

สรุปผลการอบรมเชิงปฏิบัติการ

เรื่อง “การเขียนชุดคำสั่ง API แบบ Web Service เชื่อมโยงฐานข้อมูลบนเว็บไซต์ (ขั้นสูง)”

ประจำปีงบประมาณ พ.ศ. ๒๕๖๕ ระหว่างวันที่ ๑๓ - ๑๗ ธันวาคม ๒๕๖๔

ณ โรงแรมริเวอร์ตัน อัมพวา จ.สมุทรสงคราม

นายอนุกุลกิจ พุกาธร รองผู้อำนวยการกองแผนงาน เป็นประธานเปิดการอบรมเชิงปฏิบัติการ “การเขียนชุดคำสั่ง API แบบ Web Service เชื่อมโยงฐานข้อมูลบนเว็บไซต์ (ขั้นสูง)” ประจำปีงบประมาณ พ.ศ. ๒๕๖๕ โดยวัตถุประสงค์ของการอบรมฯ ดังกล่าว เพื่อพัฒนาทักษะดิจิทัลด้านการเขียนชุดคำสั่งช่องทางการเชื่อมต่อเพื่อแลกเปลี่ยนข้อมูลระหว่างซอฟต์แวร์ (Application Programming Interface; API) เชื่อมโยงฐานข้อมูลระหว่างหน่วยงานในระดับสูงแก่นักพัฒนาข้อมูลของกรมอนามัย ตอบสนองนโยบายรัฐบาลในการปรับเปลี่ยนองค์กรสู่ การเป็นรัฐบาลดิจิทัล และส่งเสริมภาพลักษณ์ที่ทันสมัยของกรมอนามัยสู่ภาคเอกชนและภาคประชาชนโดยมีวิทยากรจากภาคเอกชนเป็นผู้บรรยายให้ความรู้แก่เจ้าหน้าที่กรมอนามัย คือความรู้พื้นฐานการเขียน API แบบ Web Service โดยประกอบด้วย การทำงานของ Web Browser ภาษาต่างๆ ที่ใช้สำหรับการเขียนเว็บไซต์ รวมทั้งเครื่องมือและการเขียนชุดคำสั่งสำหรับการเขียน API ด้วย PHP & MySQL โดยในการอบรมฯ ดังกล่าวนั้น ได้มีการฝึกปฏิบัติการเขียนชุดคำสั่ง API แบบ Web Service โดยใช้เครื่องมือ Laravel Framework เพื่ออำนวยความสะดวกในการเขียนชุดคำสั่ง API โดยได้เรียนรู้การทำงานของเครื่องมือ Laravel Framework Version ๗ รวมถึงการนำเครื่องมือดังกล่าวมาใช้ในการเขียนชุดคำสั่งในการรับ-ส่งแลกเปลี่ยนข้อมูลผ่าน RESR API ในรูปแบบข้อมูล JSON เพื่อนำไปประยุกต์ใช้ในการออกแบบหน้ารายงานข้อมูลผ่าน Web Service ต่างๆ ได้

ผู้เข้าร่วมการอบรมฯ จำนวน ๑๖ คน ประกอบด้วย

๑) เจ้าหน้าที่ผู้ปฏิบัติงานบริหารจัดการข้อมูลของหน่วยงานภายในกรมอนามัย จำนวน ๖ หน่วยงาน ได้แก่ สำนักอนามัยสิ่งแวดล้อม สำนักส่งเสริมสุขภาพ สำนักอนามัยผู้สูงอายุ กองประเมินผลกระทบ ต่อสุขภาพ กองส่งเสริมความรู้และสื่อสารสุขภาพ และกองแผนงาน จำนวน ๑๑ คน

