

YellowPages Blog

วิธีสร้างหมวดหมู่ขึ้นมาใหม่ต้องเตรียมข้อมูลในส่วนของ Tags และ Content Type

- Tags (Terms)

เข้าไปที่หมวดหมู่ในส่วนของโครงสร้างจากนั้นหมวดหมู่เพื่อทำการเพิ่มหมวดหมู่

หมวดหมู่

โดยสามารถเพิ่ม Terms ภายใน Tags ที่เป็นชุด Vocabulary หลัก หรือสร้างชุด Vocabulary ขึ้นมาใหม่ก็ได้ (ในตัวอย่างได้ทำการสร้าง YP Blog ขึ้นมาโดยการสร้าง Vocabulary เพิ่มนั้นจะมีผลต่อการสร้าง Content Type นิดหน่อย และการแสดง tags ในหน้า Views) ถ้าต้องการสร้างกลุ่มขึ้นมาใหม่ให้กด [Add vocabulary](#) ใส่ชื่อ และคำอธิบายจากนั้นกดบันทึก

หน้าแรก » Administration » Taxonomy » หมวดหมู่

ชื่อ *

คำอธิบาย

Vocabulary language

ไทย

TERM LANGUAGE

Default language

Site's default language (Thai)

Explanation of the language options is found on the languages list page.

Show language selector on create and edit pages

Enable translation

บันทึก

ถ้าไม่ต้องสร้างแล้วจะเพิ่ม Terms เข้าไปเลยในชุด Vocabulary ที่มีอยู่แล้ว ให้กดเข้าไปใน list Terms ปุ่มที่อยู่ในส่วนของ Operations ในชุด Vocabulary ที่ต้องการ

Tags List terms

YP blog List terms

เมื่อกดเข้ามาจะสามารถเพิ่ม Term ได้โดยการกดที่ [Add term](#) จากนั้นจะเข้ามาในหน้ากรอกข้อมูล โดยกรอกข้อมูล ชื่อ และกำหนด URL ด้วยแล้วบันทึก

หน้าแรก » Administration » โครงสร้าง » หมวดหมู่ » [เป็นการกำหนด URL เฉพาะให้กับธีม](#)

ชื่อ *

/blogterm/knowledge

จากนั้นกดบันทึก เมื่อกลับมาจะขึ้น Term ตามที่บันทึกไป

หน้าแรก » Administration » โครงสร้าง » หมวดหมู่ » Edit YP blog

Add term

You can reorganize the terms in YP blog using their drag-and-drop handles, and group terms under a parent term by sliding them under and to the right of the parent.

ชื่อ	Operations
+ Knowledge	แก้ไข ▾
+ Life Style	แก้ไข ▾
+ Marketing	แก้ไข ▾
+ News	แก้ไข ▾
+ Seo Tip	แก้ไข ▾

บันทึก Reset to alphabetical

- Content Type

เข้าไปที่โครงสร้างจากนั้นเข้าที่ชนิดเนื้อหาเพื่อสร้างหมวดหมู่ใหม่

ชนิดเนื้อหา

กดที่ [เพิ่มชนิดเนื้อหา](#) เพื่อสร้างหมวดหมู่ใหม่โดยในการใส่ชื่อจำเป็นต้องแก้ไขชื่อสำหรับอ้างอิงด้วย เมื่อใส่ชื่อหมวดหมู่แล้ว ให้ทำการแก้ไขชื่ออ้างอิงที่โชว์อยู่ด้านข้างด้วย โดยกดที่ แก้ไข ชื่อสำหรับอ้างอิง: movie [แก้ไข] โดยในการตั้งชื่อสำหรับอ้างอิงให้ใส่คำว่า **yp_blog_** ไว้ข้างหน้าด้วย (มีผลในส่วนต่อไป ถ้าลืมหรือผิดต้องสร้างใหม่เท่านั้น) ชื่อสำหรับอ้างอิง * ชื่อสำหรับอ้างอิง * จากนั้นกด Save and manage Fields จะเข้าหน้าเว็บที่จะสามารถเพิ่มช่องกรรับข้อมูลได้ของ Content Type โดยกดที่ [เพิ่มช่องรับข้อมูล](#) โดยช่องรับข้อมูลที่ต้องเพิ่มมี

1. Banner ให้เลือกเป็น image: field_banner

หน้าแรก » Administration » โครงสร้าง » ชนิดเนื้อหา » Movie » จัดการช่องรับข้อมูล

Add a new field หรือ Re-use an existing field

ป้ายชื่อ *

[บันทึกและไปต่อ](#)

พอกดบันทึกแล้วไปต่อจะเข้าหน้าตั้งค่าช่องรับข้อมูลโดยให้ตั้ง

ค่าเป็น Required field และตั้งขนาดรูปภาพขั้นต่ำไว้ที่

Minimum image resolution × พิกเซล

จากนั้นกดบันทึกค่า

2. Tags ให้เลือกเป็น Entity reference: field_tags

หน้าแรก » Administration » โครงสร้าง » ชนิดเนื้อหา » Movie » จัดการช่องรับข้อมูล

Add a new field Re-use an existing field

- Select a field type - หรือ Entity reference: field_tags

ป้ายชื่อ *

Tags

บันทึกและไปต่อ

เมื่อเข้าหน้าตั้งค่าช่องรับข้อมูลให้ตั้งค่าเป็น Required field จากนั้นเลื่อนลงมาด้านล่างก่อนปุ่มบันทึกในส่วนของ Reference Type ตรงคำศัพท์ ถ้าในส่วนของ Tags (Terms) ถ้าได้ทำการสร้างมาก่อนก็จะมีขึ้นให้เลือกด้วย โดยการเลือกตรงส่วนนี้จะเป็นการกำหนดหมวดหมู่ที่สามารถเลือกได้ตอนที่เพิ่มเนื้อหาในหน้า Content และเลือก **Create referenced entities if they don't already exist** คือถ้าตอนที่เพิ่ม Content แล้วถ้าเราเพิ่ม tags ที่ไม่ได้ทำการเพิ่มไว้ในชุด Vocabulary ที่เลือกตัวระบบ Drupal ก็จะทำการเพิ่มให้เองด้วยอัตโนมัติ

REFERENCE TYPE

Reference method *

Default

Create referenced entities if they don't already exist

คำศัพท์ *

Tags

YP blog

เลือกเสร็จจากนั้นกดบันทึกค่า

3. Video ให้เลือก Text (plain): field_video_link

หน้าแรก » Administration » โครงสร้าง » ชนิดเนื้อหา » Movie » จัดการช่องรับข้อมูล

Add a new field หรือ Re-use an existing field

ป้ายชื่อ *

[บันทึกและไปต่อ](#)

