

ПРОГРАММНАЯ СИСТЕМА «ПЛАТЕЖНЫЙ ШЛЮЗ ПЕЙНЕТИЗИ»

Руководство по конфигурации проектов

Версия ПО 3.36

Оглавление

1. Первичная настройка проекта.....	3
2. Дополнительная настройка проекта.....	8
3. Тестирование сконфигурированного проекта	9
3.1. Тестирование с помощью Виртуального терминала.....	9
3.2. Тестирование с помощью API	12

1. Первичная настройка проекта



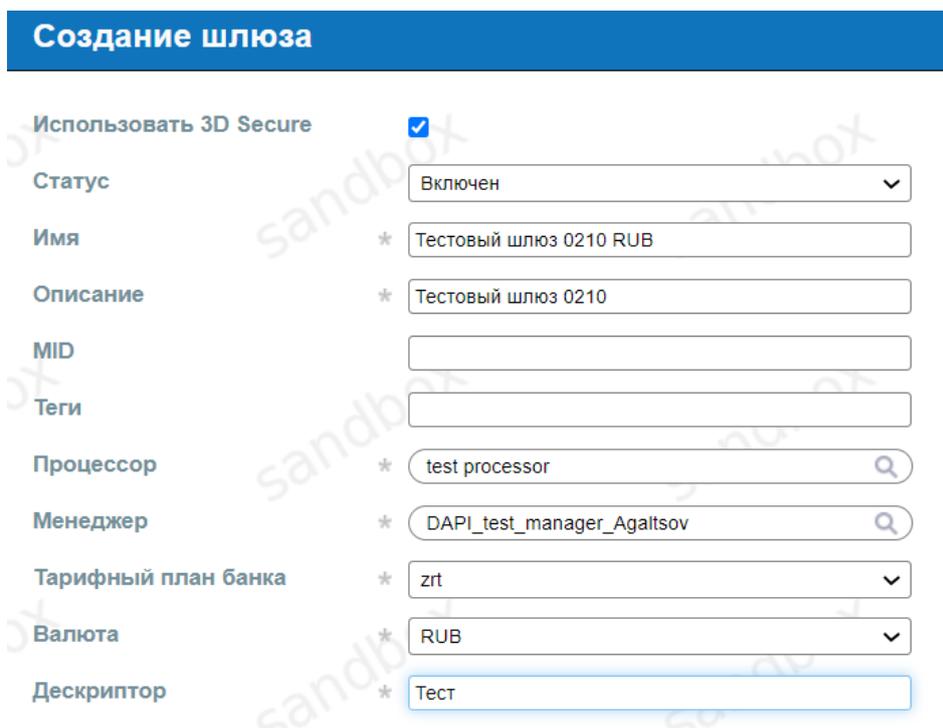
1.1. Необходимо создать аккаунт торговца в меню “Пользователи” – “Торговцы”. У торговца будет собственный логин и пароль для доступа к пользовательскому интерфейсу, а также контрольный ключ для подписи API запросов.

Создание торговца

Логин	*	<input type="text" value="test_merchant_0210"/>
Пароль		<input type="text" value="6CJnc8Zt"/> Генерировать новый пароль
Отображаемое имя	*	<input type="text" value="Тестовый торговец 0210"/>
Email	*	<input type="text" value="it@payneteasy.com"/>
Контрольный ключ		<input type="text" value="E79539DF-CAAF-47DB-AD2C-927B1798CABB"/> Генерировать новый контрольный ключ
Группа выплат	*	<input type="text" value="Common"/>
Менеджер	*	<input type="text" value="DAPI_test_manager_Agalitsov"/>
Уведомления о выплатах		<input type="text" value="Нет"/>
Вид бизнеса		<input type="text"/>
Returning customer approve sessi		<input type="text" value="1"/>
Имя	*	<input type="text" value="Тестовый"/>
Фамилия	*	<input type="text" value="Торговец"/>
Организация		<input type="text"/>
Публичный ключ (API)		<input type="text"/>
Теги		<input type="text"/>
Список IP/масок для логина		<input type="text"/>

Рисунок 1 - Создание торговца.

