

คู่มือการใช้งาน GDCatalog API Version 1.0

สำนักงานสถิติแห่งชาติ
ศูนย์สารสนเทศยุทธศาสตร์ภาครัฐ



สารบัญ

	หน้า
1. คำนำ.....	1
2. คำศัพท์ที่สำคัญ	1
3. องค์ประกอบทะเบียนบัญชีข้อมูล	2
4. คำอธิบายชุดข้อมูล	2
5. การแจ้งขอลงทะเบียนใช้งาน GDCatalog API.....	6
6. รูปแบบบริการ GDCatalog API.....	7
7. ขั้นตอนโดยย่อการใช้งานด้วยชุดคำสั่ง.....	7
8. เครื่องมือที่ใช้ในการทดสอบการเชื่อมต่อ API	7
9. CKAN API.....	7
10. Directory API	11
11. ตัวอย่างการใช้ชุดคำสั่งภาษา PHP.....	23
12. ตัวอย่างการใช้ชุดคำสั่งภาษา Python.....	27

1. คำนำ

บริการระบบบัญชีข้อมูลภาครัฐ นอกจากเข้าถึงใช้งานโดยตรงกับผ่านเว็บท่าที่ <https://gdcatalog.go.th> และนามานุกรมบัญชีข้อมูลภาครัฐที่ <https://directory.gdcatalog.go.th> แล้วเพื่ออำนวยความสะดวกในการค้นหาชุดข้อมูลที่หน่วยงานรัฐจัดเก็บ ถือครอง และได้ลงทะเบียนชุดข้อมูลกับระบบบัญชีข้อมูลภาครัฐแล้วนั้น

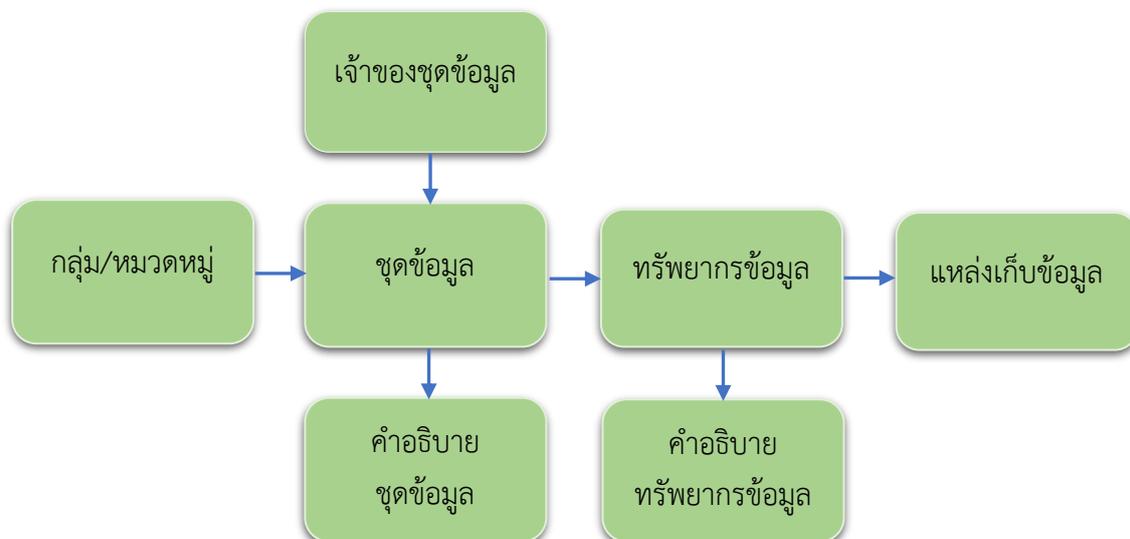
เพื่อเป็นการอำนวยความสะดวกการใช้ทะเบียนบัญชีข้อมูลภาครัฐให้เกิดประโยชน์สูงสุด สำนักงานสถิติแห่งชาติ จึงได้จัดทำบริการเชื่อมต่อด้วยช่องทางโปรแกรม หรือ API ให้กับนักพัฒนาระบบสารสนเทศของหน่วยงานที่มีความต้องการเรียกใช้ทะเบียนบัญชีข้อมูลแบบอัตโนมัติด้วยระบบคอมพิวเตอร์ เพื่อใช้งานแบบทันทีทันใดใน ๒ ลักษณะ คือ การเชื่อมต่อทางโปรแกรมแบบ CKAN API และ การเชื่อมต่อทางโปรแกรมแบบ Directory API ซึ่งจะรวมเรียกว่า GDCatalog API โดยทั้งสองลักษณะจะเป็นการเรียกใช้ทะเบียนชุดข้อมูลที่หน่วยงานภาครัฐได้ลงทะเบียนบัญชีแล้วแต่จะมีชุดฟังก์ชันการใช้งานที่แตกต่างกัน (name space)