๒) คณะทำงานจัดอบรมฯ จำนวน ๓ คน

๓) วิทยากร จำนวน ๒ คน

สรุปเนื้อหาการอบรมฯ

๑. การใช้งานโปรแกรม Laravel Framework Version ๗

๑.๑ Migration & Seeder Data

- การสร้างตารางในฐานข้อมูลต่างๆ โดยผ่านคำสั่ง command ของตัว laravel และสามารถนำไปใช้เป็นตัวต้นแบบของตารางในฐานข้อมูลกรณีมีการนำไปแจกไปให้นักพัฒนาหลายๆคน พัฒนาร่วมกัน

- การสร้าง Mockup data ต่างๆ โดยการ insert ลง ตารางในฐานข้อมูล สามารถนำไปใช้เป็นต้นแบบของตารางในฐานข้อมูลกรณีมีการนำไปโปรเจกต์ให้นักพัฒนาหลายๆคนพัฒนาร่วมกัน

๑.๒ Routing & Middleware

- การกำหนดเส้นทางต่างๆ การสร้าง Url ต่างๆ ในรูปแบบ get, post, put, delete เพื่อเข้าถึงเว็บไซต์ หรือ โปรเจกต์ ของเรา
- การกรอง Request ต่างๆ ก่อนที่จะมาถึงส่วนของ route ที่อยู่ในโปรเจกต์ของเรา คล้ายๆ ตัวป้องกันด่านแรกของระบบ

๑.๓ Controller & Model

- ส่วนของการจัดการ การทำงานของระบบต่างๆ เป็นตัวเชื่อมระหว่าง Model และ View มีหน้าที่คอยจัดการข้อมูลต่างๆ เพื่อนำไปใช้กับส่วนอื่นๆ
- ส่วนของการเชื่อมโยงฐานข้อมูล ไม่ว่าจะ เป็นในรูปแบบ select, insert, update, delete โดยจะทำหน้าที่นำเข้า หรือ ส่งออกข้อมูลไปยัง Controller เพื่อที่ส่วนของ Controller จะได้นำข้อมูลไปทำงานต่อ

๑.๔ API Authentication

- ขั้นตอนการกรองผู้ใช้งาน API การอนุญาตการเข้าถึงข้อมูลต่างๆ คล้ายๆระบบ login ที่มีในเว็บไซต์ทั่วไป มีการใช้ token เพื่อเป็นการระบุตัวตนเพื่อขอการเข้าถึงชุดข้อมูล

๑.๕ Validation & Session

- การกรองข้อมูลต่างๆที่มีการส่งเข้ามา รูปแบบข้อมูลต้องถูกต้องตามที่กำหนด เช่น ชื่อ ต้องประกอบด้วยตัวอักษรพิมพ์ใหญ่ และ พิมพ์เล็ก , อีเมล ต้องประกอบด้วยรูปแบบ อีเมล ที่ถูกต้อง เป็นต้น
- การเก็บข้อมูลของผู้ที่ต้องการเข้าถึงส่วนต่างๆของข้อมูลในรูปแบบ file, cache, memory โดยทางฝั่ง Server จะมีหน้าที่สร้างให้ เช่น session login ต่างๆ ถ้าเมื่อ session หมดอายุ หรือ หายไปแล้วนั้น จะทำให้ต้องทำการ login หรือ สร้าง session ใหม่ เป็นต้น

๒. ขั้นตอนการสร้าง API

๒.๑ วิเคราะห์ข้อมูลที่ต้องการทำ API

- ข้อมูลเกี่ยวข้องกับอะไร
- ตำแหน่งที่ข้อมูลนี้จัดเก็บ ใช้ฐานข้อมูลอะไร ขอ user password ในการเข้าถึง
- อยู่ที่ตารางไหน มีฟิลด์ไหนเกี่ยวข้องบ้าง
- มีเงื่อนไขอะไรในการดึงข้อมูล
- อยากรายงานนี้ ค้นหาอะไรได้บ้าง เช่น filter ตาม type ข้อมูลต่าง ๆ
- ข้อมูลที่ได้จาก api สามารถทำ caching ได้หรือไม่
- อยากรู้แสดงข้อมูลในแต่ละครั้งที่เรียก ต่ำสุด สูงสุด เช่นทำ pagination
- ต้องการให้ข้อมูลแสดงผลด้วยรูปแบบไหน เช่น json, xml