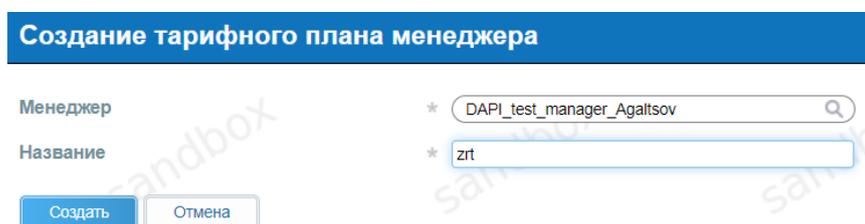
1.2. Для одного из подключённых процессоров (интегрированных банков-эквайеров) создать платёжный шлюз в меню “Настройки” – “Конфигурация” – “Шлюзы”, на котором указать валюту и данные доступа по API к операциям эквайринга в банке (вкладка “Свойства”). Если MID эквайера поддерживает только операции по картам с обязательным прохождением 3D-secure авторизации клиентом, необходимо отметить соответствующий пункт при создании шлюза.



Использовать 3D Secure	<input checked="" type="checkbox"/>
Статус	Включен
Имя	* Тестовый шлюз 0210 RUB
Описание	* Тестовый шлюз 0210
MID	
Теги	
Процессор	* test processor
Менеджер	* DAPI_test_manager_Agaltsov
Тарифный план банка	* zrt
Валюта	* RUB
Дескриптор	* Тест

Рисунок 2 - создание платёжного шлюза с режимом поддержки сценария 3D secure.

При создании платёжного шлюза также необходимо указать тарифный план банка, который создаётся в меню “Настройки” – “Тарифные планы”, нажатием кнопки “+Новый план”.



Менеджер	* DAPI_test_manager_Agaltsov
Название	* zrt

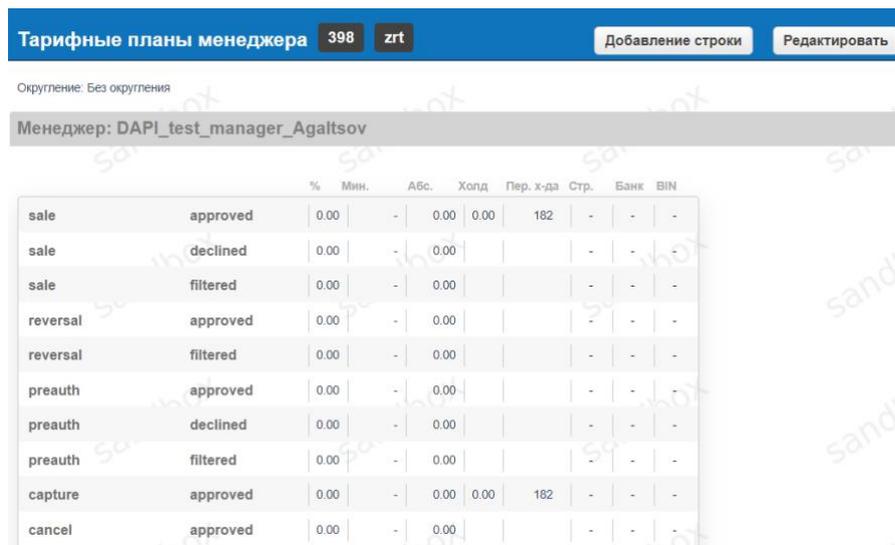
Создать Отмена

Рисунок 3 - создание тарифного плана.

После создания тарифного плана необходимо добавить в него тарификацию всех типов транзакций, которые планируется проводить. Для тестового окружения достаточно создать один тариф, который будет указан и для

шлюза, и для проекта, с нулевыми ставками тарификации. Необходимо добавить строки со следующими типами транзакций:

- Sale – approved, declined, filtered;
- Preauth – approved, declined, filtered;
- Capture – approved;
- Cancel – approved;
- Reversal – approved, filtered.

