

Plone and Volto in a Jamstack Project

Plone Conference 2020

Asko Soukka

7.12.2020



JYVÄSKYLÄN YLIOPISTO
UNIVERSITY OF JYVÄSKYLÄ





Opiskelijaksi avoimeen

Opiskelet sitten Jyväskylässä, Helsingissä, yhteistyöoppilaitoksessa tai etänä, me tuemme polkuasi opiskelijana.

Tälle sivulle on koottu ohjeita, joiden avulla pääset alkuun opinnoissasi. Jos sinulle jää jotain kysyttävää, ota yhteyttä [opintopalveluihimme](#).

Syötä tekstiä...

Opintojen suunnittelu

Kun olet tutustunut itseäsi kiinnostavaan oppiaineeseen ja tarjolla oleviin opintojaksoihin, suunnittele opinnot omaan aikatauluusi sopivaksi. Avoimessa yliopistossa päätät itse opintojesi etenemisestä ja voit aikatauluttaa opintojasi vapaasti suoritusoikeusaikasi sisällä. Huomioi suunnittelussa opintojaksojen tarjonnan aikataulut.

Lue lisää [opintojen suunnittelusta](#) ja [suoritusoikeudestasi](#).



PERUSTIEDOT

Nimike ● Opiskelijaksi avoimeen

Kuvaus

Käytetään sisältöjen listauksissa ja hakutuloksissa.

Browse... No file selected.

Nostokuva

opiskelijaksi-
avoimeen1.jpg



**Nostokuvan
teksti**

Muutoskommer

Tallennettavia muutoksia kuvaava kommentti, joka näytetään sisällön muutoshistoriassa.



[Opinto-opas 2020–23](#) > [Avoin yliopisto](#) > [Opiskelijaksi avoimeen](#)

Avoin yliopisto

[Opintotarjontamme](#)

[Opiskelijaksi avoimeen](#)

[Opiskelutavat](#)

[Ota yhteyttä](#)

Opiskelet sitten Jyväskylässä, Helsingissä, yhteistyöoppilaitoksessa tai etänä, me tuemme polkuasi opiskelijana.

Tälle sivulle on koottu ohjeita, joiden avulla pääset alkuun opinnoissasi. Jos sinulle jää jotain kysyttävää, ota yhteyttä [opintopalveluihimme](#).

Opintojen suunnittelu

Kun olet tutustunut itseäsi kiinnostavaan oppiaineeseen ja tarjolla oleviin opintojaksoihin, suunnittele opinnot omaan aikatauluusi sopivaksi. Avoimessa yliopistossa päätät itse opintojesi



Asko Soukka

Software architect at University of Jyväskylä Digital Services

Background

- Python developer since 2002
- Plone developer since 2004
- Full-time professional since 2008
- GatsbyJS user since 2018



In the beginning...

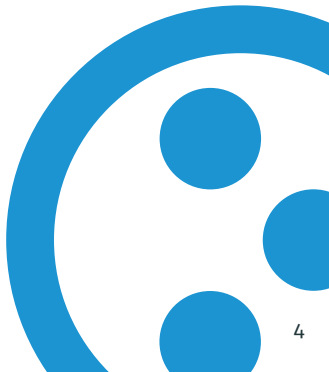


One course management system to rule them all, but...

- every organisation shall do their own integrations
- using granular REST API with deep JSON responses

Branded study guide DIY

- data in tables with JSONB column per endpoint
- custom API with database views and Hasura
- crafted with GatsbyJS + GraphQL source plugin



Let there be CMS



Starting points

- 3rd party course management system
- limited content authoring options

Requirements

- rich content authoring options
- hierarchical permission management
- merge structural data with custom content
- fast and responsive end-user experience

Solution

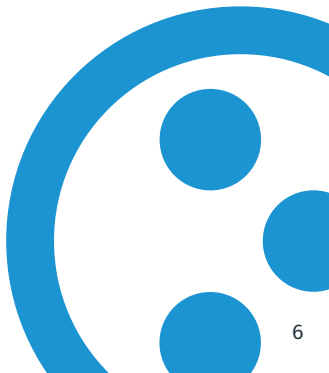
<https://www.avoin.jyu.fi/opinto-opas/fi/avoinyo/>

Ingredients

- Plone 5.2
- Volto
- GatsbyJS

Numbers

- 6 000 HTML pages
- × two languages
- 760 Volto pages



Plone: No-Code Content-Type Customizations

Page (Document)

Overview

Fields

Behaviors

Default

Sisu ja opinto-opas

Kategorisointi

Layout

Add new fieldset...

Add new field...

Settings

Päivät

Omistaja

Delete fieldset

sisu_ids - Multiple Choice

Settings... x

Sivu Sisu-sisältö, jonka yhteydessä sisältö näytetään opinto-oppaassa

sisu_topic - Choice

Settings... x

Välilehti Opinto-oppaan välilehti, jolla sisältö näytetään

Ei arvoa

Done

Save Defaults

Edit XML Field Model

Sivuston asetukset

General

- Actions
- Date and Time
- Kieli
- Posti
- Navigointi
- Sivusto
- Lisäosat
- URL Management
- Hae
- Discussion
- Teemoitus
- Social Media
- Uutisvirta
- TinyMCE

Content

- Sisältösäännöt
- Editointi

Volto: Auto-Complete Widgets with Custom Vocabularies



> Sivu Palikka

SISU JA OPINTO-OPAS

Sivu

AVOINJYU Avoin yliopisto-opetus (2020–2023) x



Sisu-sisältö, jonka yhteydessä sisältö näytetään opinto-oppaassa

Välilehti

Opiskelijaksi avoimeen



Opinto-oppaan välilehti, jolla sisältö näytetään

KATEGORISOINTI

Samasta aiheesta

Samasta aiheesta

Tagit

Select...



