

Low-coding Petriflow apps in Netgrif

This guide will show you how to get started with Netgrif low-code platform. The guide will show you how to use Netgrif platforms to create and deploy applications in low-code language Petriflow.

An applications in Petriflow language is given by a set of Petriflow classes, also called Petriflow processes. When Petriflow application is deployed, a user can create objects of Petri flow classes, also called instances or cases of Petriflow processes.

Petriflow language was designed to make development of process-based web applications easier and faster when compared with traditional full stack development. To do this, Petriflow abstracts from traditional three-layer architecture, when data layer, application layer and presentation layer of web applications are developed separately and then interconnected. Instead, a Petriflow class contains definition of data layer, application layer and presentation layer.

More precisely, a Petriflow class consists of:

- data attributes,
- life cycle of the objects of the class given by a workflow process that consists of tasks,
- forms containing subsets of data attributes that are associated with tasks.

Netgrif low code platform consists of:

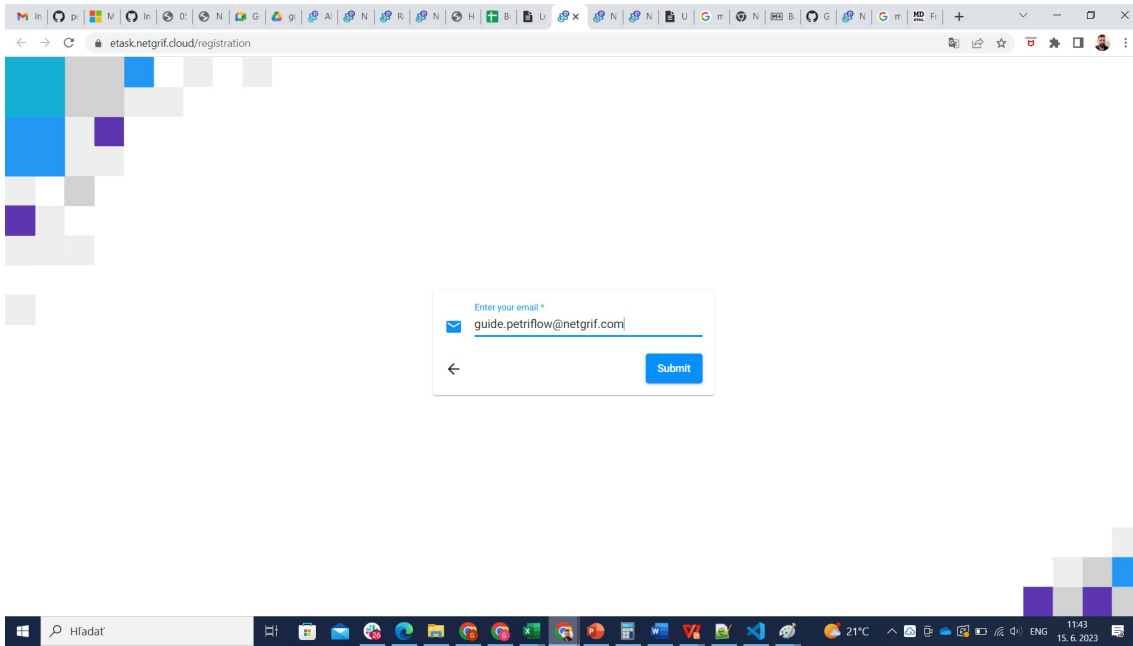
- Netgrif application builder, and
- Netgrif application engine.

Netgrif application builder available for free at <https://builder.netgrif.com> is a tool that supports graphical creation of Petriflow classes by drawing workflows and drag-and-dropping data forms.