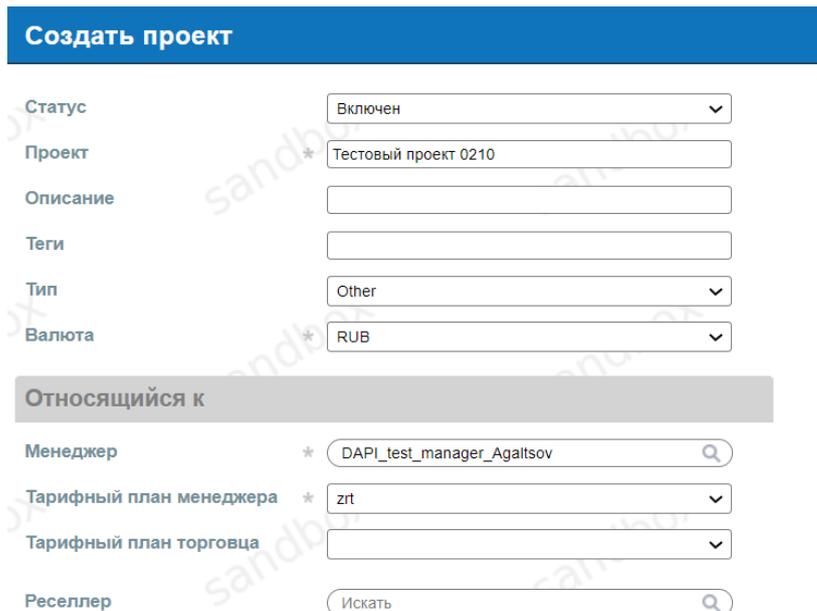


		%	Мин.	Абс.	Холд	Пер. х-да	Стр.	Банк	VIN
sale	approved	0.00	-	0.00	0.00	182	-	-	-
sale	declined	0.00	-	0.00			-	-	-
sale	filtered	0.00	-	0.00			-	-	-
reversal	approved	0.00	-	0.00			-	-	-
reversal	filtered	0.00	-	0.00			-	-	-
preauth	approved	0.00	-	0.00			-	-	-
preauth	declined	0.00	-	0.00			-	-	-
preauth	filtered	0.00	-	0.00			-	-	-
capture	approved	0.00	-	0.00	0.00	182	-	-	-
cancel	approved	0.00	-	0.00			-	-	-

Рисунок 4 - настройка тарифного плана.

При необходимости шлюз может быть клонирован с введёнными настройками для другой валюты или проекта.

1.3. Создать проект, на котором будет определена стратегия распределения трафика между доступными платёжными шлюзами. На уровне проекта также могут быть заданы пользовательские лимиты и фильтры и переопределены доступные по умолчанию платёжные формы, формы ожидания и финишные формы. Для переопределения форм, необходимо предварительно загрузить их в меню “Настройки” – “Платёжные формы”. Также на уровне проекта устанавливается тариф менеджера, предварительно созданный в меню “Настройки” – “Тарифные планы” (см. Рисунок 3, Рисунок 4).



Создать проект

Статус: Включен

Проект: Тестовый проект 0210

Описание:

Теги:

Тип: Other

Валюта: RUB

Относящийся к

Менеджер: DAPI_test_manager_Agaltsov

Тарифный план менеджера: zrt

Тарифный план торговца:

Реселлер: Искать

Рисунок 5 - создание проекта.

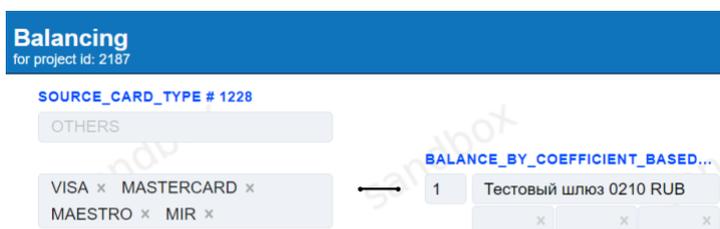
При создании проекта необходимо отметить пункт “Использовать новый балансинг”, чтобы настроить стратегию распределения транзакций максимально гибко.



Использовать новый балансинг

Рисунок 6 - указание опции использования новой системы распределения транзакций.

Настройки распределения транзакций описаны в соответствующем разделе документации. Для проекта с одним платёжным шлюзом можно выбрать любой тип маршрутизации (например, по типам всех поддерживаемых карт) и любой тип балансировки (например, балансировка по количеству транзакций).