โดยชุดฟังก์ชันการใช้งานแบบ CKAN API นั้นเป็นชุดฟังก์ชันสำหรับซอฟต์แวร์ Data Management ของผู้พัฒนาระบบ CKAN ซึ่งมีการพัฒนาอย่างต่อเนื่องเปลี่ยนแปลงไปตามรุ่นของการพัฒนาสำหรับ CKAN API ที่ใช้งานในระบบบัญชีข้อมูลภาครัฐเป็นรุ่น 2.9 ส่วนชุดฟังก์ชันการใช้งานแบบ Directory API นั้นเป็นชุดฟังก์ชันสำหรับการเรียกใช้งานเพื่องานนามานุกรมเป็นรุ่น 1.0

2. คำศัพท์ที่สำคัญ

- 2.1. Token หมายถึง รหัสชั่วคราว มีกำหนดเวลาหมดอายุสำหรับใช้ในการเชื่อมต่อระบบ เพื่อเรียกใช้งาน API
- 2.2. API-Key หมายถึง รหัสสำหรับใช้ในการเชื่อมต่อระบบโดยไม่มีกำหนดเวลาหมดอายุ เพื่อเรียกใช้งาน API
- 2.3. Parameters หมายถึง ตัวแปรที่ส่งไปยังฟังก์ชันของ API ตามข้อกำหนดของฟังก์ชันนั้นๆ เช่น full mandatory และ none
- 2.4. Client_id หมายถึง หมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้ ใช้ในการ Authorization เพื่อให้ได้รับ Token
- 2.5. Client_secret หมายถึง รหัสหมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้ ใช้ในการ Authorization เพื่อให้ได้รับ Token

3. องค์ประกอบทะเบียนบัญชีข้อมูล



รูปแสดง องค์ประกอบทะเบียนบัญชีข้อมูลภาครัฐ

4. คำอธิบายชุดข้อมูล

คำอธิบายข้อมูลส่วนหลัก (Mandatory Metadata) 14 รายการบังคับ

- 1 ประเภทข้อมูล (5 ทางเลือก)
 - ข้อมูลระเบียบ
 - ข้อมูลสถิติ
 - ข้อมูลภูมิสารสนเทศเชิงพื้นที่
 - ข้อมูลหลากหลายประเภท
 - ข้อมูลประเภทอื่น ๆ
- 2 ชื่อชุดข้อมูล
- 3 องค์กร
- 4 ชื่อผู้ติดต่อ
- 5 อีเมลผู้ติดต่อ
- 6 คำสำคัญ
- 7 รายละเอียด
- 8 วัตถุประสงค์ (14 ทางเลือก)
- 9.1 หน่วยความถี่ของการปรับปรุงข้อมูล (13 ทางเลือก)
- 9.2 ค่าความถี่ของการปรับปรุงข้อมูล
- 10 ขอบเขตเชิงภูมิศาสตร์หรือเชิงพื้นที่ (14 ทางเลือก)
- 11 แหล่งที่มา
- 12 รูปแบบการเก็บข้อมูล (16 ทางเลือก)
- 13 หมวดหมู่ข้อมูลตามธรรมาภิบาลข้อมูลภาครัฐ (4 ทางเลือก)
- 14 สัญญาอนุญาตให้ใช้ข้อมูล (7 ทางเลือก)

