

การจัดทำ Explicit Knowledge จากงานประจำ

ชื่อเรื่องการสร้างรูปภาพ BIB และ Certificate ด้วย ImageManager สำหรับ Laravel.....
ชื่อผู้เสนอนายพันธกานต์ มั่นจันทร์..... ตำแหน่งนักวิชาการคอมพิวเตอร์.....
สังกัด/ฝ่ายฝ่ายพัฒนาระบบ.....

๑. ภาระงานที่รับผิดชอบ

บริการและพัฒนาระบบสารสนเทศให้กับหน่วยงานภายในและภายนอกมหาวิทยาลัยบูรพา เช่น ระบบฐานข้อมูลงานวิจัยออนไลน์ มหาวิทยาลัยบูรพา, ระบบฐานข้อมูลงานวิจัยออนไลน์ มหาวิทยาลัยราชภัฏราชนครินทร์, ระบบสารสนเทศโครงการบริการวิชาการ มหาวิทยาลัยบูรพา, ระบบประเมินผลการปฏิบัติงาน มหาวิทยาลัยบูรพา, ระบบแจ้งบำรุงรักษาสำนักคอมพิวเตอร์มหาวิทยาลัยบูรพา, ลงทะเบียนงานวิจัยประเภท 8 กรกฎ, ระบบลงทะเบียนงานวิจัยวันสถาปนาวิทยาเขตสระแก้ว 9 กันยายน ฯลฯ

๒. การเรียนรู้ ความคาดหวัง ของ "ลูกค้า/ผู้ที่เกี่ยวข้องกับกระบวนการที่นำเสนอ"

ลูกค้าคือ ส่วนงานต่างๆ ภายในมหาวิทยาลัยบูรพา รวมถึงวิทยาเขตที่ต้องการใช้งานระบบลงทะเบียนวิจัย Virtual Run

๓. วัตถุประสงค์ (ให้บอกวัตถุประสงค์ขององค์ความรู้ที่นำเสนอ)

๓.๑ เพื่อให้ความรู้กับบุคลากรในฝ่ายให้สามารถนำองค์ความรู้ไปประยุกต์ใช้กับงานที่ได้รับมอบหมายได้

๔. บทสรุปองค์ความรู้ (สรุปเนื้อหาองค์ความรู้ที่สำคัญ และจำเป็น บอกแนวทางการดำเนินการ วิธีการ หรือวิธีการปฏิบัติเพื่อให้บรรลุองค์ความรู้ที่ต้องการ)

๔.๑. สิ่งที่เป็นและต้องเตรียม

๔.๑.๑. ตรวจสอบ Environment ของ Server ให้รองรับการใช้งาน PHP Version ≥ 5.4 , Fileinfo Extension, GD Library (≥ 2.0) หรือ Imagick PHP extension ($\geq 6.5.7$) และติดตั้ง Composer

๔.๑.๒. ผสานงานกับลูกค้าเพื่อขอข้อมูล ดังนี้

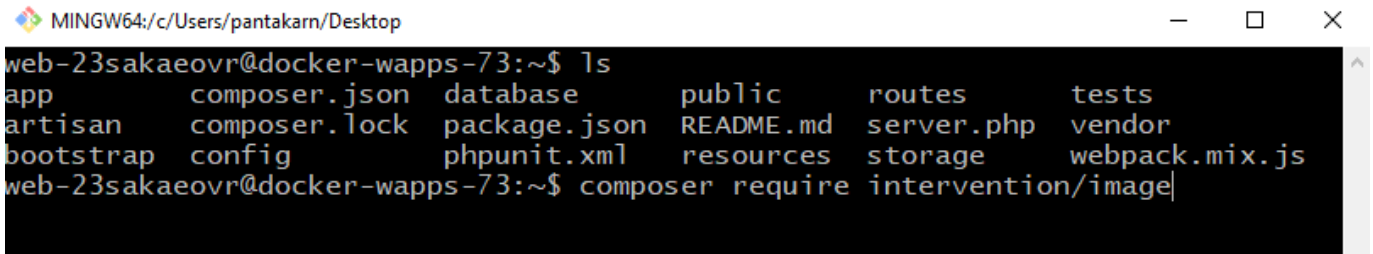
1. รูปภาพ 2 รูปประกอบด้วย

- รูป BIB ขนาด 1 A4 หรือ 2480 x 3508 Pixel (.JPG)
- รูป Certificate ขนาด 1 A4 หรือ 2480 x 3508 Pixel (.JPG)

๔.๒. ติดตั้งโปรแกรมและ config

๔.๒.๑. นำโครงโปรแกรมต้นแบบ (Laravel Framework) มาติดตั้งที่เครื่อง Server ทดสอบ

๔.๒.๒. เปิด terminal ssh ไปยัง path ที่อยู่ของโปรแกรมแล้วพิมพ์คำสั่ง composer require intervention/image



```
MINGW64:/c/Users/pantakarn/Desktop
web-23sakaevr@docker-wapps-73:~$ ls
app          composer.json  database      public        routes        tests
artisan     composer.lock  package.json  README.md    server.php    vendor
bootstrap   config         phpunit.xml  resources    storage       webpack.mix.js
web-23sakaevr@docker-wapps-73:~$ composer require intervention/image
```