Balancing
for project id: 2187

SOURCE_CARD_TYPE # 1228

OTHERS

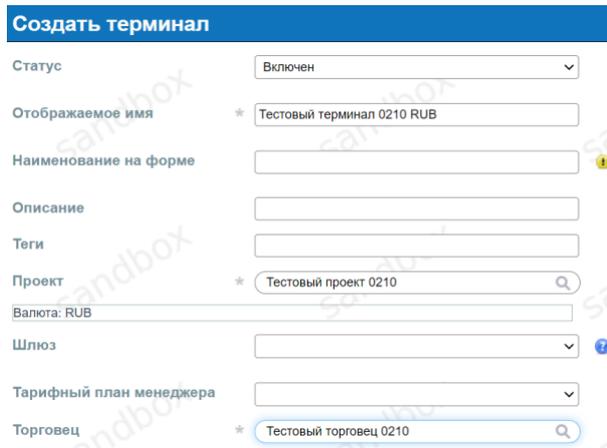
VISA x MASTERCARD x
MAESTRO x MIR x

BALANCE_BY_COEFFICIENT_BASED... 1 Тестовый шлюз 0210 RUB

Рисунок 7 - пример базовой стратегии распределения транзакций.

При необходимости проект может быть скопирован вместе со связанными терминалами и шлюзами для другой валюты или торговца.

1.4. Создать терминал, который будет использоваться как входная точка для транзакций, инициированных через API или Виртуальный терминал.



Создать терминал

Статус	Включен
Отображаемое имя	* Тестовый терминал 0210 RUB
Наименование на форме	
Описание	
Теги	
Проект	* Тестовый проект 0210
Валюта: RUB	
Шлюз	
Тарифный план менеджера	
Торговец	* Тестовый торговец 0210

Рисунок 8 - создание терминала.

На уровне терминала могут быть переопределены общие настройки проекта – платёжные формы, тарифы, лимиты. Во вкладке “Доступные операции” необходимо отметить типы операций, которые будут разрешены на этом терминале. Для интеграций по форме и тестирования с помощью виртуального терминала рекомендуется разрешить следующие операции: Sale form, Preauth form, Status, Capture, Return. Для тестирования с помощью API необходимо также разрешить Sale, Sale 3D, Preauth, Preauth 3D.

Терминал		7749	Тестовый терминал 0210 RUB
Доступные операции			
<input type="checkbox"/>	Все		
<input checked="" type="checkbox"/>	Sale		http://doc.payneteasy.eu/doc/sale-tra
<input checked="" type="checkbox"/>	Sale 3D		http://doc.payneteasy.eu/doc/sale-tra
<input checked="" type="checkbox"/>	Preauth		http://doc.payneteasy.eu/doc/preauth
<input checked="" type="checkbox"/>	Preauth 3D		http://doc.payneteasy.eu/doc/preauth
<input checked="" type="checkbox"/>	Capture		http://doc.payneteasy.eu/doc/preauth
<input type="checkbox"/>	E-Check		http://doc.payneteasy.eu/doc/echeck-
<input checked="" type="checkbox"/>	Return		http://doc.payneteasy.eu/doc/return-ti
<input type="checkbox"/>	Void		http://doc.payneteasy.eu/doc/return-ti
<input checked="" type="checkbox"/>	Status		Зависит от используемого API
<input type="checkbox"/>	Bitcoin		http://doc.payneteasy.eu/doc/bitcoin-i
<input type="checkbox"/>	Western Union		http://doc.payneteasy.eu/doc/money-
<input type="checkbox"/>	Card ref		http://doc.payneteasy.eu/doc/recurrei
<input type="checkbox"/>	Card info		http://doc.payneteasy.eu/doc/recurrei identifier
<input type="checkbox"/>	Rebill		http://doc.payneteasy.eu/doc/recurrei
<input type="checkbox"/>	Sync rebill		По запросу
<input type="checkbox"/>	Buy Now		http://doc.payneteasy.eu/doc/buy-nov
<input checked="" type="checkbox"/>	Sale Form		http://doc.payneteasy.eu/doc/paymer
<input type="checkbox"/>	Payment Form		По запросу
<input checked="" type="checkbox"/>	Preauth Form		http://doc.payneteasy.eu/doc/paymer

Рисунок 9 - определение доступных операций на терминале.

При необходимости терминал может быть скопирован для текущего или нового торговца, валюты и проекта.

1.5. Базовая настройка завершена. Для проверки корректности конфигурации рекомендуется провести тестовую оплату с помощью Виртуального терминала или API.