๒.๒ เริ่มเขียนโค้ดสำหรับ API ด้วย Laravel ๗.๐, MySQL

- เริ่มจากการสร้าง Controller คำสั่ง

```
php artisan make:controller FoodhandlerController
```

๒.๓ กำหนด Route ในการเข้าถึง

- เปิดไฟล์ routes/api.php เพิ่มคำสั่ง

```
Route::get('/foodhadlerPersonTypeTwo', 'FoodhandlerController@personTypeTwo');
```

๒.๔ เพิ่ม Function สำหรับทำ Api

- เปิดไฟล์ app/Http/Controllers/FoodhandlerController.php
- เพิ่มคำสั่ง

```
Public function personTypeTwo(){
    return response()->json(['key'=> 'testing']);
}
```

๒.๕ คำสั่งที่ใช้ในการติดต่อฐานข้อมูล MySQL

- เปิดไฟล์ app/Http/Controllers/FoodhandlerController.php
- เพิ่มคำสั่ง

```
use Illuminate\Support\Facades\DB;
```

๒.๖ กำหนด .env สำหรับเข้าฐานข้อมูล

- เปิดไฟล์ .env
- เพิ่มคำสั่ง:

- ▶ DB_HOST_FOODHANDLER=xx.xxx.x.xxx
- ▶ DB_PORT_FOODHANDLER=๓๓๐๖
- ▶ DB_DATABASE_FOODHANDLER=foodhandler
- ▶ DB_USERNAME_FOODHANDLER=apixxxx
- ▶ DB_PASSWORD_FOODHANDLER=xxxxxxxxxx

๒.๗ เพิ่มค่า config สำหรับเข้าฐานข้อมูล เพื่ออ่านค่าที่กำหนดไว้

- เปิดไฟล์ config/database.php
- เพิ่มคำสั่ง:

Copy ตัวแปร mysql ทั้งหมด แล้วเปลี่ยนชื่อเป็น mysql_foodhandler
จากนั้นแก้ไขตัว

- ▶ DB_HOST เป็น DB_HOST_FOODHANDLER

- ▶ DB_PORT เป็น DB_PORT_FOODHANDLER
- ▶ DB_DATABASE เป็น DB_DATABASE_FOODHANDLER
- ▶ DB_USERNAME เป็น DB_USERNAME_XXXXXXXXXX
- ▶ DB_PASSWORD เป็น DB_PASSWORD_XXXXXXXXXX

๒.๘ ทดสอบ Route กับข้อมูล Api

- ทดสอบตาม Route ที่ได้ตั้งชื่อไว้

localhost:๘๐๐๐/api/foodhadlerPersonTypeTwo

๒.๙ ผลลัพธ์ที่ได้ รูปแบบ json

```
{
  "current_page": ๑,
  "data": [
    {
      "abtname": "เทศบาลตำบลกำแพง",
      "subdistrict": "กำแพง",
      "district": "ละงู",
      "province": "สตูล"
    }
  ],
  "first_page_url": "http://localhost:๘๐๐๐/api/foodhandlerOne?page=๑",
  "from": ๑,
  "last_page": ๒๒๗๕,
  "last_page_url": "http://localhost:๘๐๐๐/api/foodhandlerOne?page=๒๒๗๕",
  "next_page_url": "http://localhost:๘๐๐๐/api/foodhandlerOne?page=๒",
  "path": "http://localhost:๘๐๐๐/api/foodhandlerOne",
  "per_page": ๒๐,
  "prev_page_url": null,
  "to": ๒๐,
  "total": ๔๕๔๙๒
}
```

