Passwordless package allows you to create a login for users without the need for user to provide password. Upon registering or login an email is sent to the user's email with a code to enter to confirm login and a link to login directly. Since the user is responding to the email it will also verify the email.

The first step to in the passwordless process is for the user to sign-up or request a token to their email address. You can do that with the following: {% apibox "Accounts.requestLoginTokenForUser" "module":"accounts-base" %}

If the user is signing up you can pass in the `userData` object like in [Accounts.createUser](#).

{% apibox "Meteor.passwordlessLoginWithToken" %} The second step in the passwordless flow. Like all the other `loginWith` functions call this method to login the user with the token they have inputted.

{% apibox "Accounts.sendLoginTokenEmail" "module":"accounts-base" %} Use this function if you want to manually send the email to users to login with token from the server. Do note that you will need to create the token/sequence and save it in the DB yourself. This is good if you want to change how the tokens look or are generated, but unless you are sure of what you are doing we don't recommend it.

## E-mail templates

`accounts-passwordless` brings new templates that you can edit to change the look of emails which send code to users. The email template is named `sendLoginToken` and beside `user` and `url`, the templates also receive a data object with `sequence` which is the user's code.

```
sendLoginToken: {
  text: (user, url, { sequence }) => { /* text template */ }
}
```

## Enable 2FA for this package

You can add 2FA to your login flow by using the package [accounts-2fa](#). You can find an example showing how this would look like [here](#).