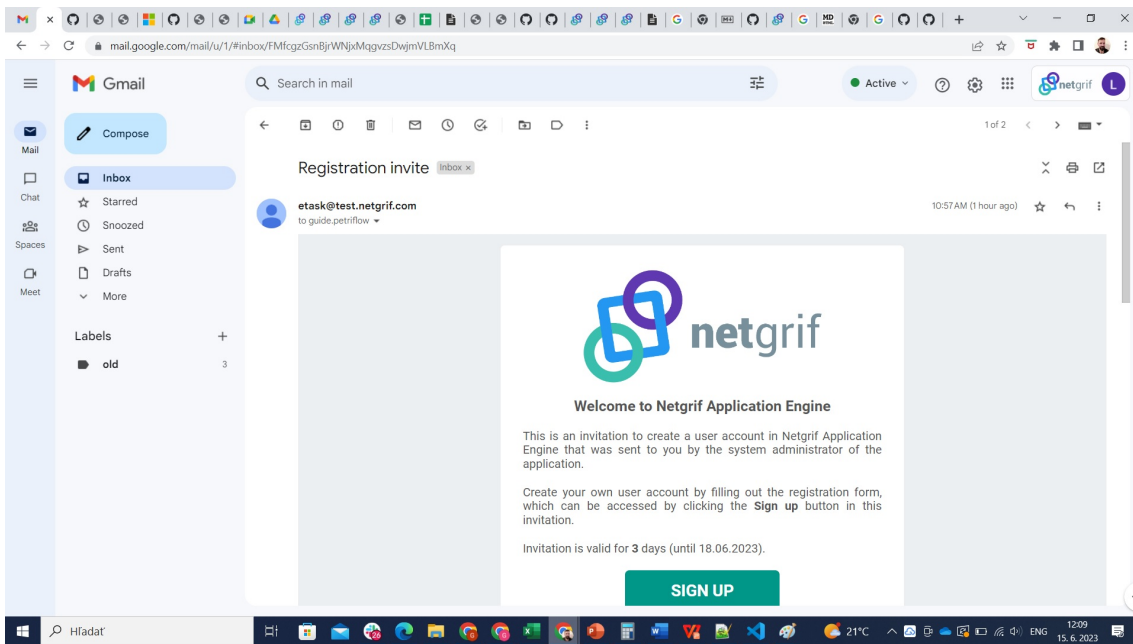
Netgrif application engine is an interpreter of Petriflow applications. It may either be downloaded from github <https://github.com/netgrif> or used as a free service after registration at <https://etask.netgrif.cloud/>.

In this guide we will use <https://builder.netgrif.com> to build examples of Petriflow classes, to describe generated code in Petriflow language and then deploy generated Petriflow classes using <https://etask.netgrif.cloud/>.

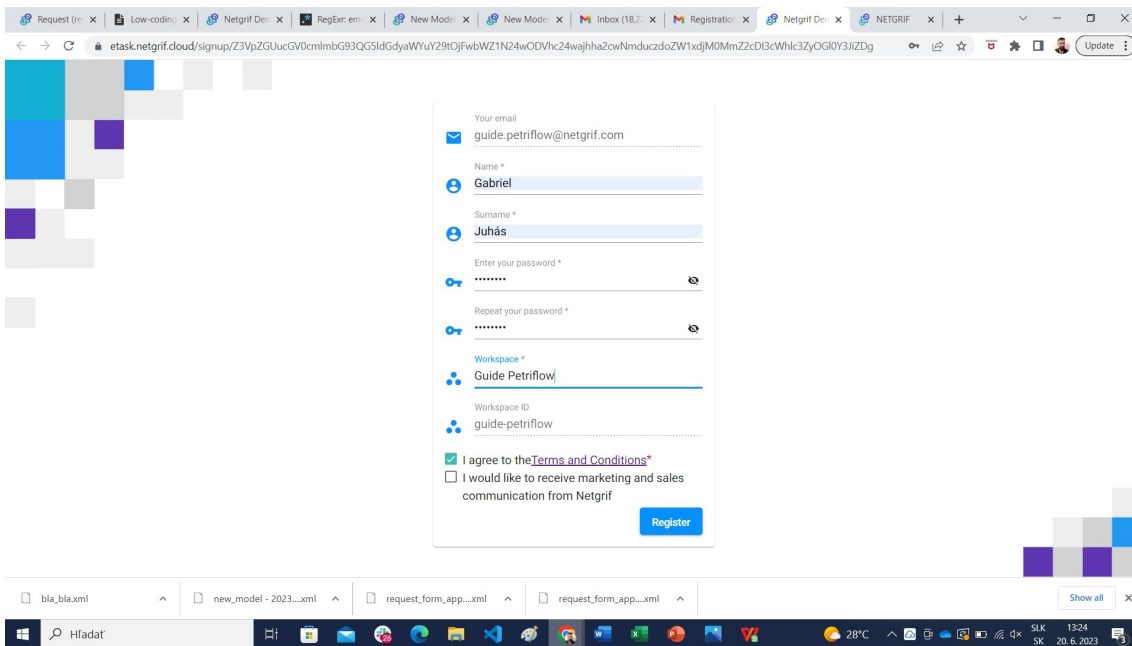
If you want to try yourself just sign up using your preferred email at <https://etask.netgrif.cloud/registration>.