สรุปประเมินผลการอบรมฯ

๑. การประเมินความรู้ของผู้เข้ารับการอบรมฯ

ผลการประเมินความรู้ความเข้าใจและการนำความรู้ไปใช้ของผู้เข้าร่วมการอบรม พบว่า

- ความรู้ความเข้าใจก่อนการอบรม

ระดับปานกลาง ร้อยละ ๕๖.๔

- ความรู้ความเข้าใจหลังการอบรม

ระดับมากที่สุด ร้อยละ ๘๗.๒

- สามารถนำความรู้ที่ได้จากการอบรมมาประยุกต์กับการใช้ในการปฏิบัติงาน หรือเผยแพร่ ให้คำปรึกษาแก่เพื่อนร่วมงาน

ระดับมากที่สุด ร้อยละ ๘๕.๔

๒. การประเมินความพึงพอใจของผู้เข้ารับการอบรมฯ

- ความพึงพอใจต่อวิทยากร

ระดับมากที่สุด ร้อยละ ๙๒.๘

- ความพึงพอใจต่อการจัดการอบรมภาพรวม

ระดับมากที่สุด ร้อยละ ๘๕.๘

๓. ข้อเสนอแนะของผู้เข้ารับการอบรมฯ

- ผู้เข้าร่วมอบรมเสนอให้มีการจัดอบรมเขียน Code รับ API ของหน่วยงานและให้มีการจัดการอบรมหลักสูตรนี้ซ้ำ (Refresh) เพื่อทบทวนความรู้แก่ผู้ผ่านอบรมสม่ำเสมอ และเปิดโอกาสให้แก่ผู้รับผิดชอบงานใหม่ได้เรียนรู้ API เพราะยังมีอีกหลายเรื่องที่ต้องเรียนรู้เพิ่มและนำความรู้ที่ได้จากการอบรมนำไปใช้ในการทำงานของหน่วยงานมากขึ้น

ประโยชน์ที่ได้รับจากการอบรมฯ

๑. ผู้เข้ารับการอบรมมีความรู้และทักษะในการเขียนชุดคำสั่ง API เชื่อมโยงฐานข้อมูล MYSQL ซึ่งเป็นฐานข้อมูลที่นิยมใช้กันมากที่สุด โดยจากการ Workshop ของการอบรมครั้งนี้ ทำให้กรมอนามัยได้ชุดคำสั่ง API มาใช้ประโยชน์จำนวน ๒ เรื่อง ได้แก่

๑) ข้อมูลทะเบียนผู้ประกอบการ <https://api.anamai.moph.go.th/api/foodhandlerOne>

๒) ข้อมูลทะเบียนผู้สัมผัสอาหาร <https://api.anamai.moph.go.th/api/foodhandlerTwo>

๒. กรมอนามัยมีชุดข้อมูลมาตรฐานแบบเปิด (Open data) ในรูปแบบ API สำหรับให้บริการข้อมูลภาครัฐที่ทันสมัย โดยจัดเก็บ API ไว้บนเว็บไซต์ Open API Zone ที่ URL : <https://api.anamai.moph.go.th> ซึ่งเป็นแหล่งจัดเก็บและรวบรวมข้อมูลเปิดของกรมอนามัยในรูปแบบ API

๓. กรมอนามัยดำเนินงานได้ตามเกณฑ์การประเมินส่วนราชการของสำนักงานคณะกรรมการพัฒนาระบบราชการ ในตัวชี้วัด เรื่อง “การพัฒนากระบวนการบัญชีข้อมูล (Data Catalog) เพื่อนำไปสู่การเปิดเผยข้อมูลภาครัฐ (Open Data)” ซึ่งเป็นหนึ่งในตัวชี้วัดความสำเร็จของอภิศักกรมอนามัย ประจำปีงบประมาณ พ.ศ. ๒๕๖๕

