

User Generated Content

Introduction

Content Edit

V1.2

eZ Platform comes with content edition features via the Symfony stack. They are meant, in addition to the PlatformUI's editing capabilities, to allow the implementation of User Generated Content from the frontend.

Usage

Creating a content item without using a draft

As of eZ Platform 1.2, a new `/content/edit/nodraft` route is available. It will show a Content item creation form for a given Content Type:

argument	type	description
<code>contentTypeIdentifier</code>	<code>string</code>	The identifier of the content type to create. Example: <code>folder</code> , <code>article</code>
<code>languageCode</code>	<code>string</code>	Language code the content item must be created in. Example: <code>eng-GB</code>
<code>parentLocationId</code>	<code>integer</code>	ID of the Location the content item must be created in. Example: <code>2</code>

For now a very limited subset of Field Types is supported. These are:

- **V1.2** `TextLine` and `TextBlock`
- **V1.4** `Selection`, `Checkbox` and `User`

More will be added in the near future.

Registering new users

V1.4

You can allow your users to create accounts by employing the `/register` route. This route leads to a registration form that, when filled in, creates a new User Content item in the repository.

User Groups

By default, new Users generated in this way are placed in the Guest accounts group. You can select a different default group in the following section of configuration:

```
eZpublish:
  system:
    default:
      user_registration:
        group_id: <userGroupContentId>
```

In this topic:

- Introduction
 - Content Edit
- Usage
 - Creating a content item without using a draft
 - Registering new users
 - Repository Forms

Registration form templates

You can use custom templates for the registration form and for the registration confirmation page.

These templates are defined with the following configuration:

```
ezpublish:
  system:
    default:
      user_registration:
        templates:
          form:
            'user/registration_form.html.twig'
          confirmation:
            'user/registration_confirmation.html.twig'
```

with the following templates in `app/Resources/views/user/registration_form.html.twig`:

```
{% extends noLayout is defined and noLayout == true ?
viewbaseLayout : pagelayout %}
{% block content %}
    {% import
    "EzSystemsRepositoryFormsBundle:Content:content_form.html.twig" as contentForms %}

    <section class="ez-content-edit">
        {{ contentForms.display_form(form) }}
    </section>
{% endblock %}
```

and in `app/Resources/views/user/registration_confirmation.html.twig`:

```
{% extends noLayout is defined and noLayout == true ?
viewbaseLayout : pagelayout %}
{% block content %}
    <h1>Your account has been created</h1>
    <p class="user-register-confirmation-message">
        Thank you for registering an account. You can
        now <a href="{{ path('login') }}">login</a>.
    </p>
{% endblock %}
```

Repository Forms

- Repository: <http://github.com/ezsystems/repository-forms>
- Package: <http://packagist.org/packages/ezsystems/repository-forms>

This package provides form-based interaction for the Repository Value objects.

It is currently used by:

- `ezsystems/platform-ui-bundle` for most management interfaces: Sections, Content

Types, Roles, Policies, etc.

- `ezytems/ezpublish-kernel` for user registration, and user generated content