You will receive an email

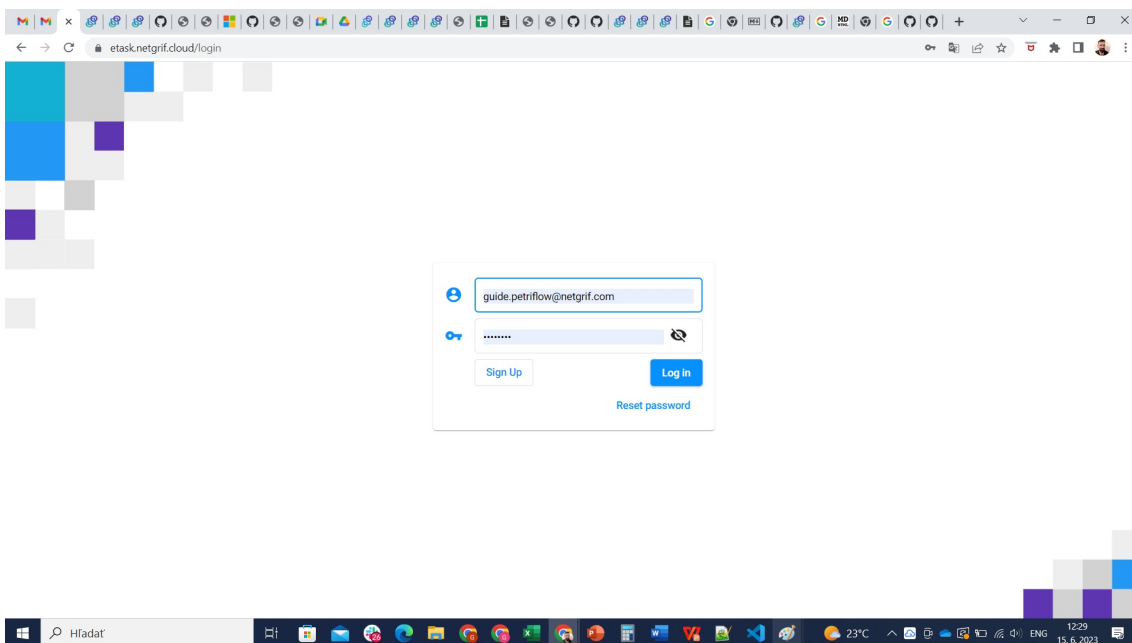


If you click on SING UP button, you will be redirected to the web page, where you can finish your registration.