กดบันทึกค่าแล้วไปต่อ แล้วก็บันทึกค่าได้เลยไม่มีการแก้ไข

[เพิ่มช่องรับข้อมูล](#)

ป้ายชื่อ	ชื่อสำหรับอ้างอิง
Body	body
New Banner	field_banner
Tags	field_tags
Video URL	field_video_link

เมื่อเพิ่มครบทั้ง 3 ตัว จะได้อย่างในรูปแบบ ถือเป็นอันเสร็จสิ้นการสร้าง

หมวดหมู่ ขึ้นมา 1 หมวด

ตัวอย่างการเพิ่มข้อมูล Content 1 ตัว

1. กดเพื่อเนื้อหา
2. เลือก Content Type [Movie](#)
3. เพิ่มหัวข้อ เนื้อหาและรูปภาพตามปกติ (Video URL ใส่เป็นลิงค์)
4. ในส่วนของ Tags ใส่ข้อความตามที่เคยเพิ่ม แล้วเลือกให้ขึ้นเลข tags ด้วย หรือใส่ tags เองจากนั้นต้องไปตั้ง URL ด้วย

Tags * [Movie](#) [เพิ่มรายการอื่น](#)

Tags * [เพิ่มรายการอื่น](#)

5. กำหนด url /blog/หมวดหมู่ (ในนี้คือmovie) /ชื่อเนื้อหาหรืออื่น ๆ

เป็นการกำหนด URL เฉพาะให้กับธีม

เป็นการกำหนด URL เฉพาะให้กับธีม


Specify an alternative path by which this data can be accessed. For example, type "/about" when writing an about page.

6. จากนั้นกดบันทึก

Yellow Blog

หมวดหมู่: SEO, Marketing, Life style, Knowledge, News, See all

หน้าหลัก / Movie / avenger



Movie

By ผู้ดูแลระบบของเว็บไซต์ ๒๑ Admin | วันที่ August 27, 2021 | ผู้เข้าชมของเว็บไซต์ ๒๑ ผู้

0 0 0 0 0

avenger

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Pellentesque nec justo non magna facilisis luctus vel sed enim. Nullam at pulvinar nibh. Sed fermentum vehicula vulputate. Nunc aliquet, nunc vitae auctor tristique, enim nisi laculis neque, at varius arcu ligula non nulla. Morbi aliquam magna id tempor suscipit. Cras eget purus nec magna porta consequat et at quam. Proin faucibus commodo sapien, sed imperdiet nunc gravida vel. Vivamus vel cursus odio. Nunc consectetur nibh ut maximus ultrices. Quisque rhoncus massa eget lectus viverra, eget aliquam metus ullamcorper. Suspendisse sodales, sapien in malesuada mattis, mauris quam mattis ipsum, at bibendum ex ligula ut risus. Praesent sed nunc eu urna dignissim sagittis nec sed arcu. Donec pharetra maximus felis, id suscipit orci pellentesque quis. Ut rhoncus tempus ornare.

Suspendisse auctor lacus vel est pulvinar, eu vulputate ligula congue. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Suspendisse potenti. Sed sagittis dictum consequat. Ut malesuada massa sit amet erat tristique, nec malesuada risus iaculis. Aliquam eu felis non elit dictum mollis. Fusce quis mattis orci, non ultrices sapien. Etiam egestas justo quis nulla efficitur, eget venenatis mauris tincidunt. Donec vel pharetra ligula. Nam mollis quam tellus, eget pretium orci congue sed. Sed commodo a arcu in ornare. Proin porttitor accumsan est, a ultrices lectus fermentum ut. Donec volutpat quis justo et tincidunt. Vestibulum tincidunt velit vitae elit condimentum, vel tincidunt mi cursus. Aliquam dictum sed ligula eget tincidunt. Sed efficitur eros eget turpis sodales, sed efficitur ipsum porta.

Praesent tempus erat sit amet magna varius condimentum. Duis congue elementum nulla, ac faucibus ante dignissim a. Suspendisse sodales eget libero eget volutpat. Etiam efficitur sodales semper. Vivamus et urna iaculis, dignissim ante nec, facilisis magna. In eleifend tempus metus laoreet porttitor. Nullam id tristique ligula. Nunc efficitur tortor suscipit elit tincidunt, sed finibus erat congue. Mauris eget suscipit velit.

In maximus tellus vel risus pulvinar aliquam. Sed pretium turpis justo, nec pulvinar est semper in. Fusce a eros porttitor, aliquam tellus vitae, rhoncus lorem. Nam consectetur tincidunt orci, nec scelerisque quam. Suspendisse euismod tincidunt odio non malesuada. In rhoncus turpis a nunc pharetra accumsan. Suspendisse tempus eget turpis nec tempor. Praesent tincidunt aliquet lobortis.

Interdum et malesuada fames ac ante ipsum primis in faucibus. Sed congue ullamcorper lectus, a venenatis augue convallis sed. Cras nec pretium lectus. Donec tortor magna, consectetur at pulvinar sit amet, ultrices id neque. Suspendisse et sollicitudin tellus, ut tristique mauris. Donec condimentum metus ut nunc hendrerit pharetra. Donec nec ex malesuada, accumsan mi vitae, aliquam purus. Aenean fringilla diam eleifend, sollicitudin urna at, tempor velit. Nulla imperdiet interdum purus, ut placerat metus sagittis nec.

Knowledge | Life Style | Marketing | Movie | News | Seo Tip

Yellow Pages Digital Media Solutions

ผู้ดูแลระบบของเว็บไซต์

We are a provider of business listing information including products and services that covers many categories which every business has passed the examination and confirmed that they have real identity throughout the country.

