

Connection methods

Regardless of the chosen connection method, you are given access to the [personal account](#) on the payment gateway server.

Below is a complete list of connection methods with brief descriptions. Click on the selected connection method to go to the detailed description.

Accepting payments via Android Pay

The Android Pay payment method is currently not supported by Google. This payment method will be replaced by Pay with Google (it is currently temporarily unavailable for integration, documentation is under development).

Android Pay is a mobile payment system that can be used by owners of Android devices. Interaction with the Android Pay system is carried out through [programming interface](#) (API) of the payment gateway.

Programming interface (API)

Connecting via a programming interface (API) allows you to use all the available functionality of the payment gateway.

Accepting payments via Apple Pay

Apple Pay is a mobile payment system that can be used by owners of Apple devices. Interaction with the Apple Pay system is carried out through [programming interface](#) (API) of the payment gateway.

Plugins for site management systems (CMS)

If you are planning to create an online store using a site management system (CMS), you can use a payment plugin. We support the most popular site management systems. Install the plugin on your CMS and accept payments.

Fiscalization - 54-FZ

Most of the plugins presented support fiscalization. To clarify whether the plugin you are interested in supports the fiscalization functionality, you can visit our technical support address – support@ecom.sberbank.ru.

The plugins support **ApplePay/GooglePay/SamsungPay** payment methods.

Accepting payments via Google Pay



By accepting payments via Google Pay, you agree to [Google terms of service](#).

Google Pay is a payment system that can be used by users with an account in this system. Interaction with the Google Pay system is carried out through [programming interface](#) (API) of the payment gateway.

Information for developers is available in [Google documentation](#).

[Читать дальше...](#)

Payment button

A simple way to ensure that payments are accepted on any website is to place a button next to the description of a product or service that will open a payment form in a modal window (iframe). An example of using the payment button is available at [this page](#).

You can use [test cards](#) to try out the functionality.

Issuing an invoice to a buyer from a personal account

The merchant issues an invoice to the buyer from their personal account, and a link to the payment page is sent to the buyer's email address. You don't need your own website to bill this way.

Accepting payments via Samsung Pay

Samsung Pay is a mobile payment system that can be used by owners of Samsung devices. Interaction with the Samsung Pay system is carried out through [programming interface](#) (API) of the payment gateway.

There are two ways to pay with Samsung Pay:

- payments in Mobile App;
- payment from the web page (the payment page is located on the side of the payment gateway);
- payment from the web page (the payment page is on the merchant's side).

Payment by static link

[invoice-to-messenger.mp4](#)