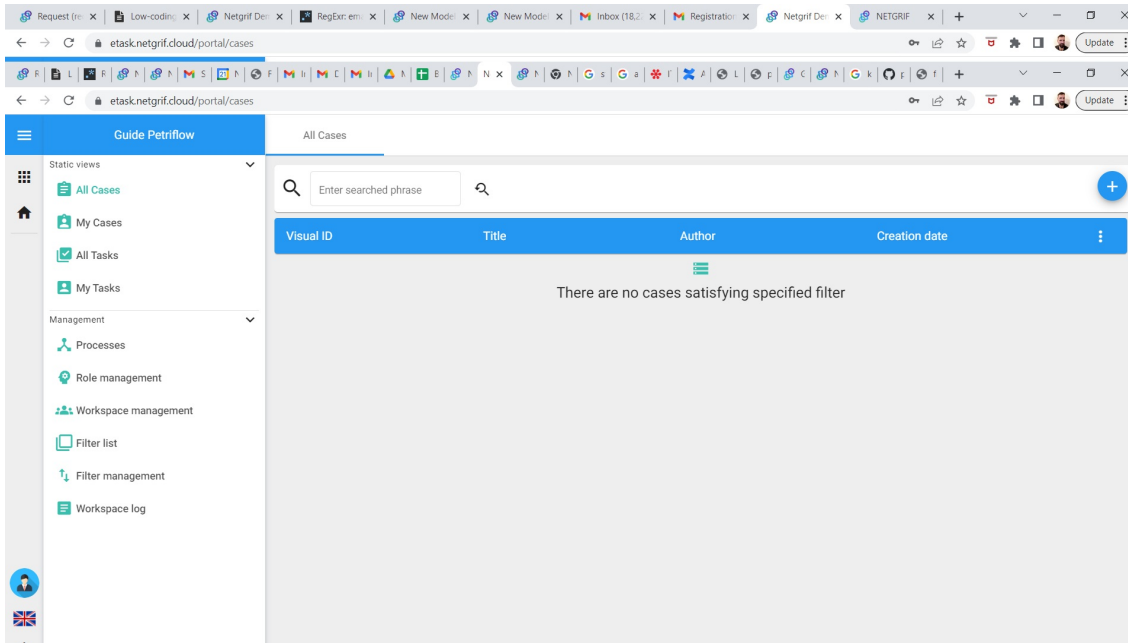
As <https://etask.netgrif.cloud/> is a multi-tenant service, here you can choose a unique Workspace name for your account. We have chosen Guide Petriflow as a name for the account. The Workspace ID will automatically be generated from your Workspace name, in our case the generated Workspace ID is guide-petriflow.

After you fill your name, surname, create your password, agree with the Terms and Conditions and press Register button, you will be redirected to the login page.



Here you enter your email and password to Log In your account of Netgrif Etask, which enables you to upload Petriflow classes. Further, users registered to your group (to your workspace) can create

Petriflow objects and perform tasks according to their workflow processes, fill the form associated to tasks etc.



In following paragraphs, we will illustrate how to crate, deploy and use Petriflow applications starting with the most simple case.

Petriflow Web application with form

Let us start with a web application in Petriflow language that consist of just one Petriflow class.