เกี่ยวกับเรา

- Corporate History
- ประวัติความเป็นมา
- Cyber Policy
- Cookies Policy
- นโยบายการใช้งาน
- เกี่ยวกับผู้ดูแลระบบ
- Global YellowPages Network
- COVID-19

Our products

- โฆษณาแบบโต้ตอบ
- Yellow.co.th

ติดต่อเรา

- พื้นที่บริการของเรา
- โฆษณาของเรา
- ฝ่ายบริการลูกค้าสัมพันธ์
- โทรฟรี

Download Application

Download on the App Store

GET IT ON Google Play

Teleinfo 11.2 ผู้ดูแลระบบของเว็บไซต์

© 2564 โฆษณาแบบโต้ตอบ บริการที่มีชีวิตชีวาทุกสาขา โดย บริษัท เทคโนโลยี ดิจิทัล จำกัด (มหาชน)

DSD Comodo

วิธีการสร้าง Views และ Context เพื่อกำหนดการแสดงผลของข้อมูลภายในหน้า

- Views

Views ต้องสร้างสองตัว ตัวแรกไว้เป็นหน้า Page แสดงเนื้อหาปกติ และอีกตัวไว้เป็นแสดง บทความแนะนำ (ในตัวอย่างคือ YP Blog และ YP Blog R)

YP Blog	Block Page (/blog) Page (/blog/knowledge) Page (/blog/lifestyle) Page (/blog/marketing) Page (/blog/news) Page (/blog/see-all) Page (/blog/seo_tip)	แก้ไข ▾
YP Blog R	Block Block Block Block	แก้ไข ▾

วิธีการสร้าง Views ทั้งสองตัว เริ่มที่กด

[Add view](#)

ใส่ชื่อใน ข้อมูลพื้นฐาน แล้วบันทึกได้เลย

ดูข้อมูลพื้นฐาน

ดูชื่อ *

YP Blog

ชื่อสำหรับอ้างอิง: yp_blog [แก้ไข]

ดูข้อมูลพื้นฐาน

ดูชื่อ *

YP Blog R

ชื่อสำหรับอ้างอิง: yp_blog_r [แก้ไข]

- YP Blog

เมื่อเข้าหน้าแก้ไขกด

[+ เพิ่ม](#)

จากนั้นเลือก

[Page](#)

ตั้งชื่อที่แสดงเพื่อให้ง่ายต่อการจัดการ ชื่อที่แสดง: [Movie](#) จากนั้น

ตั้งค่าหัวข้อไว้เป็นชื่อแสดงในหน้าเว็บ

หัวข้อ

[หัวข้อใหม่](#)

และต้องเลือกในส่วน

สำหรับด้วย โดยให้เลือกเป็น

สำหรับ

[นี่คือ page \(override\)](#)

เพื่อให้ไม่ไปทับกับ

หน้าอื่น ๆ จากนั้นกดใช้ (แสดงผลนี้) และรูปแบบให้เลือกเป็น ตะแกรง

ที่จะวาง Content เป็นช่อง ๆ และสามารถปรับได้ด้วยการทำให้แสดงที่ Column โดยตั้งค่า ให้เป็น เอาที่ Default column classes และ Default row classes ออก และใส่

- Custom column class เป็น
col-md-4 col-sm-4 col-xs-4 margin-bottom-fifty padding-top-ten aos-eff
- Custom row class เป็น
col-md-12 col-sm-12 col-xs-12 no-gutter

จากนั้นกด Apply

ในส่วนของ fields ที่ดึงข้อมูลมาโชว์ให้เพิ่มมีดังนี้

FIELDS



- Content: Video URL [ซ่อน] **1**
- Content: Banner **2**
- Content: หัวข้อ **3**
- Content: Body **4**
- Content: Tags **5**
- Content revision: โพสต์โดย **6**
- Content revision: โพสต์เมื่อ **7**
- Content statistics: Total views **8**



Test SEO1 **3**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed sit amet posuere lacus, sed dictum augue. Nullam... **4**

📌 Seo Tip **5**

👤 admin **6**

📅 วันที่ : 19 สิงหาคม 2021 **7**

📄 อ่าน 13 **8**

1. Content : video URL [ซ่อน]
2. Content : Banger
 - ตั้งค่า STYLE SETTINGS และเขียนผลลัพธ์ใหม่ด้วย
 - a. STYLE SETTINGS

STYLE SETTINGS

Customize field HTML

อีลิเมนต์ HTML

DIV ▼

Choose the HTML element to wrap around this field, e.g. H1, H2, etc.

Create a CSS class

คลาส CSS

col-md-12 col-sm-12 col-xs-12 no-gutter

คุณอาจจะใช้แทนโทเค็นจากส่วนการเขียนใหม่ในขั้นนี้

b. เขียนผลลัพธ์ใหม่

```
{% if field_video_link %}<div class="add-video"><i class="fas fa-play-circle"></i> {{ field_banner}}</div>{% else %}{{ field_banner}}{% endif %}
```

เขียนผลลัพธ์ใหม่

Override the output of this field with custom text

ข้อความ

```
{% if field_video_link %}<div class="add-video"><i class="fas fa-play-circle"></i> {{ field_banner}}</div>{% else %}{{ field_banner}}{% endif %}
```

3. Content : หัวข้อ ตั้งค่า STYLE SETTINGS

a. STYLE SETTINGS

STYLE SETTINGS

Customize field HTML

อีลิเมนต์ HTML

H2 ▼

Choose the HTML element to wrap around this field, e.g. H1, H2, etc.

Create a CSS class

คลาส CSS

col-md-12 col-sm-12 col-xs-12 no-gutter field-header

คุณอาจจะใช้แทนโทเค็นจากส่วนการเขียนใหม่ในขั้นนี้

4. Content : Body

- ตั้ง รูปแบบ Formatter , Trimmed limit และ ตั้งค่า STYLE SETTINGS

a. Formatter

Formatter

b. Trimmed limit

Trimmed limit *

 characters

c. STYLE SETTINGS

col-md-12 col-sm-12 col-xs-12 no-gutter text-justify field-body text-truncate

STYLE SETTINGS

Customize field HTML

อีลีเมนต์ HTML

Choose the HTML element to wrap around this field, e.g. H1, H2, etc.

Create a CSS class

คลาส CSS

คุณอาจจะใช้แทนโทเค็นจากส่วนการเขียนใหม่ในขั้นนี้

5. Content : Tags

- ตั้ง รูปแบบ Formatter , ตั้งค่า STYLE SETTINGS และ เขียนผลลัพธ์ใหม่

a. Fomatter

Formatter

Link label to the referenced entity

b. STYLE SETTINGS

STYLE SETTINGS

Customize field HTML

อีลิเมนต์ HTML

- ไข่คำปริยาย - ▼

Choose the HTML element to wrap around this field, e.g. H1, H2, etc.