2. Дополнительная настройка проекта

Множество дополнительных настроек проекта позволяет сформировать платёжное решение для торговца практически любой сложности, в зависимости от поставленных задач. Возможные дополнительные настройки:

- Создание группы терминалов для мультивалютных решений;
- Создание мастер-терминала для отображения нескольких платёжных методов на единой платёжной форме;

- Добавление в проект роли реселлера (представитель компании-посредника, подключающей новых торговцев по франшизе) или роли дилера (представитель эквайрингового канала).

3. Тестирование сконфигурированного проекта

3.1. Тестирование с помощью Виртуального терминала

- 3.1.1. Перейти в “Инструменты” – “Виртуальный терминал” – “ВТ”, кликнуть на поле “Терминал” и выбрать терминал с необходимым именем, либо ввести его номер (в примере 7749) и выбрать его.

Виртуальный терминал

Проект: Все

Торговец: Все

Терминал: 7749

Тестовый терминал 0210 RUB
identifier:7749

Выберите терминал

Рисунок 10 - выбор терминала для проведения тестовой транзакции.

- 3.1.2. Заполнить обязательные поля, помеченные звездочкой, необходимыми значениями и нажать “Отправить”:

Тип транзакции	*	SALE	▼
Сумма заказа (RUB)	*		100
Содержание заказа	*	test order description	
Номер заказа	*	1	
Имя	*	Firstname	
Фамилия	*	Lastname	
Адрес	*	Red sq, 1	
Город	*	Moscow	
Индекс	*	123123	
Телефон	*	89161112233	
Эл. почта	*	it@payneteasy.com	
IP адрес покупателя	*	194.50.24.3	
User-Agent покупателя		Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit	
Дата рождения			
4 цифры SSN			
Страна	*	Russian Federation	▼
Штат			▼
Язык		Russian	▼
URL сайта			
URL возврата пользователя			

Получить URL на оплату

Сохранить как шаблон Отправить

Рисунок 11 - заполнение данных тестовой транзакции.

3.1.3. Ввести на появившейся форме карточные данные и нажать “Process payment”:

Pay securely with credit card

Credit or Debit card number:

Card expiration date:

Card printed name:

Card security code (CVV2/CVC2):





 **SECURE CONNECTION** [Cancel](#)

Рисунок 12 - введение карточных данных на форме оплаты.

Терминал принимает любой валидный карточный номер и срок действия карты не ранее следующего календарного месяца. Валидность номера карты проверяется с помощью алгоритма Луна. Пример валидного номера карты: 4444555566661111. Имя держателя карты не проверяется, но должно содержать только буквы. В зависимости от введённого кода CVV результат транзакции будет разным:

- 123 или 1234 для получения успешной транзакции (*approved*);
- 321 для получения эмуляции формы 3D Secure (пароль 3Ds для получения успешной транзакции - *hint*);
- 111 для пропуска фазы 3Ds и получения успешной транзакции (*approved*);
- 217 для получения статуса *unknown* (эмуляция проблемы связи с эквайером);
- Любой другой CVV для получения неуспешной операции (*decline*).

Валидный номер карты для разных платёжных систем можно сгенерировать в инструменте отладки на странице документации:

http://doc.payneteasy.com/integration_helpers/test_card_data.html

3.1.4. После завершения процессинга транзакции, оказавшись на странице с транзакцией, проверить её статус:



Рисунок 13 - отображение успешной транзакции в личном кабинете.



Рисунок 14 - отображение транзакции с неизвестным статусом в личном кабинете.



Рисунок 15 - отображение неуспешной транзакции в личном кабинете.

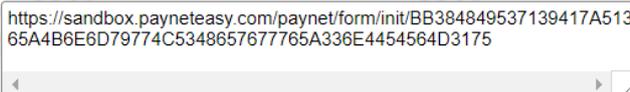
3.1.5. Виртуальный терминал также позволяет создавать ссылки на оплату, чтобы карточные данные вводились удалённо. Для этого в пункте 4. необходимо отметить соответствующий пункт:

 **Получить URL на оплату**

Рисунок 16 - отметка получения ссылки на оплату.

После нажатия кнопки “Отправить” в таком случае будет создан URL, при переходе на который будет отображена платёжная форма (аналогично представленной в 3.1.3).