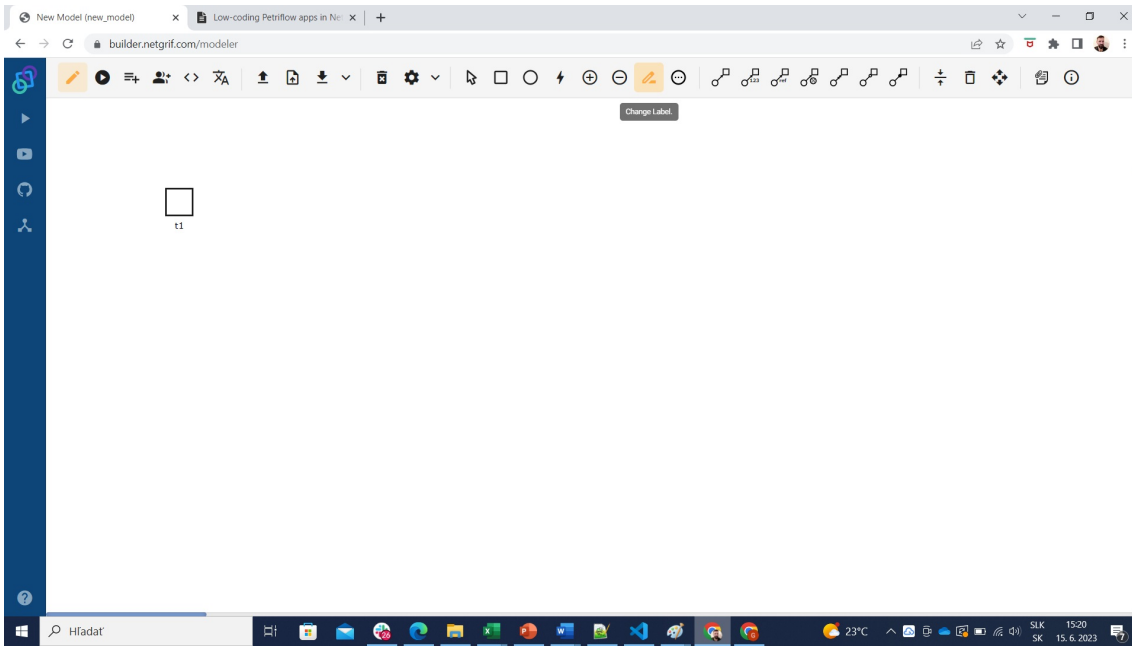
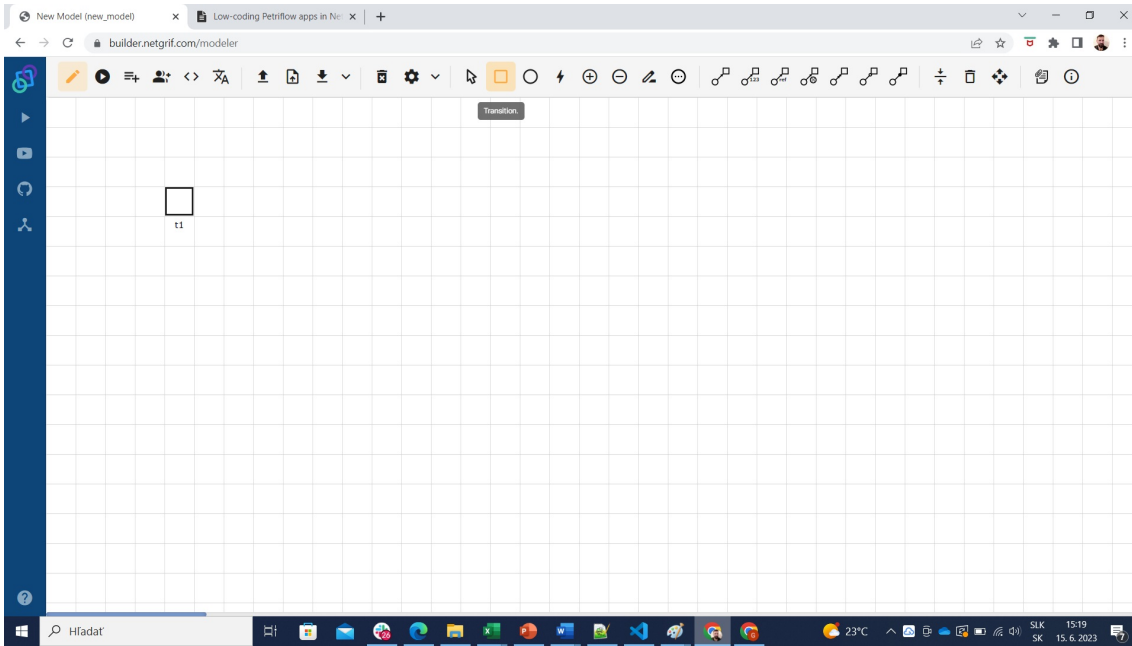
Imagine, that we want to create an application with the following functionality:

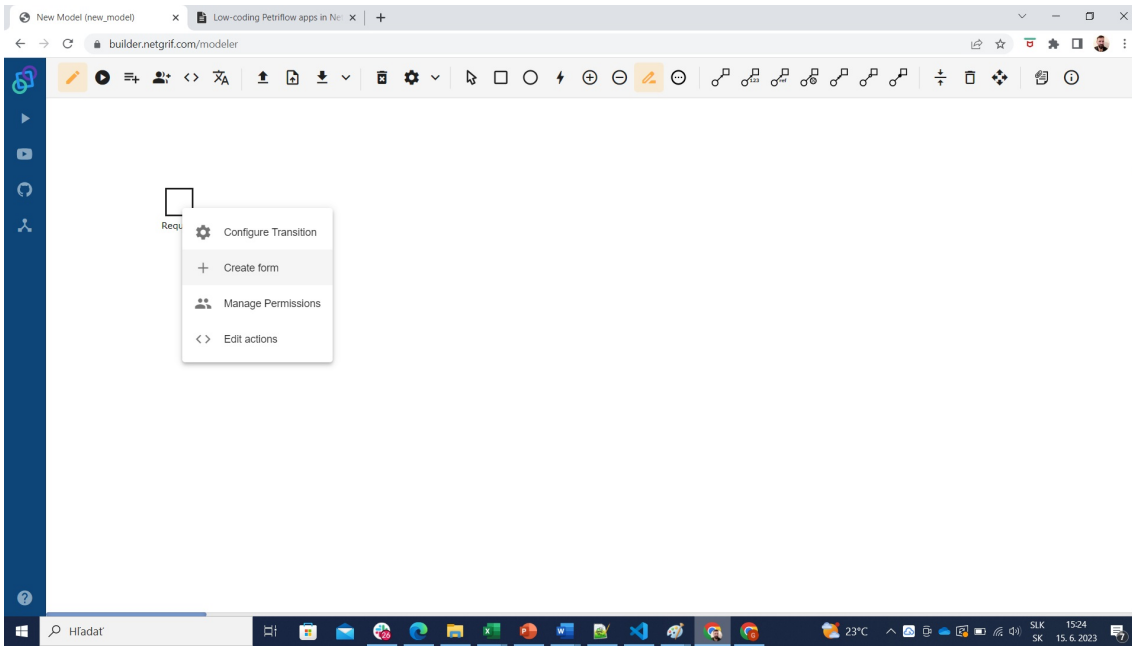
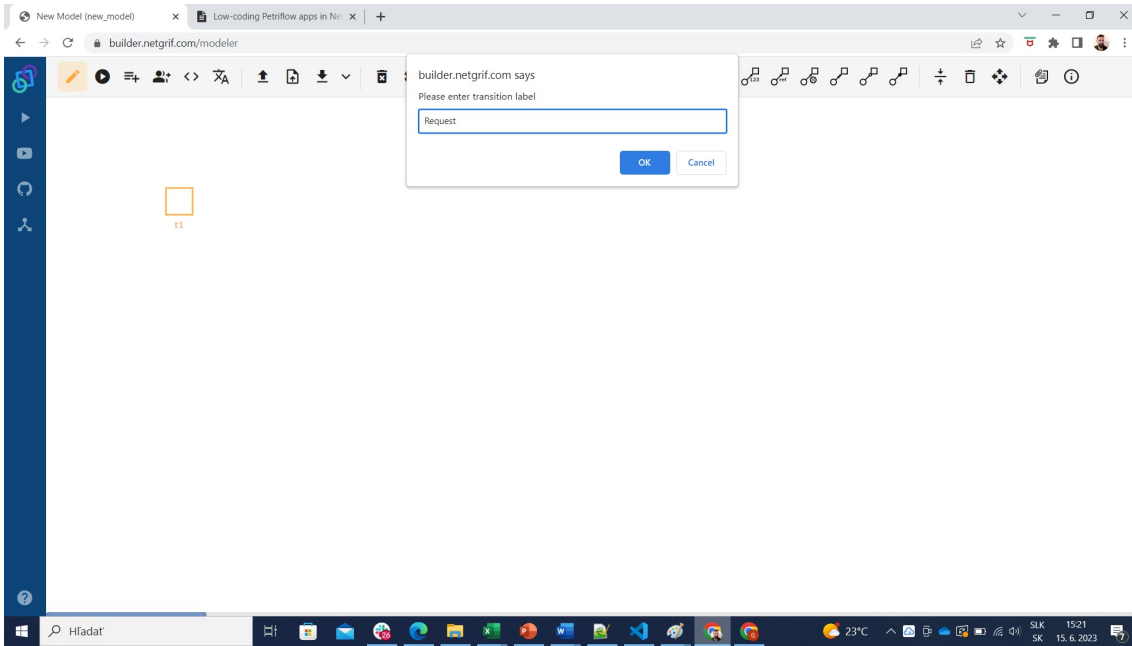
- whenever a user goes to a specific URL in a web browser, new instance of a web form will be displayed in the web browser
- the form will contain text input fields for entering user name, surname, email as required input fields, a text input field for entering phone number and finally a multiline text input field for entering request of the user
- the form should also contain a button for finishing/submitting the form
- a registered user that is deploying the web application at <https://etask.netgrif.cloud/> should see all submitted forms