คำอธิบายข้อมูล ส่วนที่เป็นทางเลือก (Optional Metadata)

<div style="background-color: #f080f0; border-radius: 10px; padding: 5px; text-align: center; margin-bottom: 5px;">ข้อมูล ระเบียบ</div> <ul style="list-style-type: none"> <li style="background-color: #f080f0; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">15 เจ็อนไซในการเข้าถึงข้อมูล <li style="background-color: #f080f0; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">16 วันที่เริ่มต้นสร้าง <li style="background-color: #f080f0; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">17 วันที่ปรับปรุงข้อมูลล่าสุด <li style="background-color: #f080f0; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">18 URL <li style="background-color: #f080f0; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">19 ผู้สนับสนุนหรือผู้ร่วมดำเนินการ <small>(7 ทางเลือก)</small> <li style="background-color: #f080f0; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">20 หน่วยที่ย่อยที่สุดของการจัดเก็บข้อมูล <small>(13 ทางเลือก)</small> <li style="background-color: #f080f0; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">21 ภาษาที่ใช้ <small>(14 ทางเลือก)</small> <li style="background-color: #f080f0; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">22 ชุดข้อมูลที่มีคุณค่าสูง <li style="background-color: #f080f0; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">23 ข้อมูลอ้างอิง 	<div style="background-color: #ffff00; border-radius: 10px; padding: 5px; text-align: center; margin-bottom: 5px;">ข้อมูล หลากหลาย ประเภท</div>	<div style="background-color: #ffa500; border-radius: 10px; padding: 5px; text-align: center; margin-bottom: 5px;">ข้อมูล ประเภท อื่น ๆ</div>	<div style="background-color: #d8bfd8; border-radius: 10px; padding: 5px; text-align: center; margin-bottom: 5px;">ข้อมูลสถิติ</div> <ul style="list-style-type: none"> <li style="background-color: #d8bfd8; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">15 เจ็อนไซในการเข้าถึงข้อมูล <li style="background-color: #d8bfd8; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">16 ปีข้อมูลที่เริ่มต้นจัดทำ <li style="background-color: #d8bfd8; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">17 ปีข้อมูลล่าสุดที่เผยแพร่ <li style="background-color: #d8bfd8; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">18 วันที่กำหนดเผยแพร่ข้อมูล <li style="background-color: #d8bfd8; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">19 วันที่ปรับปรุงข้อมูลล่าสุด <li style="background-color: #d8bfd8; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">20 การจัดทำแยก <small>(14 ทางเลือก)</small> <li style="background-color: #d8bfd8; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">21 หน่วยวัด <li style="background-color: #d8bfd8; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">22 หน่วยตัวคูณ <small>(15 ทางเลือก)</small> <li style="background-color: #d8bfd8; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">23 วิธีการคำนวณ <li style="background-color: #d8bfd8; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">24 มาตรฐานการจัดทำข้อมูล <li style="background-color: #d8bfd8; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">25 URL <li style="background-color: #d8bfd8; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">26 ภาษาที่ใช้ <li style="background-color: #d8bfd8; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">27 สถิติทางการ 	<div style="background-color: #6495ed; border-radius: 10px; padding: 5px; text-align: center; margin-bottom: 5px;">ข้อมูลภูมิสารสนเทศเชิงพื้นที่</div> <ul style="list-style-type: none"> <li style="background-color: #6495ed; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">15 เจ็อนไซในการเข้าถึงข้อมูล <li style="background-color: #6495ed; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">16 ชุดข้อมูลภูมิศาสตร์ <small>(13 ทางเลือก)</small> <li style="background-color: #6495ed; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">17 มาตราส่วน <small>(6 ทางเลือก)</small> <li style="background-color: #6495ed; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">18.1 คำพ้องที่ครอบคลุมที่ด้านทิศตะวันตก <li style="background-color: #6495ed; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">18.2 คำพ้องที่ครอบคลุมที่ด้านทิศตะวันออก <li style="background-color: #6495ed; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">18.3 คำพ้องที่ครอบคลุมที่ด้านทิศเหนือ <li style="background-color: #6495ed; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">18.4 คำพ้องที่ครอบคลุมที่ด้านทิศใต้ <li style="background-color: #6495ed; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">19 ความถูกต้องของตำแหน่ง <li style="background-color: #6495ed; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">20 เวลาอ้างอิง <li style="background-color: #6495ed; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">21 วันที่ปรับปรุงข้อมูลล่าสุด <li style="background-color: #6495ed; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">22 วันที่กำหนดเผยแพร่ข้อมูล <li style="background-color: #6495ed; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">23 วันที่เผยแพร่ข้อมูล <li style="background-color: #6495ed; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">24 URL <li style="background-color: #6495ed; border-radius: 10px; padding: 2px 5px; margin-bottom: 2px;">25 ภาษาที่ใช้
---	--	--	---	---