Create a CSS class

คลาส CSS

field-type field-detail

คุณอาจจะใช้แทนโหนดอื่นจากส่วนการเขียนใหม่ในขั้นนี้

c. เขียนผลลัพธ์ใหม่

เขียนผลลัพธ์ใหม่

Override the output of this field with custom text

ข้อความ

```
<i class="far fa-bookmark"></i> {{field_tags}}
```

6. Content Revision : ไรต์โดย

- ตั้ง Formatter ตั้งค่า STYLE SETTINGS และเขียนผลลัพธ์ใหม่

a. Formatter

Formatter

ป้ายชื่อ ▼

b. STYLE SETTINGS

STYLE SETTINGS

Customize field HTML

อิลิเมนต์ HTML

- ไร้ค่าปริยาย - ▼

Choose the HTML element to wrap around this field, e.g. H1, H2, etc.

Create a CSS class

คลาส CSS

col-md-1 col-sm-1 col-xs-1 text-center no-gutter img-blog

คุณอาจจะใช้แทนโทเค็นจากส่วนการเขียนใหม่ในขั้นนี้

c. เขียนผลลัพธ์ใหม่

เขียนผลลัพธ์ใหม่

Override the output of this field with custom text

ข้อความ

```
<i class="far fa-user"></i> {{ uid_1 }}
```

7. Content Revision : โพลต์เมื่อ

- กำหนด Formatter , รูปแบบวันที่ , ตั้งรูปแบบ Custom date format , เลือกเขต , ตั้ง STYLE SETTINGS และเขียนผลลัพธ์ใหม่

a. Formatter

Formatter

Default ▼

b. รูปแบบวันที่

รูปแบบวันที่

กำหนดเอง ▼

c. Custom date format

Custom date format

วันที่ : j F Y

See [the documentation for PHP date formats](#).

d. เลือกเขต

เขตเวลา

Bangkok

e. STYLE SETTINGS

STYLE SETTINGS

Customize field HTML

อิลิเมนต์ HTML

DIV

Choose the HTML element to wrap around this field, e.g. H1, H2, etc.

Create a CSS class

คลาส CSS

col-md-6 col-sm-6 col-xs-6 text-center field-detail no-gutter

คุณอาจจะใช้แทนโทเค็นจากส่วนการเขียนใหม่ในขั้นนี้

f. เขียนผลลัพธ์ใหม่

เขียนผลลัพธ์ใหม่

Override the output of this field with custom text

ข้อความ

<i class="fts ft-yp-article-calendar"></i> {{created }}

8. Content statistics : Total views

- ตั้ง STYLE SETTINGS และ เขียนผลลัพธ์ใหม่

a. STYLE SETTINGS

STYLE SETTINGS

Customize field HTML

อีลีเมนต์ HTML

DIV ▼

Choose the HTML element to wrap around this field, e.g. H1, H2, etc.

Create a CSS class

คลาส CSS

col-md-3 col-sm-3 col-xs-3 field-detail no-gutter text-align-center

คุณอาจจะใช้แทนโทเค็นจากส่วนการเขียนใหม่ในขั้นนี้

b. เขียนผลลัพธ์ใหม่

เขียนผลลัพธ์ใหม่

Override the output of this field with custom text

ข้อความ

```
<i class="fas fa-book-open"></i> อ่าน {{totalcount }}
```

- คัดกรองตามเงื่อนไข จะเป็นส่วนดึงข้อมูล โดยเลือกเป็นหมวดตามเนื้อหา

คัดกรองตามเงื่อนไข

Content: Published (= ใช่)

Content: ชนิดของเนื้อหา (ใน Knowled...)

- หน้า HOME เลือกทั้งหมด

- Knowledge
- Life Style
- News
- Marketing
- SEO

- หน้าหมวดหมู่เลือกแค่หมวดตัวเอง

- Movie

- กำหนดจำนวนที่จะแสดงแต่หน้า

- 2 แบบ แสดงแบบ fix กับ เปลี่ยนหน้าได้

- Fix

- ตัวเปลี่ยนหน้า

- ใช้ pager: แสดงจำนวนรายการที่ระบุ | Field ที่เสียหาย

- เชื่อมโยงเพิ่มเติม: ไม่

- สำหรับ

นี่คือ page (override) ▼

Items per page

๒

Enter 0 for no limit.

- แสดงจำนวนรายการที่ระบุ
- Display all items
- Paged output, full pager
- Paged output, full pager

Offset (number of items to skip)

0

For example, set this to 3 and the first

- เปลี่ยนหน้าได้

- ตัวเปลี่ยนหน้า

- ใช้ pager: Full | Field ที่เสียหาย

- เชื่อมโยงเพิ่มเติม: ไม่

- สำหรับ

All displays (except overridden) ▼

- แสดงจำนวนรายการที่ระบุ
- Display all items
- Paged output, full pager
- Paged output, full pager

Items per page

๒

Enter 0 for no limit.

PAGER LINK LABELS

First page link text

« First

Previous page link text

< Previous

Next page link text

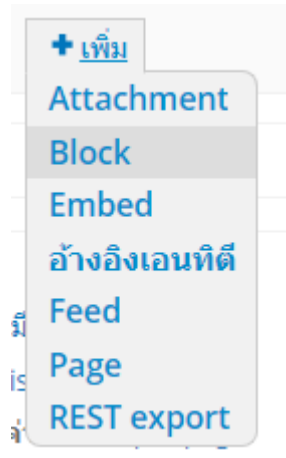
Next >

Last page link text

Last »

- วิธีการสร้าง SlideShow ในระบบ Drupal

- สร้าง Block



- ตั้งชื่อตั้งหัวข้อ

- ชื่อที่แสดง: slideshow
 - หัวข้อ
 - หัวข้อ: บทความล่าสุด

- เลือกรูปแบบการแสดงผลข้อมูล

- รูปแบบ

รูปแบบ: HTML List | ตั้งค่า

สำหรับ
นี่คือ block (override) ▼

- ตะแครง
- HTML List
- ตาราง
- รายการไม่ได้จัดรูปแบบ

- ตั้งค่า

- Row class : *swiper-slide*

Row class
swiper-slide

- List type

List type
 Unordered list
 Ordered list

- Wrapper class : *swiper-container mySwiper*

Wrapper class

swiper-container mySwiper

The class to provide on the wrapper, outside the list.

- List class : *swiper-wrapper*

List class

swiper-wrapper

The class to provide on the list element itself.

■ แสดง ไม่ต้องตั้งอะไรเลย

[แสดง: Fields](#) | [ตั้งค่า](#)

■ แสดงจำนวนแบบระบุ

รายการที่จะแสดง

6

Enter 0 for no limit.