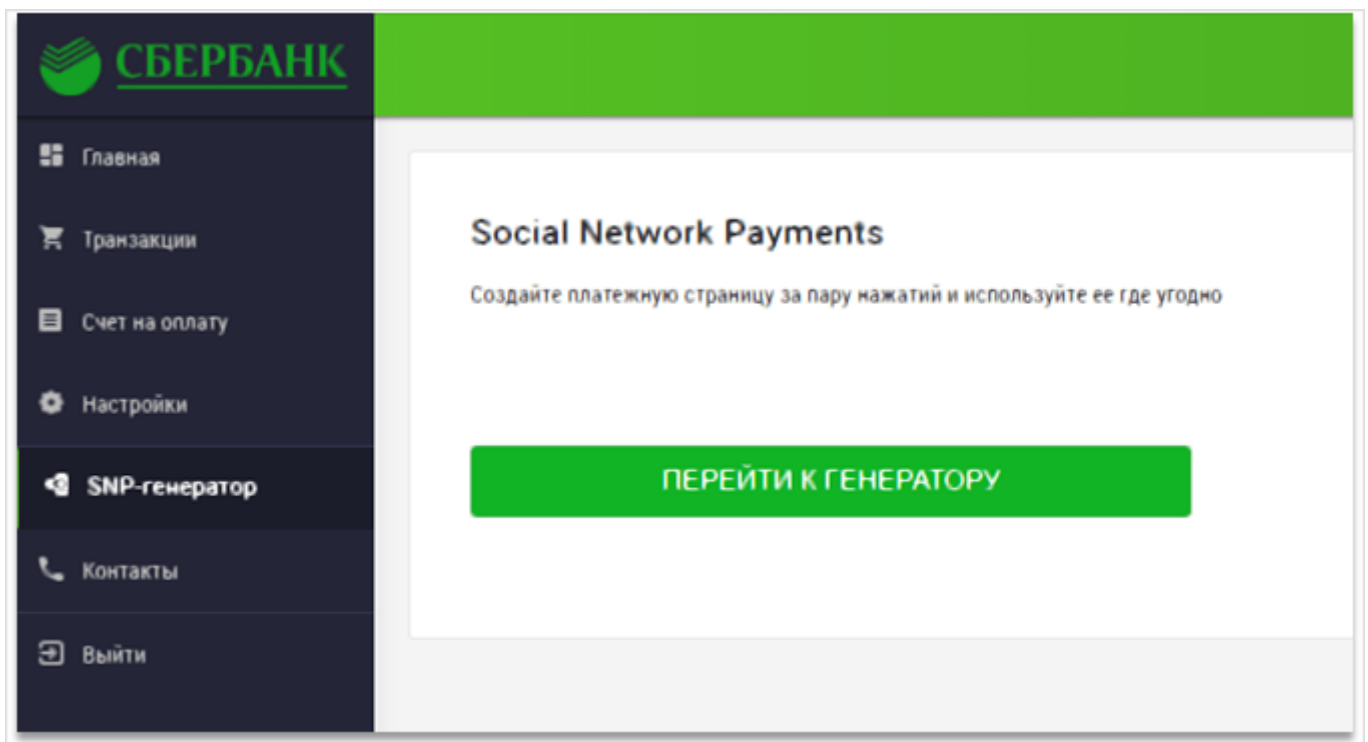
The easiest way to accept payment is to give the buyer the opportunity to click on a regular link and enter the amount, payment purpose and bank card details on their own. The merchant generates a link that leads to the payment page. This link can be:

- place on the store's website;
- send to the buyer by email;
- send to the buyer using the instant messaging system (WhatsApp, Telegram, Viber, etc.);
- post on social networks.

The buyer follows the link to the payment page, enters bank card details and transfers funds. This method of connection does not require programming skills or your own website.

Generating a payment link using a generator

The **Account for the messenger** section will allow you to expand the list of payment acceptance methods that do not require time-consuming integration.



To create a link to pay using the payment link generator, follow these steps.

Go to the link creation web page from your personal account ([test environment/production environment](#)), **Invoice for messenger** section.

Click on the **Go to generator** button.

The following page will be displayed.

Формирование счета на оплату

RU

Сумма платежа

₽

Описание платежа

Осталось символов: 150

Дополнительные поля

☒ Электронная почта

☐ ФИО

☐ Телефон

☐ Адрес доставки

+

ДОБАВИТЬ ПОЛЕ

СОЗДАТЬ ССЫЛКУ

СОЗДАТЬ QR-КОД

Fill in the fields with values:

Setting/Field	Description	Mandatory
Payment amount	The order amount.	No
Description	Description of the order in any format.	No
Email	Whether or not to display an email input field the buyer on the payment page.	Yes

Setting/Field	Description	Mandatory
Full name	Whether or not to display a field for entering the last name, first name and patronymic of the buyer on the payment page.	No
Telephone	Whether or not to display a field for entering the number the buyer's phone number on the payment page.	No
Delivery address	Whether or not to display a field for entering an address delivery on the payment page.	No

To create an additional field that will be displayed on the payment page, click on the **ADD FIELD** button and enter the name of the new parameter, under which the field will be displayed on the payment page.

Default payment link lifetime is 6 months from creation or last payment.

To check the functionality of the generated link, copy and paste it into the address bar of your browser. After that, a ready-made payment form will be displayed.

You can use [test cards](#) to try out the functionality.

Example of payment link

Here is an example of a link with dynamic parameters for payment – the client needs to [enter the amount and email](#).

Below is an example of the link code.

```
<a target="_blank"
href="https://3dsec.sberbank.ru/payment/docsite/payform-1.html?token=YRF3C5R
FICWISEWFR6GJ&ask=amount&ask=email&ask=description">
    Pay for services
</a>
```

You can use [test cards](#) to try out the functionality.