๔.๒.๒. ไปที่ config/app.php แล้วเพิ่ม Intervention\Image\ImageServiceProvider::class ที่ 'providers' => [...] และเพิ่ม 'Image' => Intervention\Image\Facades\Image::class ที่ 'aliases' => [...]

```
'providers' => [
    Intervention\Image\ImageServiceProvider::class
],

'aliases' => [
    'Image' => Intervention\Image\Facades\Image::class
]
```

๔.๓. การเรียกใช้งาน ImageManager เพื่อสร้างรูป BIB

๔.๓.๑. ไปที่ app/Http/Controllers แล้วสร้างไฟล์ Controllers ขึ้นมา

๔.๓.๒. เรียกใช้ Library โดยการ use Intervention\Image\ImageManager

๔.๓.๓. สร้างเลข BIB โดยที่ Controllers ให้เรียกใช้ \$manager = new ImageManager(array('driver' => 'imagick')); จะเป็นการเรียกใช้งาน Library และกำหนดค่าเริ่มต้น driver เป็น imagick จากนั้นโหลดไฟล์ Image Template โดยใช้ \$image = \$manager->make('images/backend/bib.jpg'); ซึ่งในพารามิเตอร์ make จะต้องกำหนด path image ให้ถูกต้อง ต่อมาเพิ่มข้อความใน BIB โดยใช้ \$image->text('เลข BIB', 'ตำแหน่งแกน x', 'ตำแหน่งแกน y', 'กำหนด font') การกำหนด font นั้นมีให้เลือกปรับหลายอย่างเช่น

- กำหนดชื่อ font จะใช้ \$font->file(public_path('font/THSarabun.ttf')); โดยจะต้องนำ font มาใส่ไว้ที่ path ที่เรียกใช้งานด้วย
- กำหนดขนาด font จะใช้ \$font->size(350);
- กำหนดสี font จะใช้ \$font->color('#000000');
- กำหนดตำแหน่งตัวอักษรจะใช้ \$font->align('center');

๔.๓.๔. บันทึกรูปภาพลง server จะใช้ \$image-

>save(Config::get('constants.path.storage_image').(date('Y')+543).'/'.\$filename); ให้ใส่ path image และชื่อไฟล์ลงไป

```
// create an image manager instance with favored driver
$imageManager = new ImageManager(array('driver' => 'imagick'));

// to finally create image instances
$image = $imageManager->make( 'images/backend/bib.jpg' );

// use callback to define details
$image->text($row->REG_BIBNAME, 1750, 1700, function($font) {
    $font->file( public_path('font/THSarabun.ttf') );
    $font->size(350);
    $font->color('#000000');
    $font->align('center');
});
$image->text($row->REG_BIB, 1750, 1400, function($font) {
    $font->file( public_path('font/THSarabun.ttf') );
    $font->size(1000);
    $font->color('#000000');
    $font->align('center');
});

// save the file in png format
$image->save(Config::get('constants.path.storage_image').(date('Y')+543).'/'.$filename);
```

๔.๔. การเรียกใช้งาน ImageManager เพื่อสร้างรูป Certificate

๔.๔.๑ ให้ทำตามขั้นตอนทั้งหมดจากข้อ ๔.๓ แต่ให้เปลี่ยนรูป Image Template ใหม่ให้เป็นรูป Certificate

\$image = \$imageManager->make('images/backend/cert.jpg'); จากนั้นปรับตำแหน่งกับกำหนด font อื่นๆตามต้องการและบันทึกรูปภาพลง server

```
// create an image manager instance with favored driver
$imageManager = new ImageManager(array('driver' => 'imagick'));

// to finally create image instances
$image = $imageManager->make( 'images/backend/cert.jpg' );

// use callback to define details
$image->text($row->REG_NAME, 1750, 1450, function($font) {
    $font->file( public_path('font/THSarabun.ttf') );
    $font->size(450);
    $font->color('#000000');
    $font->align('center');
});

// save the file in png format
$image->save(Config::get('constants.path.storage_cert').(date('Y')+543).'/'.$filename);
```

๔.๕. ผลลัพธ์ที่ได้



๕. ประโยชน์ที่ได้รับ (บอกประโยชน์ที่จะได้รับจากองค์ความรู้ที่นำเสนอทั้งภายในและภายนอกส่วนงาน)

บุคลากรในฝ่ายสามารถนำองค์ความรู้ไปใช้ในการทำงานของตนเองได้และส่วนงานที่ขอใช้งานระบบนำรูปภาพไปใช้งาน
ได้จริง