ภาพการอบรมเชิงปฏิบัติการ
เรื่อง “การเขียนชุดคำสั่ง API แบบ Web Service เชื่อมโยงฐานข้อมูลบนเว็บไซต์ (ขั้นสูง)”
ประจำปีงบประมาณ พ.ศ. ๒๕๖๕ ระหว่างวันที่ ๑๓ - ๑๗ ธันวาคม ๒๕๖๔
ณ โรงแรมริเวอร์ตัน อัมพวา จ.สมุทรสงคราม

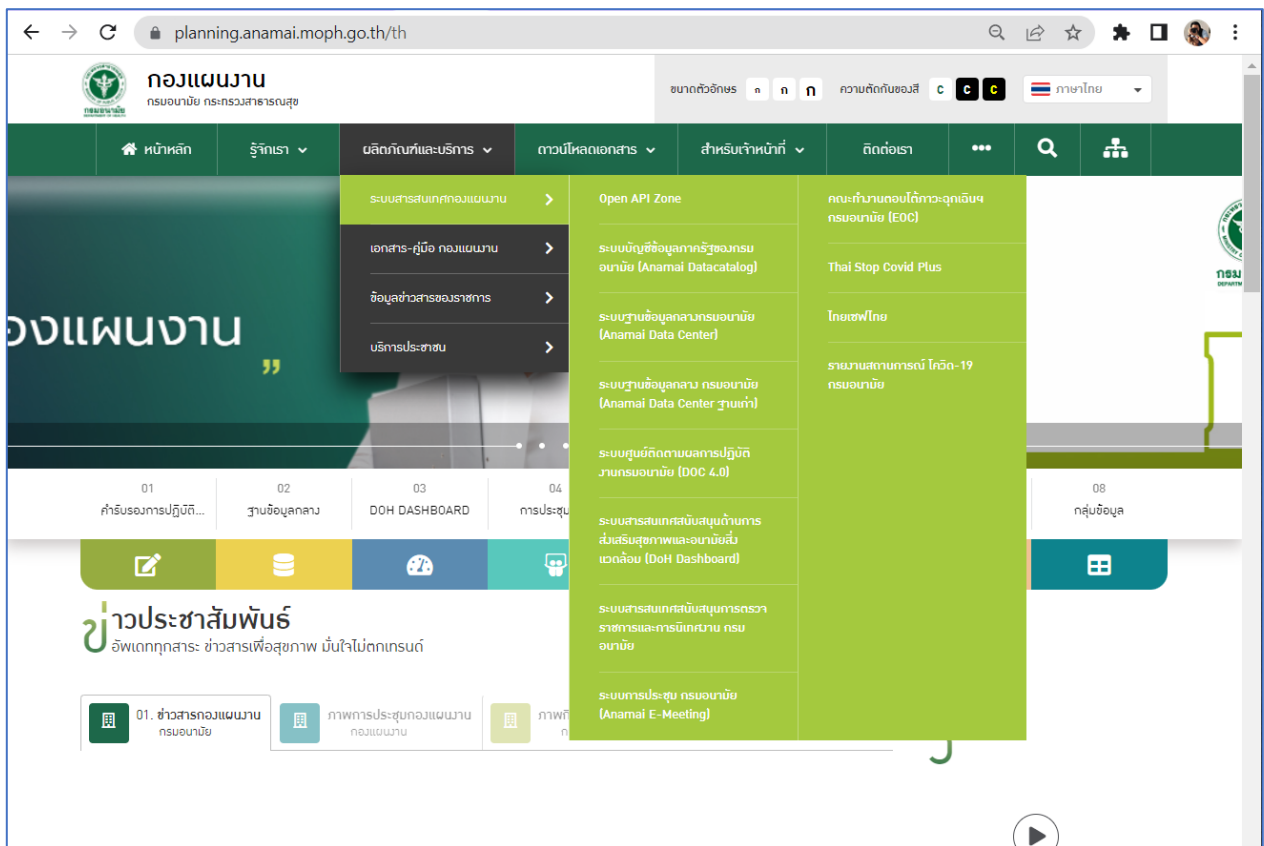
<https://www.facebook.com/media/set/?set=a.5189353664458779&type=3>