■ FIELDS ที่ดึงมา

FIELDS

[Content: Banner](#)

[Content: หัวข้อ](#)

[Content: Body](#)

- Content หัวข้อ

การกำหนดค่า field: Content: หัวข้อ

สำหรับ

นี่คือ block (override)

สร้างป้ายชื่อ

Exclude from display

Enable to load this field as hidden. Often used to group fields, or to use as token in another field.

- Content Body

คอลัมน์ที่ใช้สำหรับการเรียงลำดับการคลิก
 summary ▼
Used by Style: Table to determine the actual column to click sort the field on. The default is usually fine.

Formatter
 Trimmed ▼

Trimmed limit *
 150 characters
If the summary is not set, the trimmed `node.body` field will end at the last full sentence before this character limit.

STYLE SETTINGS

Customize field HTML

ลิ้มเนต HTML
 - ไม่มี - ▼
Choose the HTML element to wrap around this field, e.g. H1, H2, etc.

Create a CSS class

คลาส CSS
 margin-left-sixty padding-left-ten
คุณอาจจะใช้แทนที่แทนที่จากส่วนการเขียนใหม่ในขั้นนี้

เขียนผลลัพธ์ใหม่

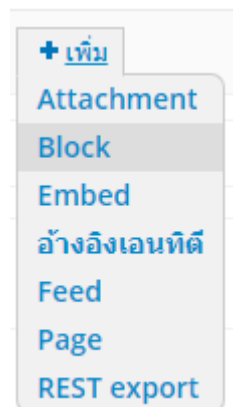
Override the output of this field with custom text

ข้อความ

```
<div class="content">
  {{ title }}
  {{body}}
</div>
```

- YP Blog R

ไว้สำหรับแสดงบทความแนะนำที่ขึ้นข้าง ๆ ในหน้าบทความให้
 สร้าง views ขึ้นมาปกติ แต่ให้เพิ่ม block แทน



สร้างเสร็จให้กำหนดชื่อที่แสดงโดยจะมีผลต่อการในไปใช้ใน Twig

Administrative name
 Seo Tip โดยให้ตั้งตาม Path ของหมวดหมู่ที่เคยตั้งไว้

ตั้งชื่อหัวข้อ หัวข้อ จากนั้นเลือกรูปแบบ รายการไม่จัดรูปแบบ

บทความแนะนำ จากนั้นเลือกรูปแบบ รายการไม่จัดรูปแบบ

แล้วตั้งค่า Row Class

- col-md-12 col-sm-12 col-xs-12 margin-bottom-ten no-gutter

Grouping field Nr.1

You may optionally specify a field by which to group the records. Leave blank to not

Row class

The class to provide on each row. You may use field tokens from as per the "Replace

- Add views row classes

Add the default row classes like views-row to the output. You can use this to quic

และเลือก FIELDS ข้อมูลตามนี้

FIELDS

[Content: Banner](#)

[Content: หัวข้อ](#)

[Content: Body](#)

- Content : Banner สร้างได้เลย

- Content : หัวข้อ

เลือก Formater เป็น Plain Text และตั้งค่า STYLE SETTINGS

STYLE SETTINGS

- Customize field HTML

อิลิเมนต์ HTML

Choose the HTML element to wrap around this field, e.g. H1, H2, etc.

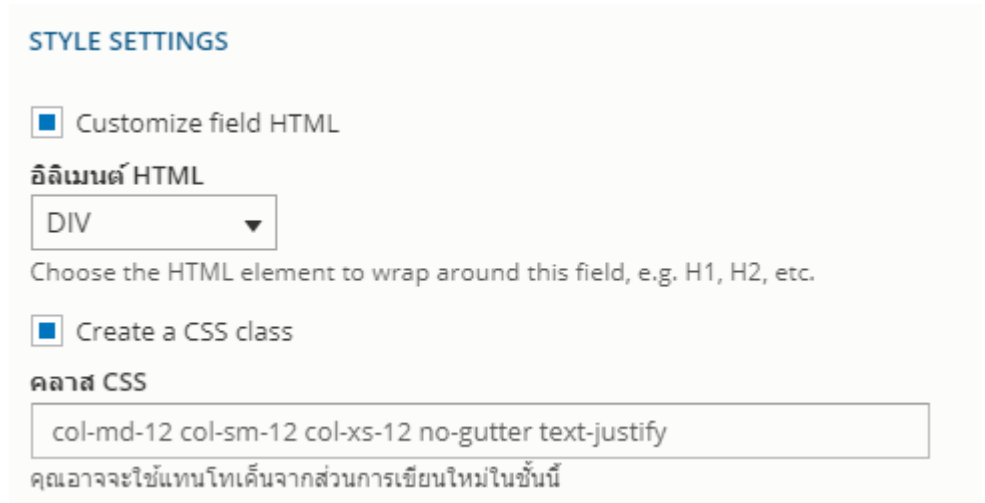
- Create a CSS class

คลาส CSS

คุณอาจจะใช้แทนโทเค็นจากการเขียนใหม่ในขั้นนี้

- Content : Body

เลือก Formatter Trimmed Limit ไว้ที่ 150 และตั้งค่า STYLE SETTINGS



คัดกรองตามเงื่อนไข จะเป็นส่วนดึงข้อมูล โดยเลือกเป็นหมวดตามเนื้อหา

คัดกรองตามเงื่อนไข

Content: Published (= ใช่)

Content: ชนิดของเนื้อหา (ใน Knowled...)

หน้าหมวดหมู่เลือกแค่หมวดตัวเอง

Movie **ตัวเปลี่ยนหน้า**

ใช้ pager: แสดงจำนวนรายการที่ระบุ | Field ที่เสียหาย

เชื่อมโยงเพิ่มเติม: ไม่

แสดงการเชื่อมโยง: ไม่มี

เลือกจำนวนตรงตัวเปลี่ยนหน้า

โดยเลือกเป็น แสดงจำนวนรายการที่ระบุ

จากนั้นตั้งค่า ขึ้นสูงในส่วนของ ชื่อของเครื่อง: โดยให้ตั้งตาม url ใน YP blog แต่ถ้ามี 2 บรรทัด ต้องเชื่อมด้วย _ อย่างในรูป

ขึ้นสูง
CONTEXTUAL FILTERS
ความสัมพันธ์
แบบฟอร์มการสัมผัส
ลักษณะแบบฟอร์มExposed: พื้นฐาน ตั้งค่า
อื่นๆ
ชื่อของเครื่อง: seo_tip

ขึ้นสูง
CONTEXTUAL FILTERS
ความสัมพันธ์
แบบฟอร์มการสัมผัส
ลักษณะแบบฟอร์มExposed: พื้นฐาน ตั้งค่า
อื่นๆ
ชื่อของเครื่อง: marketing

วิธีการสร้าง Context สำหรับจับส่วน ต่าง ๆ มารวมกันและการดึงไฟล์ Twig

Context

กดสร้าง ใส่ชื่อและกลุ่ม จากนั้นกดบันทึก

GENERAL DETAILS

ป้ายชื่อ *

 ชื่อสำหรับอ้างอิง: yp_blog [แก้ไข]
Enter label for this context.