ลำดับ	ชื่อทางเทคนิค (Field name)	ชื่อรายการภาษาอังกฤษ	ชื่อรายการ ภาษาไทย	Value
1	data_type	Data Type	ประเภทชุด ข้อมูล	ข้อมูลระเบียบ
				ข้อมูลสถิติ
				ข้อมูลหลากหลายประเภท
				ข้อมูลภูมิสารสนเทศเชิงพื้นที่
				ข้อมูลประเภทอื่นๆ
2	title	Dataset Name	ชื่อชุดข้อมูล	ผู้ใช้ระบุ
3	owner_org	Organization	องค์กร	ผู้ใช้เลือกตามองค์กรที่มีในระบบ
4	maintainer	Contact Person	ชื่อผู้ติดต่อ	ผู้ใช้ระบุ
5	maintainer_email	Contact Email	อีเมลผู้ติดต่อ	ผู้ใช้ระบุ
6	tag_string	Tags	คำสำคัญ	ผู้ใช้ระบุ
7	notes	Description	รายละเอียด	ผู้ใช้ระบุ
8	objective	Objective	วัตถุประสงค์	ยุทธศาสตร์ชาติ
				แผนพัฒนาเศรษฐกิจและสังคมแห่งชาติ
				แผนความมั่นคงแห่งชาติ
				แผนแม่บทภายใต้ยุทธศาสตร์ชาติ

ลำดับ	ชื่อทางเทคนิค (Field name)	ชื่อรายการภาษาอังกฤษ	ชื่อรายการ ภาษาไทย	Value
				แผนปฏิรูปประเทศ แผนระดับที่ 3 (ม.ค.ร. 4 ธ.ค. 2560) นโยบายรัฐบาล/ข้อสั่งการ นายกรัฐมนตรี มติคณะรัฐมนตรี เพื่อการให้บริการประชาชน กฎหมายที่เกี่ยวข้อง พันธกิจหน่วยงาน ดัชนี/ตัวชี้วัดระดับนานาชาติ ไม่ทราบ อื่นๆ
9.1	update_frequency_unit	Update Frequency Unit	หน่วยความถี่ของการปรับปรุงข้อมูล	ไม่ทราบ ปี ครึ่งปี ไตรมาส เดือน สัปดาห์ วัน วันทำการ ชั่วโมง นาที ตามเวลาจริง ไม่มีการปรับปรุงหลังจากการ จัดเก็บข้อมูล อื่นๆ
9.2	update_frequency_interval	Update Frequency Interval	ค่าความถี่ของการปรับปรุงข้อมูล (ความถี่น้อยที่สุด)	ผู้ใช้ระบุ
10	geo_coverage	Geo Coverage	ขอบเขตเชิงภูมิศาสตร์หรือเชิงพื้นที่	ไม่มี โลก ทวีป/กลุ่มประเทศในทวีป