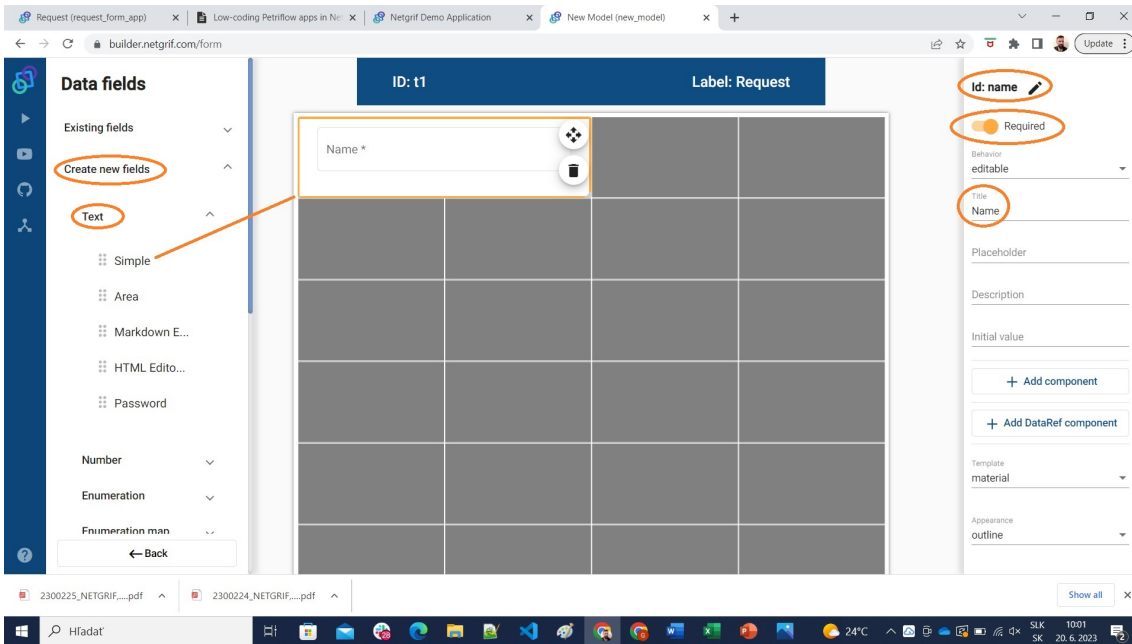
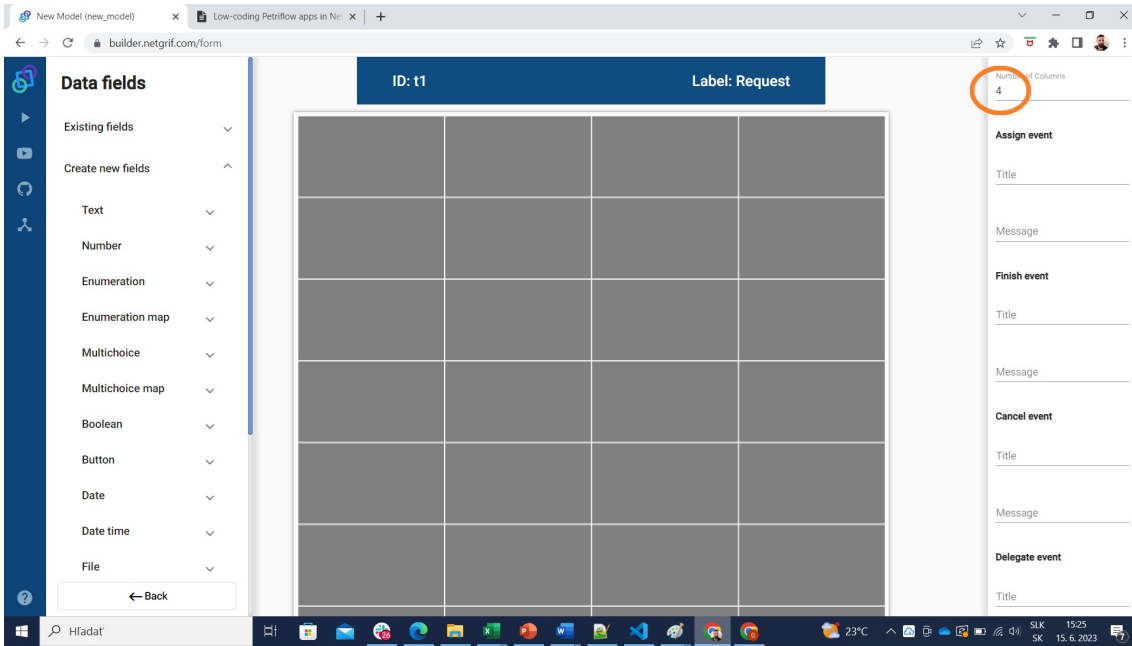
To create the application, we will use Netgrif application builder accesible at <https://builder.netgrif.com> and describe the generated output in low-code language Petriflow.

As we already said, a Petriflow class consists of data attributes, a workflow process that consists of tasks, which can have associated a form with a subset of data attributes.

First we will create a workflow process for the Petriflow class.







The screenshot shows the Petriflow Builder interface. On the left, a sidebar titled "Data fields" contains a list of fields: "Email [email]" (highlighted with an orange circle and an arrow pointing to a field in the form), "Surname [surname]", and "Create new fields" with a "Back" button. The main workspace shows a form with a header "ID: t1" and "Label: Request". The form includes input fields for "Name *" and "Surname *", and a table with 4 columns and 4 rows. The "Email [email]" field is placed in the top-left cell of the table. On the right, there are sections for "Assign event", "Finish event", and "Cancel event", each with a "Title" and "Message" field. The top navigation bar includes "Initials", "Title", "Version", and "Author".

```
<data type="number">
  <id>vratit</id>
  <title>Vrátit</title>
  <init>0</init>
</data>
```