Tageja käytetään yleisesti sisällön ad-hoc järjestämiseen.

Kieli

Suomi



GatsbyJS: Query Connected Pages with GraphQL

```
{  
  ...  
  pages: allPloneDocument(filter: {  
    sisu_ids: { in: [$id] },  
    sisu_topic: { token: { eq: "avoin-subject" } }  
  }) {  
    edges {  
      node {  
        ...VoltoDocument  
      }  
    }  
  }  
  ...  
}
```

ReactJS: Render Volto Layouts with React Components

```
import VoltoDocument from '../components/VoltoDocument';  
  
...  
{props.data.pages.edges.map(({ node }) => (  
  <VoltoDocument key={node.id} data={node} />  
))}  
...
```

GatsbyJS recap

GatsbyJS is a ReactJS-based site generator

Why GatsbyJS?

- multi-source plugin architecture
- GraphQL based data lookup
- convention over configuration
- seamless developer experience
- comprehensive documentation



gatsby-source-plone



<https://github.com/collective/gatsby-source-plone/>

Features

- supports default types and most TTW types
- supports files and images from default types
- supports richtext from default types
- resolves linked objects from default types
- incremental updates by modification date



Basic Configuration

```
{  
  resolve: 'gatsby-source-plone',  
  options: {  
    baseUrl: 'https://plonedemo.kitconcept.com/en',  
  },  
},  
{  
  resolve: 'gatsby-source-filesystem',  
  options: {  
    path: `${__dirname}/src/static`,  
  },  
},
```

Advanced Configuration

```
{
  resolve: 'gatsby-source-plone',
  options: {
    baseUrl: 'https://plonedemo.kitconcept.com',
    searchParams: {
      path: [
        '/en/'
      ],
    },
    token: process.env.PLONE_TOKEN,
  },
},
```

Explorer

x

GraphiQL



Prettify

History

Explorer

Code Exporter

< Docs

query MyQuery

- ▶ SISU
- ▶ allDirectory
- ▶ allFile
- ▶ allGraphQLSource
- ▶ allImageSharp
- ▶ allPloneBreadcrumbs
- ▶ allPloneDocument
- ▶ allPloneFile
- ▼ allPloneFolder
 - filter:
 - ▶ UID:
 - ▶ _backlinks:
 - ▶ _components:
 - ▼ _id:
 - ☒ eq: "historia"
 - ☐ glob:
 - ☐ in:
 - ☐ ne:
 - ☐ nin:
 - ☐ regex:
 - ▶ _parent:
 - ▶ _path:
 - ▶ _type:
 - ▶ allow_discussion:
 - ▶ batching:
 - ▶ children:
 - ▶ created:

```

1 query MyQuery {
2   allPloneFolder(filter: {_id: {eq: "historia"}}) {
3     edges {
4       node {
5         _id
6         _type
7         _path
8         title
9         nodes {
10          ... on PloneDocument {
11            _id
12            _type
13            _path
14          }
15          ... on PloneFolder {
16            _id
17            _type
18            _path
19            nodes {
20              ... on PloneFolder {
21                _id
22                _type
23                _path
24              }
25            }
26          }
27        }
28      }
29    }
30  }
31 }
32

```

```

{
  "data": {
    "allPloneFolder": {
      "edges": [
        {
          "node": {
            "_id": "historia",
            "_type": "Folder",
            "_path": "/opintotarjonta/oppiaineet/historia/",
            "title": "Historia",
            "nodes": [
              {
                "_id": "opintojaksot",
                "_type": "Folder",
                "_path": "/opintotarjonta/oppiaineet/historia/opintojaksot/",
                "nodes": [
                  {
                    "_id": "hisa1001-aate-jaksitehistoria-poliittinen-kulttuuri-5-op",
                    "_type": "Folder",
                    "_path": "/opintotarjonta/oppiaineet/historia/opintojaksot/hisa1001-aate-jaksitehistoria-poliittinen-kulttuuri-5-op/"
                  },
                  {
                    "_id": "hisa1002-arjen-kulttuurihistoria-5-op",
                    "_type": "Folder",
                    "_path": "/opintotarjonta/oppiaineet/historia/opintojaksot/hisa1002-arjen-

```

Not Only Rainbows and Unicorns

Full “GatsbyJS experience” requires practice

- query connected pages, images and files
- replace inline links with GatsbyJS links
- replace inline images with GatsbyJS images
- replace file links with direct download

Using @plone/volto seemed like a good idea...

- required Webpack overrides to be importable
- could not be used for images and links

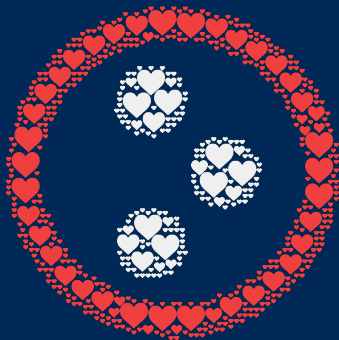


Discussion



GatsbyJS – the ugly parts

- GraphQL source plugin cannot cache
- build may take hours
- build may take gigabytes of memory
- build result is readonly
- monorepo is painful to follow



datakurre.github.io/ploneconf2020