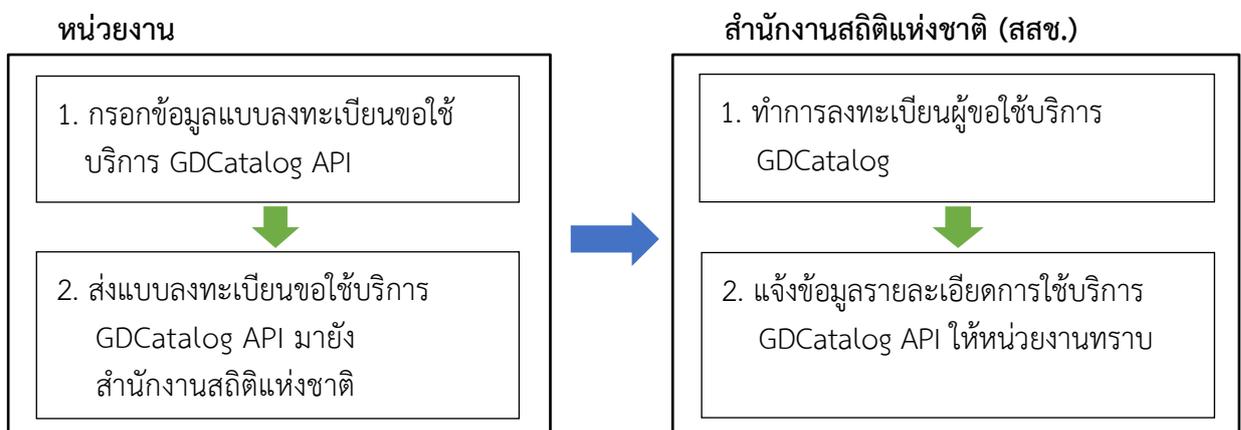
ลำดับ	ชื่อทางเทคนิค (Field name)	ชื่อรายการภาษาอังกฤษ	ชื่อรายการ ภาษาไทย	Value
				กลุ่มประเทศทางเศรษฐกิจ
				ประเทศ
				ภาค
				จังหวัด
				อำเภอ
				ตำบล
				หมู่บ้าน
				เทศบาล/อบต.
				พิกัด
				ไม่ทราบ
				อื่นๆ
11	data_source	Data Source	แหล่งที่มา	ผู้ใช้ระบุ
12	data_format	Data Format	รูปแบบการเก็บ ข้อมูล	ไม่ทราบ
				Database
				CSV
				XML
				Image
				Video
				Audio
				Text
				JSON
				HTML
				XLS
				PDF
				RDF
				NoSQL
				Arc/Info Coverage
				Shapefile
				GeoTiff
				GML
				อื่นๆ

ลำดับ	ชื่อทางเทคนิค (Field name)	ชื่อรายการภาษาอังกฤษ	ชื่อรายการภาษาไทย	Value
13	data_category	Data Category	หมวดหมู่ข้อมูลตามธรรมชาติ บาลข้อมูลภาครัฐ	ข้อมูลสาธารณะ
				ข้อมูลส่วนบุคคล
				ข้อมูลความมั่นคง
				ข้อมูลความลับทางราชการ
14	license_id	License	สัญญาอนุญาตให้ใช้ข้อมูล	License not specified
				Creative Commons Attributions
				Creative Commons Attribution Share-Alike
				Creative Commons Non-Commercial (Any)
				Open Data Common
				GNU Free Documentation License
				อื่นๆ

5. การแจ้งลงทะเบียนใช้งาน GDCatalog API

การขอใช้บริการเชื่อมต่อข้อมูลผ่าน API ของระบบ GD Directory Service นั้นผู้ขอใช้บริการจะต้องทำการลงทะเบียนขอ Client ID, Client Secret, Username, Password สำหรับใช้ทำการ Authorization ก่อนถึงจะเข้าถึงข้อมูลผ่าน API ของระบบได้

- 5.1. แจ้งลงทะเบียนการใช้งานโดยกรอกข้อมูลแบบลงทะเบียนและส่งมายังสำนักงานสถิติแห่งชาติ (โหลดแบบลงทะเบียนได้จาก https://gdhelppage.nso.go.th/data/04/files/api_register_form.pdf)
- 5.2. สำนักงานสถิติแห่งชาติลงทะเบียนผู้ใช้ API และ แจ้งข้อมูลรายละเอียดการเชื่อมต่อให้กับผู้พัฒนาระบบ