Group

 Enter name of the group this context should belong to.

คำอธิบาย

Enter a description for this context definition.

บันทึกเสร็จจะเข้าหน้าตั้งค่า ตรง Conditions ต้องเพิ่ม Request Path โดยกดที่ Add Condition จากนั้นใส่คำว่า Request Path เลือกตัวที่เป็น (system)

Request Path

เงื่อนไข

Request path exclusion (context)

Request Path (system) ←

จากนั้นใส่

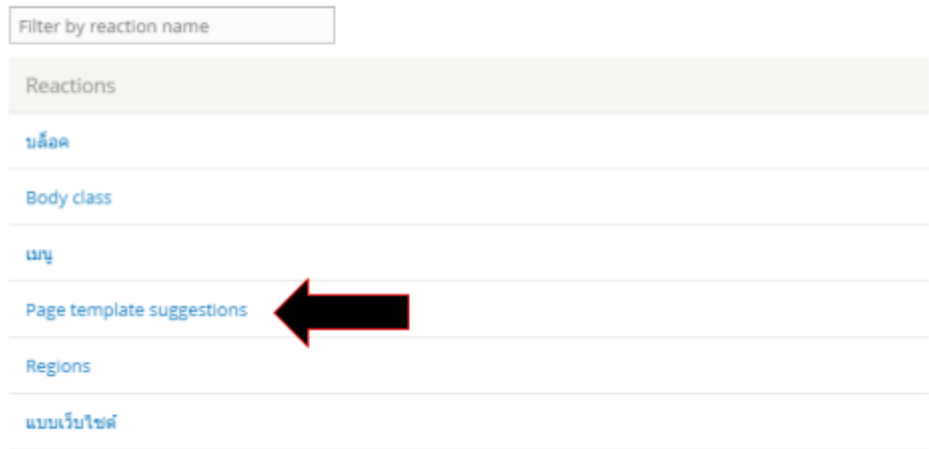
/blog
/blog/*
/blogterm/*

หน้า

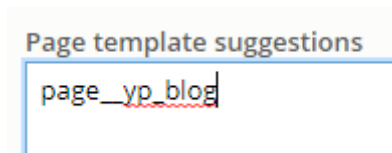
เพื่อให้ path ที่กำหนดสามารถ ดึงไฟล์ Twig ได้

จากนั้นในส่วนของ Reactions ต้องเลือก 3 อย่าง

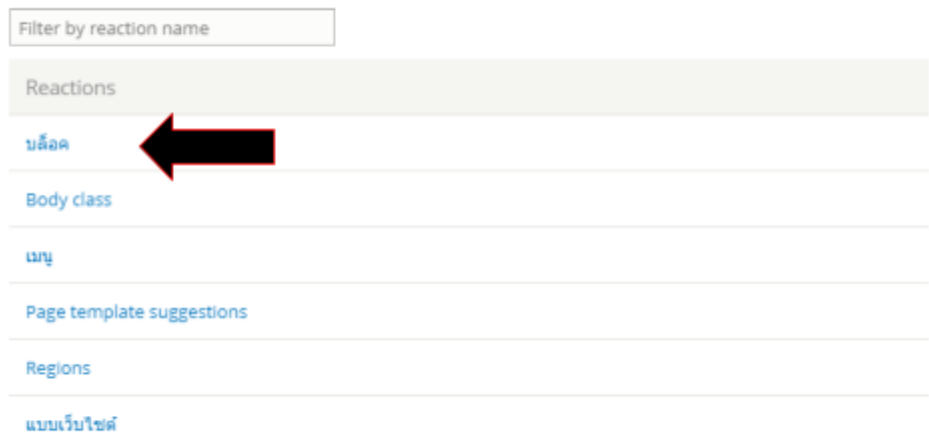
- Page Template Suggestions



จากนั้นใส่ชื่อไฟล์ Twig โดยใช้เป็น page__yp_blog



- บล็อก จะเป็นส่วนที่ดึงข้อมูลจากที่ต่าง ๆ เช่น Views หรือ Tags



โดยเลือกตามที่ขึ้นในรูปได้เลย โดยจะมี 2 ส่วนคือ Content กับ Content Right

Content	Content Right
✚ YP Blog: slideshow	✚ YP Blog R: Marketing
✚ Page title	✚ YP Blog R: Knowledge
✚ Main page content	✚ YP Blog R: Life style
✚ Tags in YP blog	✚ YP Blog R: News
	✚ YP Blog R: Seo Tip

ทุก Block ใน Content ไม่จำเป็นต้องเลือก แสดงชื่อเรื่อง

- ตัว Tags in YP Blog ตั้งค่าตามรูป คือจะให้แสดง แค่ 10 ตัว ตั้ง Vocabulary Machine name ตามที่สร้างไว้โดย ใช้รหัสอ้างอิง

Block description

Tags in YP blog

หัวข้อ *

Tags in YP blog

แสดงชื่อเรื่อง

Tags to show

10

The number of tags to show in this block. Enter '0' to display all tags.