ช่องทางเข้า Open API Zone

(แหล่งรวบรวมข้อมูลเปิดของกรมอนามัย ในรูปแบบ API)

๑. เว็บไซต์กองแผนงาน <https://planning.anamai.moph.go.th/>
๒. เมนูระบบสารสนเทศกองแผนงาน
๓. Open API Zone (แหล่งรวบรวมข้อมูลเปิดของกรมอนามัย ในรูปแบบ API)
<https://api.anamai.moph.go.th/?cate=๕>



Open API Zone (แหล่งรวบรวมข้อมูลเปิดของกรมอนามัย ในรูปแบบ API)

<https://api.anamai.moph.go.th/?cate=๕>

ลงทะเบียน
เข้าสู่ระบบ

Search

Open API Zone

แหล่งรวบรวมข้อมูลเปิดของกรมอนามัย ในรูปแบบ API

สตรีและเด็กปฐมวัย
วัยเรียนวัยรุ่น
วัยทำงาน
ผู้สูงอายุ
อนามัยสิ่งแวดล้อม
ความรอบรู้สุขภาพ
Anamai Poll

อนามัยสิ่งแวดล้อม

ขยะติดเชื้อ

ลำดับ	หัวข้อ	Content-Type	API URL	เอกสารอธิบายข้อมูล
		ไม่พบข้อมูล		

สุขาภิบาลน้ำ

ลำดับ	หัวข้อ	Content-Type	API URL	เอกสารอธิบายข้อมูล
1.	ข้อมูลจำนวนตัวอย่างน้ำบริโภคประเภทต่างๆ ได้มาตรฐานตามเกณฑ์คุณภาพน้ำบริโภคกรมอนามัย	JSON	https://dashboard.anamai.moph.go.th/envwaterquality/envwaterquality/data?year=2020	

เหตุรำคาญ

ลำดับ	หัวข้อ	Content-Type	API URL	เอกสารอธิบายข้อมูล
		ไม่พบข้อมูล		

สิ่งปฏิกูล

ลำดับ	หัวข้อ	Content-Type	API URL	เอกสารอธิบายข้อมูล
		ไม่พบข้อมูล		

สุขาภิบาลอาหาร

ลำดับ	หัวข้อ	Content-Type	API URL	เอกสารอธิบายข้อมูล
1.	ข้อมูลทะเบียนผู้ประกอบการ	JSON	https://api.anamai.moph.go.th/api/foodhandlerOne	
2.	ข้อมูลทะเบียนผู้สัมผัสอาหาร	JSON	https://api.anamai.moph.go.th/api/foodhandlerTwo	

กลุ่มพัฒนาระบบข้อมูล กองแผนงาน
 © 2021 Anamai, Inc Email : data.e@anamai.mail.go.th
 โทรศัพท์ 0-2590-4648, 0-2590-4301

Code สร้าง API

```

<?php
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\DB;
class ApiFoodhandlerController extends Controller
{
    public function fsregisterDetailOne(Request $req){
        $perPage = $req->has('perPage') ? $req->perPage : ๒๐ ;
        $rs = DB::connection('mysql_foodhandler')
            ->table('fsregister_detail')
            ->select([
                'abtname',
                'subdistrict',
                'district',
                'province'])
            ->where('person_type',๑)
            ->paginate($perPage);
        return response()->json($rs);}

    public function fsregisterDetailTwo(Request $req){
        $perPage = $req->has('perPage') ? $req->perPage : ๒๐ ;
        $rs = DB::connection('mysql_foodhandler')
            ->table('fsregister_detail')
            ->select([
                'abtname',
                'subdistrict',
                'district',
                'province'])
            ->where('person_type',๒)
            ->paginate($perPage);
        return response()->json($rs);
    }
}

```