6. รูปแบบบริการ GDCatalog API

- 6.1. CKAN API เป็น API ที่เปิดให้หน่วยงานภาครัฐสามารถเข้าถึงและปรับปรุงข้อมูลบน GDCatalog
- 6.2. Directory API เป็น API เปิดให้หน่วยงานภาครัฐสามารถเข้าถึงบริการนามานุกรมชุดข้อมูลและข้อมูล

7. ขั้นตอนโดยย่อการใช้งานด้วยชุดคำสั่ง

7.1. กรณีใช้ CKAN API

- 1) เลือกวิธีการตรวจสอบการเข้าถึง API เป็นแบบ API-Key หรือ แบบ Token
- 2) การเรียกใช้ชื่อฟังก์ชัน และ ตัวแปร (Parameter) ที่ใช้ตามข้อกำหนดของฟังก์ชัน
- 3) การอ่าน JSON Object เพื่อใช้งาน

7.2. กรณี Directory API

- 1) ต้องร้องขอรับ Token Key
- 2) การรีเฟรช Token Key และ ตัวแปร (Parameter) ที่ใช้ตามข้อกำหนดของฟังก์ชัน
- 3) การเรียกใช้ชื่อฟังก์ชัน และ ตัวแปร (Parameter) ที่ใช้ตามข้อกำหนดของฟังก์ชัน
- 4) การอ่าน JSON Object เพื่อใช้งาน

8. เครื่องมือที่ใช้ในการทดสอบการเชื่อมต่อ API

การเชื่อมต่อด้วย API จำเป็นต้องใช้เครื่องมือเพื่อทดสอบการเรียกใช้งาน ปัจจุบันมีเครื่องมือที่ช่วยนักพัฒนาที่ได้รับความนิยม เช่น Postman (<https://learning.postman.com/docs/getting-started/introduction>) หรือ CURL (<https://curl.se/docs>) ซึ่งนักพัฒนาสามารถโหลดจากผู้พัฒนาเครื่องมือได้โดยตรง

9. CKAN API

CKAN API สามารถเรียกดูจาก <https://docs.ckan.org/en/2.9/api/index.html>

ฟังก์ชัน	รายละเอียด	Authorization Roles
package_list	แสดงรายการชื่อชุดข้อมูล *	-
current_package_list_with_resources	แสดงรายการรายละเอียดชุดข้อมูล พร้อมทรัพยากร *	-
member_list	แสดงรายการสมาชิก เช่น ผู้ใช้ในองค์กร, ชุดข้อมูลในองค์กร, ชุดข้อมูลในกลุ่มชุดข้อมูล *	-
group_list	แสดงรายการกลุ่มชุดข้อมูล	-
organization_list	แสดงรายการหน่วยงาน	-
tag_list	แสดงรายการคำสำคัญ	-
user_list	แสดงรายการผู้ใช้	API-Key User
package_show	แสดงรายละเอียดชุดข้อมูลที่ต้องการ *	-