Vocabulary machine name

yp_blog

ตำแหน่ง

Content

ชื่อสำหรับอ้างอิง: yp_blog

เลือกเขตที่บล็อกนี้ควรจะถูกแสดง

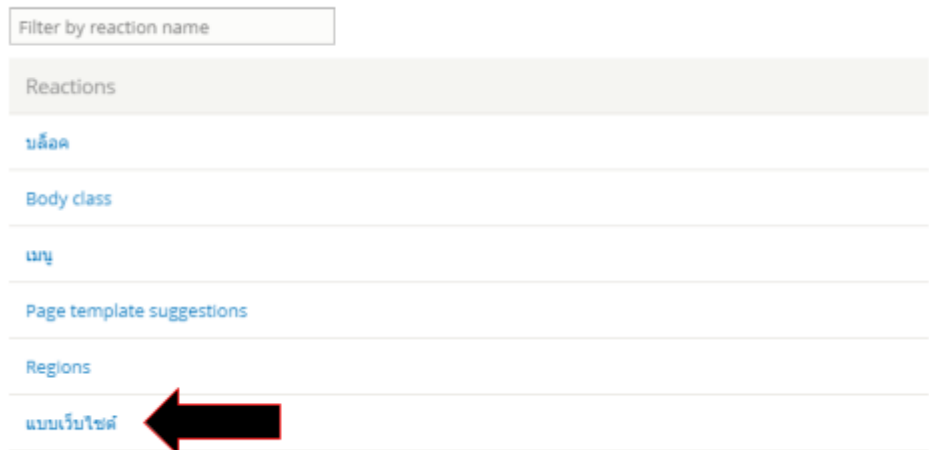
Unique

Check if the block should be uniquely placed, this means that the block can not be

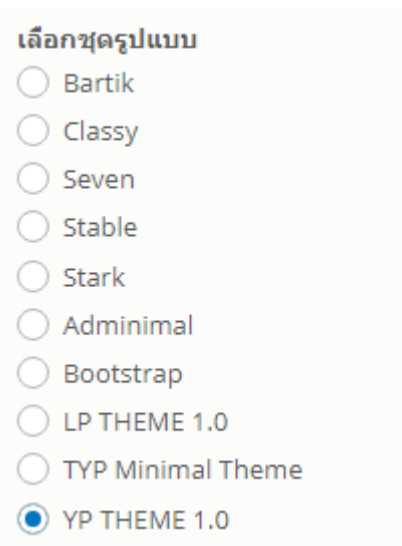
Block Class

col-md-12 col-sm-12 col-xs-12 tag-taxonomy no-gutter

- แบบเว็บไซต์



จากนั้นเลือก



เสร็จหมดแล้วกด บันทึก

พอเตรียม VIEWS CONTEXT เสร็จแล้วก็ทำหลังบ้าน โดยต้องสร้างไฟล์สำหรับเขียน HTML.TWIG, .JS และ .Scss

[Code](#)

และเพิ่ม path ไฟล์ css และ js รวมทั้ง aos-master ไว้ใช้ในการแสดงผล

ภายในไฟล์ yptheme.libraries.yml

```
yp-blog:  
  js:  
    js/yp-blog.js: {}  
  css:  
    theme:  
      css/theme/yp-blog.css: {}  
      css/fonts/helvethaica/helvethaica.css: {}  
  dependencies:  
    - typ_contrib/aos-master
```

จากนี่จะเป็นวิธีการทำให้สามารถดึงไฟล์ CSS และ JS ได้ถูกต้อง ในไฟล์ PageAttachments.php ใน function setAttachedLibraryWebContent

```
protected function setAttachedLibraryWebContent(array $pathSlot, &$attachments)
{
    try {

        $node = \Drupal::routeMatch()->getParameter('node');
        $route = \Drupal::routeMatch()->getRouteName();
        if ($node) {
            $nodeBundle = $node->bundle();
            if ($nodeBundle === 'typ_app') {
                $attachments['#attached']['library'][] = 'typ_contrib/swiper4';
                $attachments['#attached']['library'][] = 'yptHEME/app-store';
            } elseif ($nodeBundle === 'typ_covid19') {
                $attachments['#attached']['library'][] = 'typ_contrib/datatable';
                $attachments['#attached']['library'][] = 'typ_contrib/swiper4';
                $attachments['#attached']['library'][] = 'yptHEME/covid19';
            } elseif (substr($nodeBundle, 0, 7) === "yp_blog") {
                $attachments['#attached']['library'][] = 'yptHEME/yp-blog';
            } elseif ($nodeBundle === 'page') {
                $nid = $node->id();
                if (!empty($nid)) {
                    $alias = \Drupal::service('path.alias.manager')->getAliasByPath('/node/' . $nid);
                    $alias = ltrim($alias, '/');
                    $explodeAlias = explode('/', $alias);
                    switch (current($explodeAlias)) {
                        case 'sourcing':
                            $attachments['#attached']['library'][] = 'yptHEME/sourcing';
                            break;
                        case 'campaign':
                            $attachments['#attached']['library'][] = 'yptHEME/campaign';
                            break;
                        case 'advertising-consultants':
                            $attachments['#attached']['library'][] = 'yptHEME/advertising-consultants';
                            break;
                        default:
                            break;
                    }
                }
            }
        }
        } elseif (strpos($route, 'view.yp_blog') === 0 || $pathSlot[1] === 'blogterm') {
            $attachments['#attached']['library'][] = 'yptHEME/yp-blog';
            $attachments['#attached']['library'][] = 'typ_contrib/swiper4';
        } elseif
        (strpos($route, 'view.typ_store') === 0) {
            $attachments['#attached']['library'][] = 'typ_contrib/swiper4';
            $attachments['#attached']['library'][] = 'yptHEME/app-store';
        } elseif
        (strpos($route, 'typ_profile.digi_name_card') === 0) {
            $attachments['#attached']['library'][] = 'yptHEME/diginame';
            if (strpos($route, 'typ_profile.digi_name_card_google_map') === 0) {
                foreach ($attachments['#attached']['library'] as $key => $val) {
                    if ($val === 'yptHEME/global-styling') {
                        unset($attachments['#attached']['library'][$key]);
                    }
                }
            }
        }
    }
}
```

โดยส่วนที่ขีดเส้นใต้จะเป็นส่วนที่เพิ่มมา โดยใน node จะเช็คจากชื่อสำหรับอ้างอิง หรือตัวแปร \$nodebudle ที่ตั้งไว้เป็น yp_blog (ตามด้วย _ชื่อหมวดหมู่) และในหน้า views ปกติจะเช็คด้วย strpos(\$route, 'view.yp_blog') === 0 และ \$pathSlot[1] === 'blogterm' โดยทั้งสองเพื่อให้ดึงไฟล์ css และ js

ชื่อสำหรับอ้างอิง *

yp_blog_movie

ระบบถูกใจ

page.php

ใน function preprocessVariables เพิ่มข้างล่างสุด

```
if($path[1] === 'blog'){  
    $this->getBlog($variables);  
}
```

