------------------	---

"lrc" – десяткове значення контрольної суми байтів рядкового поля "receipt", обчисленої з використанням побітової операції виключного АБО (XOR) для всіх байтів поля "receipt".

"fontName" – опціональний атрибут, використовуйте його, тільки якщо ви бажаєте замінити стандартний пропорційний шрифт друку **PTM55F.ttf** на інший, яким потрібно надрукувати текст. Файл шрифту повинен знаходитися на внутрішньому диску терміналу у директорії **fonts**. На терміналі присутній за замовченням наступний набір шрифтів для друку:

JetBrainsMono-Regular	ttf
PTM55F	ttf
PTM75F	ttf
RobotoMono-Regular	ttf
RobotoMono-SemiBold	ttf
SourceCodePro-Regular	ttf
SourceSansPro-Bold	ttf
SourceSansPro-Light	ttf
SourceSansPro-Regular	ttf
SourceSansPro-SemiBold	ttf

"noFeed" - при виконанні друку не додає пусті рядки після та може використовуватися, якщо потрібно друкувати декількома послідовними викликами методу, наприклад, у випадку перевищення розміру тіла метода 64Кб

"receipt" - XML текст, який описує чек, який потрібно надрукувати. Він повинен мати наступні об'єкти для друку:

<text /text> Друк тексту з наступними властивостями:

"text align" - позиціювання тексту відносно границ друку по горизонталі з наступними значеннями: **left, center, right**

"size" - встановлення розміру тексту з наступними значеннями: **small, normal, large**

"weight" - встановлення стилю друку тексту з наступними значеннями: **normal, bold**

"darkness" - встановлення щільності друку тексту з наступними значеннями: **normal**

"| |" - розділення рядка тексту на дві частини з його розташуванням з позиціюванням зліва та справа від меж границ друку

<image /image> - дані картинки у форматі Base64, яку потрібно роздрукувати, рекомендується створювати картинку з роздільністю 384 пікселя для максимально якісного друку та уникнення зайвого простору навколо картинки. Приклад картинки малого розміру:

```
<image>Qk1iAAAAAAAAAD4AAAAoAAAAcGAAAAkAAAAVAAEAAAAACQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAP///wDqAAA
AB4AAAIeAAAAHgAAAF4AAAHgAAAB4AAAA+AAAAcAAAA=</image>
```

Додавання пустого рядку при друку: <text align="center" size="normal" weight="normal"darkness="normal"> </text>

5.5.10. ДРУК PDF-ФАЙЛА ПО URL-ЛІНКУ "PRINTPDFURL"

Даний метод дозволяє друкувати PDF-файл з вказаного URL лінку або локального файла, який завантажений на термінал з автоматичним масштабуванням друку під ширину паперу 57 мм, який використовується у POS-терміналах.

⚡ Тільки для Android терміналів.

<i>Запит</i>	<pre>{ "method": "PrintPdfUrl", "params": { "url": " https://s28.q4cdn.com/392171258/files/doc_downloads/test.pdf" або "url": "file:///sdcard/pdfrcpt.pdf" } }</pre>

<i>Відповідь</i>	<pre>{ "method": "PrintPdfUrl", "error":false, "errorCode": "", "errorDescription": "" }</pre>
------------------	--

5.5.11. ОТРИМАННЯ ДАНИХ ПО ТЕРМІНАЛУ "GETTERMINALINFO"

Даний метод повертає інформацію щодо терміналу та по встановленому програмному забезпеченню на терміналі

