

For an unauthorized user: the LOGIN home page opens, with buttons proceeding to the registration page or resetting the password. The sidebar on the left has only the Login section.

For an authorized user opens the start page without content. The display of sections of the sidebar on the left depends on the role which user had for authorization.

There are two possible roles with different rights: admin and customer.

Authorization:

A new user can be registered at /register. All fields are required, Email must be unique, password and password confirmation must match. A message with the code is sent to the email specified during registration. After entering this code on the site, the user registers in the system and becomes authorized on the site.

When the password is lost, it can be restored on the /restore page. Password recovery consists of three stages:

- 1) Entering an email (incorrect or non-existent will be rejected).
- 2) Entering the recovery code. A message with a code is sent to the specified email address. When the code is entered, a password change form will open.
- 3) The form of the change. You need to enter the password and confirm the password. After submitting this form, the user's password is changed.

Authorization. You need to enter the email address and password of an existing user. If the authorized user is an administrator, then in the sidebar on the left there will be sections: Users, Stores, Transactions, Invoices, Invoices Builder, Buttons, Buttons Builder, Logout. If the authorized user is a customer, there will be sections: Stores, Transactions, Invoices, Invoices Builder, Buttons, Buttons Builder, Logout.

If the user is blocked by the admin or his token is not up-to-date, the next time he tries to go to another page, he will be thrown to the /login page. The /stores, /transactions, /users, /payments pages require authorization. The /login /restore /register pages require that the user is not logged in.

Any person can enter the /bill/[id] (bill payment) page, even if they are not authorized.

ADMIN

Cannot be created from the front end. It is created manually in the database. Admin can view all the data of all users on the pages /users, /stores, /transactions, /Invoices, /Invoice Builder, /Buttons, /Buttons Builder, but at the same time cannot have its own stores (respectively, either admin cannot create its own payments).

The admin can block/unblock the user by clicking the block/unblock button on the /users/[id] page. In this case, the user will not be able to log in to the site until the lock is lifted (there will be an error "user is blocked" when logging in).

The admin can block/unblock stores by clicking the block/unblock button on the /stores/[id] page. In this case, the user will not see the blocked store in the drop-down list of available stores on the /payments and /transactions pages (respectively, it is impossible to create a payment on behalf of this store).

CUSTOMER

The client can create and edit stores, create payments linked to a specific store, and make payment transactions.

CREATING A STORE

First you need to create a store. To do this, on the /stores page, click the "Add store" button and fill in the fields Name, Callback URL (a post request with payment information will be sent to the specified URL when the payment status changes) and ApiKey (click on the "generate" button) and click OK. The ApiKey field is not mandatory, but it is impossible to create a payment without it (a store without an ApiKey will not be displayed in the list of available stores). The created store will appear in the table on the /stores page.

To edit the store data, click on the line of the desired store in the table, the page of this store will open. The Edit button opens the edit form. All the new data entered inside, will replace the current one after clicking on the "OK" button

The admin can block/unblock the store by pressing the block/unblock button.

CREATING A PAYMENT

Next, you need to create a payment. There are two pages for this:

`/invoicesBuilder` and `/buttons Builder`. On the `/invoicesBuilder` page, you need to select from which store the payment will be created and fill in the Wallet fields (the Casper user's wallet is specified) and Amount (it must be at least 2.5) and click the "Add Payment" button. On the `/button builder` page, in addition to the payment, a button is created with a transition to the payment receipt. This page specifies the fields: Name (button name), Amount, Store, Comment (payment comment) and Button type (three button styling options). Next, you need to click on the "Save" button, after which the clickable button itself and its HTML code will appear, which can be copied by clicking on the "copy" button. On the `/invoices` page, the created payments will appear via `/invoicesBuilder`, and on the `/buttons` page via `/button Builder`.

Payments created in `/invoices Builder` are stored in the `/invoices` table.

Payments created in `/buttons Builder` are stored in the `/buttons` table.

BILL PAYMENT (TRANSACTION)

After creating a payment by clicking on the payment line, you can go to the page of a specific payment, the `/bill/[id]` page opens, a link to which can be shared with any user (including unauthorized) who must pay this bill.

Note: there are 2 modes of operation of the `/bill/[id]` page - fake and Casper.

It is managed via `.env` variable

`NEXT_PUBLIC_FAKE_TRANSACTION`

In the FAKE mode (`NEXT_PUBLIC_FAKE_TRANSACTION=1`)

To do this, the user needs to open the link, fill in the Email field (the mail where the message about the successful transaction will arrive) and Wallet

(the sender's wallet, it can be obtained in Casper signer) and click the "Sign in Signer" button, after which the "Pay" button will appear. Click the "Pay" button. The "Check last transaction" button will appear, by which you can view the status of the last transaction (in fake mode, this is blocked, the transaction is always the same). After that, 2 letters should arrive in the mail within a few minutes. One letter is to the owner of the payment, with a text about changing the payment status (from Not_Paid to Paid), the other is to the sender, with a text about changing the transaction status (from fake_processing to fake_success).

In the CASPER mode (NEXT_PUBLIC_FAKE_TRANSACTION=0) for invoices

To do this, the user needs to open the link, fill in the Email field (the mail where the letter about the successful transaction will arrive) and click the "Sign in Signer" button, after which the CasperLabs Signer extension window will appear, in which you need to enter the wallet password, if necessary, consent to connect the extension to the site, and click the "Sign in Signer" button again, after which the "Pay" button will appear. Click the "Pay" button, click the "SIGN" button in the extension window, a tooltip will appear in the upper right corner of "Transaction completed". After that, 2 letters should arrive in the mail within a few minutes. One letter is to the owner of the payment, with a text about changing the payment status (from Not_Paid to Paid), the other is to the sender, with a text about changing the transaction status (from processing to success). Each transaction can be paid only once. Payment is not possible if the status of the last transaction is processing or success. If the result of the last transaction is an error, then another transaction can be carried out.

The transaction can be performed an unlimited number of times.

Working with Casper signer and possible errors

<https://chrome.google.com/webstore/detail/casperlabs-signer/djhndpflfiibmcdbnmaaahkhchcoijce> - download extension

<https://testnet.cspr.live/tools/faucet> - here you can get test tokens

Casper signer is required to make transactions. Most of the errors of interaction with it are debugged. All errors that occur are displayed in the console as Casper error: \${error}

Sometimes Casper may give the error Value Not Found (wallet not found). Therefore, you should first receive a transfer to your account from another wallet. After receiving the first transfer, the new wallet will be registered with Casper.

Transactions

After the payment is paid and the "Transaction completed" popup appears, the created transaction appears on the /transactions page in the table. The /transactions page displays those transactions that relate to the store selected in the drop-down list. By clicking on the line with the transaction, the /transactions/[id] page opens with information on the transaction and the "Check transaction" button, by clicking on which you can view the full information of this transaction on the website <https://testnet.cspr.live/>.

JSON object with transaction information:

<https://event-store-api-clarity-testnet.make.services/deloys/1c4E67848D6058FE85f3541C08d9B85f058959fb8C959Bf8A798235bc8614Bc5> - the sequence of characters at the end should be replaced with the txHash of an existing transaction.

Also for checking the transaction status:

<https://testnet.cspr.live/deploy/4b9CA4Ff6F896edCF5eDAFc2e86B1739A8259312e34d608F5D2C44464fA0c957> - the sequence of characters at the end should be replaced with the txHash of an existing transaction.