และเขียน function getBlog เพิ่ม

```
protected function getBlog(&$variables)  
{  
    if ($variables['node']) {  
        try {  
            //View count  
            $requestPath = \Drupal::request()->getRequestUri();  
            $query['PathURL'] = $requestPath;  
            $ses = TYPESearchService::query('db.get.ypblog.total.view', $query)->getJson();  
            $variables['var']['blog']['view_count'] = 0;  
            if (!empty($ses['hits']['total'])) {  
                $variables['var']['blog']['view_count'] = $ses['hits']['total'];  
            }  
            //Taxonomy  
            $tid = $variables['node']->get('field_tags')->getValue()[0]['target_id'];  
            if(!empty($tid)){  
                try {  
                    $term = Term::load((int)$tid);  
                    $aliasManager = \Drupal::service('path.alias_manager');  
                    //Navigation  
                    $variables['var']['navigate']['home'] = $variables['base_root'];  
                    $variables['var']['navigate']['blog'] = '/blog/';  
                    $variables['var']['navigate']['title_blog'] = trim($term->getName());  
                    $node = \Drupal::routeMatch()->getParameter('node');  
                    $nodeBundle = $node->bundle();  
                    $alias = $aliasManager->getAliasByPath('/taxonomy/term/' . $tid);  
                    $variables['var']['blog']['name'] = $term->getName();  
                    $variables['var']['blog']['url'] = $alias;  
                }catch (\Exception $e){}  
            }  
            //Node  
            $nid = $variables['node']->id();  
            $liked = TYPAPILikeNodeBlockService::getCountLikedNode((int)$nid);  
            $variables['var']['blog']['liked'] = number_format($liked);  
            $suuid = TYPAPILikeNodeBlockService::getUUUID();  
            $variables['var']['blog']['is_liked'] = TYPAPILikeNodeBlockService::isLiked($nid, $suuid);  
        } catch (\Exception $e){  
        }  
    }  
}
```

และสร้าง modules มา 2 ไฟล์ Controller และ Service ใน TYP_API

TYPAPILikeNodeBlockController.php สำหรับถูกใจและยกเลิกถูกใจ สิ่งที่ดึงมาและโครงสร้างไฟล์

```
<?php
```

```
namespace Drupal\typ_api\Controller;

use Drupal\Core\Controller\ControllerBase;
use Drupal\typ_api\Service\TYPAPILikeNodeBlockService;
use Symfony\Component\HttpFoundation\JsonResponse;

class TYPAPILikeNodeBlockController extends ControllerBase
{
    //Code
}
```

function สำหรับ ถูกใจ

```
public function likeNodeBlock(): ?JsonResponse
{
    $nid = \Drupal::request()->get('nid');
    $x_csrf_token = \Drupal::request()->headers->get('x-csrf-token');
    $count = 0;
    if (!empty($x_csrf_token) && !empty($nid)) {
        $data = [];
        $uuid = TYPAPILikeNodeBlockService::getUUID();
        $data = [
            'nid' => (int)$nid,
            'uuid' => $uuid
        ];
        TYPAPILikeNodeBlockService::insert($data);
        $count = TYPAPILikeNodeBlockService::getCountLikedNode((int)$nid);
    } else {
        return new JsonResponse(['missing some required'], 422);
    }
    return new JsonResponse(['liked' => $count]);
}
```

function สำหรับลบข้อมูลการถูกใจ

```
public function dislikeNodeBlock(): ?JsonResponse
{
    $nid = \Drupal::request()->get('nid');
    $x_csrf_token = \Drupal::request()->headers->get('x-csrf-token');
    $count = 0;
    if (!empty($x_csrf_token) && !empty($nid)) {
        $data = [];
        $uuid = TYPAPILikeNodeBlockService::getUUID();
        $data = [
            'nid' => (int)$nid,
            'uuid' => $uuid
        ];
        TYPAPILikeNodeBlockService::delete($data);
        $count = TYPAPILikeNodeBlockService::getCountLikedNode((int)$nid);
    } else {
        return new JsonResponse(['missing some required'], 422);
    }
    return new JsonResponse(['liked' => $count]);
}
```

TYPAPILikeNodeBlockService.php

การดึงข้อมูล

```
public static function getCollection($collection = 'y_node_liked')
{
    return TYPMongoDB::getCollection($collection);
}
```

การ Insert ข้อมูล

```
public static function insert(array $data = []){
    if(!empty($data)){
        TYPDatabaseHelperService::helper(TYPAPILikeNodeBlockService::getCollection())->insert($data);
    }
    return true;
}
```

การ Delete ข้อมูล

```
public static function delete(array $data = []){
    if(!empty($data)){
        TYPDatabaseHelperService::helper(TYPAPILikeNodeBlockService::getCollection())->delete($data);
    }
    return true;
}
```

เช็คจำนวนไลค์

```
public static function getCountLikedNode(int $nid = 0) : ? int {
    $count = 0;
    if(!empty($nid)){
        $criteria = ['nid' => $nid];
        $cursor = self::getCollection()->find($criteria);
        $count = $cursor->count();
    }
    return $count;
}
```

เช็คว่าได้ไลค์อยู่หรือไม่

```
public static function isLiked(int $nid = 0, string $uuid = null) : ? bool
{
    $liked = true;
    if(!empty($nid) && !empty($uuid)){
        $criteria = [
            'nid' => $nid,
            'uuid' => $uuid
        ];
        $cursor = self::getCollection()->find($criteria);
        if($cursor->count() > 0){
            $liked = false;
        }
    }
    return $liked;
}
```

สร้าง BUID สำหรับการเพิ่มและลบข้อมูลของคนไม่ใช่ User

```
public static function generateUUID(){
    $uuid = \Drupal::service('uuid')->generate();
    self::setCookie($uuid);
    return $uuid;
}
```

setCookie

```
public static function setCookie(string $uuid = null){
    $cookie = new Cookie('_node_uuid', $uuid, time()+31556926, '/', self::getCookieDomain(), false, false);
    $response = new Response();
    $response->headers->setCookie($cookie);
    return $response->sendHeaders();
}
```

ดึงค่า UID

```
public static function getUID() : ? string
{
    $uuid = null;
    $uid = \Drupal::currentUser()->id();
    if(!empty($uid)){
        $account = User::load($uid);
        $uuid = $account->uuid();
    }else{
        $uuid = \Drupal::request()->cookies->get('_node_uuid');
    }
    if(empty($uuid)){
        $uuid = self::generateUUID();
    }
    return $uuid;
}
```

ดึงค่า Cookie

```
public static function getCookieDomain(){
    $host = \Drupal::request()->getHost();
    if(strpos($host, '.dev') !== false || strpos($host, '.tmc') !== false){
        return $host;
    }else{
        return 'yellowpages.co.th';
    }
}
```