URL для оплаты



```
https://sandbox.payneteasy.com/paynet/form/init/BB384849537139417A51365A4B6E6D79774C5348657677765A336E4454564D3175
```

Рисунок 17 - получение ссылки на оплату.

3.2. Тестирование с помощью API

Подробное описание протоколов API Payneteasy описано в соответствующем разделе документации. Для тестирования предлагается инициировать sale-транзакцию и, в случае прохождения сценария с 3Ds-авторизацией, также провести запрос статуса транзакции и получить в ответе ссылку на прохождение 3Ds-авторизации в параметре “redirect-to”.

3.2.1. Проведение транзакции sale.

Для примера расчёта подписи взяты следующие значения:

- Терминал (endpoint_id): 7749;
- Номер заказа торговца (client_orderid): 1;
- Сумма транзакции (amount): 100 RUB;
- E-mail клиента (email): it@payneteasy.com;
- Контрольный ключ торговца (merchant control key): E79539DF-CAAF-47DB-AD2C-927B1798CABB.

Сформированная строка для подписания:

7749110000it@payneteasy.comE79539DF-CAAF-47DB-AD2C-927B1798CABB

Полученная SHA-1 подпись:

adf6810e892fdc0ba9dd9c487967c87e18ea75ce

Сформированный curl-запрос с параметрами и подписью из примера:

```
curl --data "client_orderid=1&order_desc=Test Order
Description&first_name=John&last_name=Smith&ssn=1267&birthday=19820115
&address1=100 Main
st&city=Seattle&state=WA&zip_code=98102&country=US&phone=+1206358204
3&amount=100&email=it@payneteasy.com&currency=RUB&ipaddress=1.2.3.4&
site_url=www.google.com&credit_card_number=4444555566661111&card_print
ed_name=CARD
HOLDER&expire_month=12&expire_year=2099&cvv2=123&purpose=payment&r
edirect_url=http://doc.payneteasy.com/doc/dummy.htm&server_callback_url=http
://doc.payneteasy.com/doc/dummy.htm
&control=adf6810e892fdc0ba9dd9c487967c87e18ea75ce"
https://sandbox.payneteasy.com/paynet/api/v2/sale/7749
```

3.2.2. Запрос статуса транзакции.

Для примера расчёта подписи взяты следующие значения:

- Логин торговца (login): test_merchant_0210;
- Номер заказа торговца (client_orderid): 1;
- Номер заказа в системе Payneteasy, возвращается в Sale response (orderid): 2;
- Контрольный ключ торговца (merchant control key): E79539DF-CAAF-47DB-AD2C-927B1798CABB.

Сформированная строка для подписания:

test_merchant_021012E79539DF-CAAF-47DB-AD2C-927B1798CABB

Полученная SHA-1 подпись:

c9c7ab2dc1d77485a0bd7f41c87c95d201b6b7aa

Сформированный curl-запрос с параметрами и подписью из примера:

```
curl --data
"login=test_merchant_0210&client_orderid=1&orderid=2&control=c9c7ab2dc1d7
7485a0bd7f41c87c95d201b6b7aa"
```

<https://sandbox.payneteasy.com/paynet/api/v2/status/7749>

Пример фрагмента ответа:

```
type=status-response
&merchant-order-id=1
&paynet-order-id=2
&status=processing
&redirect-to= https://sandbox.payneteasy.com/paynet/form/redirect/A_1...
&amount=100
...
```

- 3.2.3. После завершения процессинга транзакции её статус можно получить в личном кабинете (см. 3.1.4) или запросом статуса по API (ниже пример ответа):

```
type=status-response
&serial-number=00000000-0000-0000-0000-0000005b5eec
&merchant-order-id=1
&processor-tx-id=9568-47ed-912d-3a1067ae1d22
&paynet-order-id=2
&status=approved
&amount=100
&descriptor=no
&transaction-type=sale
&receipt-id=2050-3c93-a061-8a19b6c0068f
&name=FirstName
&cardholder-name=FirstName
&card-exp-month=3
&card-exp-year=2028
&email=it@payneteasy.com
&processor-rrn=510000000111
&approval-code=380111
&order-stage=sale_approved
&last-four-digits=1111
&bin=444455
&card-type=VISA
&phone=%2B79685787194
&bank-name=UNKNOWN
&paynet-processing-date=2020-09-29+10%3A23%3A34+MSK
```