Запит	<pre>{ "method": "GetTerminalInfo" }</pre>
-------	--

Відповідь для Android OS	<pre>{ "method": "GetTerminalInfo", "errorCode": "", "errorDescription": "", "error": false, "params": { "securityDriver": { "name": "ServiceVer", "versionName": "X990-V1.0.12(202111221449)" }, "terminalModel": "Verifone X990", "sdkVersion": 25, "androidVersion": "7.1.2 (N_MR1)", "serviceApps": [{ "name": "VFSystemService", "packageName": "com.vfi.smartpos.system_service", "versionName": "1.11.2" }, { "name": "VFService", "packageName": "com.vfi.smartpos.deviceservice", "versionName": "3.11.2.3" }], } }</pre>
-----------------------------	--

```
"currentApp":
  {
    "name": "MonoBank",
    "packageName": "com.paydustry.banking.monobank",
    "versionName": "1.2.0.010"
  },
"terminalSerialNumber": "V1E0207420",
"thirdPartyApps": [
  {
    "name": "SSI QMate POS kiosk (d)",
    "packageName": "com.ssi.kiosk.debug",
    "versionName": "1.0.10-dev"
  },
  {
    "name": "SM Pos",
    "packageName": "com.TFPK.SMPos",
    "versionName": "1.10.19"
  },
  {
    "name": "Agent-debug",
    "packageName": "com.paydustry.agent",
    "versionName": "4.1.5"
  },
  {
    "name": "SSI QMate POS kiosk",
    "packageName": "com.ssi.kiosk",
    "versionName": "1.1.1"
  },
  {
```

```
        "name": "checkbox Verifone",
        "packageName": "ua.in.checkbox.verifone",
        "versionName": "0.6.49"
    }
    ],
    "rom": {
        "name": "ROM",
        "versionName": "V1.1.1.202111150949 INTLv7"
    },
    "paymentApps": [
        {
            "name": "MonoBank",
            "packageName": "com.paydustry.banking.monobank",
            "versionName": "1.2.0.010"
        },
        {
            "name": "Raif",
            "packageName": "com.paydustry.banking.raif",
            "versionName": "1.0.1.75"
        },
        {
            "name": "PivdennyBank",
            "packageName": "com.paydustry.banking.pivdennybank",
            "versionName": "1.5.0.008"
        }
    ],
    "externalRegisterApps": []
}
}
```

<p>Відповідь для Linux OS</p>	<pre>{ "method": "GetTerminalInfo", "error":false, "errorCode": "", "errorDescription": "", "params": { "model": "SP880", "boot_version": "01.00_00_000019", "devcfg_version": "SP880.3652010.014", "os_version": "9.0.3.0_20231017", "napi_api_version": "NAPI_API_V2.7.0", "pci_fw_version": "D_01.00_9000300", "pos_sn": "B9AA00176476" } }</pre>
-----------------------------------	--

5.5.12. ПЕРЕЗАВАНТАЖЕННЯ ПЛАТІЖНОГО ЗАСТОСУНКУ "RESTARTPAYMENTAPP"

Метод можна використовувати, коли термінал по невідомій причині не переходить до статусу **S00** після або у процесі виконання операції, або потрібно перезавантажити платіжний додаток для завантаження оновлених налаштувань платіжного додатку.

⚡ Тільки для Android терміналів.

<p>Запит</p>	<pre>{ "method": "RestartPaymentApp" }</pre>
--------------	--

<i>Відповідь</i>	<pre>{ "method": "RestartPaymentApp", "error":false, "errorCode":""," "errorDescription":""," "params":{} }</pre>
------------------	---

Відповідь у випадку некоректного запиту аналогічна пункту 5.2.1.3.

5.6. ДОДАТКОВІ ОПЕРАЦІЇ ДЛЯ МУЛЬТИ-ЕКВАЙРИНГОВОГО ФУНКЦІОНАЛУ

5.6.1. ОТРИМАННЯ СПИСКУ БАНКІВ ТА ТОРГОВЦІВ "GETMULTIPOS BANKLISTDETAILED"

Даний метод повертає список банків та торговців, які заведено на термінали. Працює в форматі запит-відповідь. Формат запиту завжди однаковий, відповідь не залежить від попереднього запиту.

<i>Запит</i>	<pre>{ "method": "GetMultiPosBankListDetailed", "params": {} }</pre>
--------------	--

Відповідь на

запит

```
{
  "errorCode": "",
  "errorDescription": "",
  "error": false,
  "method": "GetMultiPosBankListDetailed",
  "params": {
    "bankList": [
      {
        "bankEnabled": true,
        "bankId": "MonoBank",
        "bankName": "mono",
        "bankProfileId": "MONO0000000062002282",
        "merchantList": [
          {
            "merchantId": "ACDCSS10000000",
            "merchantName": "SSI",
            "terminalId": "ACDCSS1"
          }
        ]
      },
      {
        "bankEnabled": true,
        "bankId": "PivdennyBank",
        "bankName": "pivdenny",
        "bankProfileId": "PIVD0000000062002282",
        "merchantList": [
          {
            "merchantId": "7777798",
            "merchantName": "ФОП Олефіренко Сергій Олексійович",
            "terminalId": "S0130027"
          }
        ]
      }
    ]
  },
  "bankCount": 2
}
```

6. ДОДАТКИ

6.1. РОЗРАХУНОК LRC

Приклад реалізації функції для розрахунку LRC

<i>Розрахунок LRC</i>	<pre>const byteArray = encoder.encode(jsonString); // Convert byte array to hex string let jsonHex = ""; byteArray.forEach(byte => { jsonHex += ('0' + byte.toString(16)).slice(-2); }); // Calculate LRC var lrc=0; byteArray.forEach(byte => { lrc ^= byte; // XOR each byte with the current LRC value });</pre>
-----------------------	---