ฟังก์ชัน	รายละเอียด	Authorization Roles
resource_show	แสดงรายละเอียดทรัพยากรที่ต้องการ *	-
group_show	แสดงรายละเอียดกลุ่มชุดข้อมูลที่ต้องการ	-
organization_show	แสดงรายละเอียดหน่วยงานที่ต้องการ	-
group_package_show	แสดงรายการชุดข้อมูลในกลุ่มชุดข้อมูลที่ต้องการ *	-
user_show	แสดงรายละเอียดผู้ใช้ที่ต้องการ	API-Key User
package_search	แสดงรายละเอียดชุดข้อมูลที่สัมพันธ์กับคำค้น *	-
resource_search	แสดงรายละเอียดทรัพยากรที่สัมพันธ์กับคำค้น *	-
package_create	สร้างชุดข้อมูล	API-Key Editor
resource_create	สร้างทรัพยากรในชุดข้อมูล	API-Key Editor
member_create	เพิ่มชุดข้อมูลเข้ากลุ่มชุดข้อมูล	API-Key Editor
group_create	สร้างกลุ่มชุดข้อมูล	API-Key Sysadmin
organization_create	สร้างหน่วยงาน	API-Key Sysadmin
user_create	สร้างผู้ใช้	API-Key Sysadmin
organization_member_create	เพิ่มผู้ใช้เข้าหน่วยงาน	API-Key Sysadmin
resource_update resource_patch	ปรับปรุงทรัพยากรในชุดข้อมูล	API-Key Editor
package_update package_patch	ปรับปรุงชุดข้อมูล	API-Key Editor
group_update	ปรับปรุงกลุ่มชุดข้อมูล	API-Key Sysadmin
organization_update organization_patch	ปรับปรุงหน่วยงาน	API-Key Sysadmin
user_update	ปรับปรุงผู้ใช้	API-Key Sysadmin
user_generate_apikey	สร้าง API-Key ให้กับผู้ใช้	API-Key Sysadmin
user_delete	ลบผู้ใช้	API-Key Sysadmin
package_delete	ลบชุดข้อมูล	API-Key Editor
resource_delete	ลบทรัพยากรในชุดข้อมูล	API-Key Editor
member_delete	นำชุดข้อมูลออกจากกลุ่ม	API-Key Editor

ฟังก์ชัน	รายละเอียด	Authorization Roles
group_delete	ลบกลุ่มชุดข้อมูล	API-Key Sysadmin
organization_delete	ลบหน่วยงาน	API-Key Sysadmin
organization_member_delete	นำผู้ใช้ออกจากหน่วยงาน	API-Key Sysadmin

หมายเหตุ * API ที่เกี่ยวกับชุดข้อมูลอาจจะต้องใช้ API-Key ในกรณีแสดงผลชุดข้อมูลที่เป็น private

ตัวอย่างการใช้ชุดคำสั่งภาษา Python ที่ใช้ API-Key

1) การเขียนโปรแกรมภาษา Python เรียกใช้ CKAN API แบบ POST เพื่อสร้างชุดข้อมูล

```

1 import json
2 import requests
3 ckan_url = 'http://xxx.go.th'
4 api_key = 'xxx'
5 dataset_dict = {'name': 'dataset_04',
6                 'data_type': 'ข้อมูลระเบียน',
7                 'title': 'ชื่อชุดข้อมูล',
8                 'owner_org': 'dep-1',
9                 'allow_harvest': 'True',
10                'maintainer': 'หน่วยงานเจ้าของข้อมูล',
11                'maintainer_email': 'maintainer@mail.go.th',
12                'tag_string': ['แท็ก 1', 'แท็ก 2'],
13                'notes': 'รายละเอียดชุดข้อมูล',
14                'objective': ['อื่นๆ'],
15                'objective_other': 'เพื่อสนับสนุนภาครัฐ',
16                'update_frequency_unit': 'ตามเวลาจริง',
17                'geo_coverage': 'อื่นๆ',
18                'geo_coverage_other': 'อื่นๆ',
19                'data_source': 'จากการดำเนินงาน',
20                'data_format': ['JSON', 'XLS'],
21                'data_category': 'ข้อมูลสาธารณะ',
22                'license_id': 'Open Data Common',
23                'private': True}
24 headers = {'Content-type': 'application/json', 'Authorization': api_key}
25 response = requests.post(url = ckan_url+'/api/3/action/package_create', data =
26                          json.dumps(dataset_dict), headers = headers)
27 response.json()

```

โค้ดโปรแกรมดังกล่าวสามารถอธิบายได้ดังนี้

- บรรทัดที่ 1 และบรรทัดที่ 2 เป็นการประกาศการใช้ python library json และ requests
- บรรทัดที่ 3 กำหนดค่าตัวแปร url ของ catalog
- บรรทัดที่ 4 กำหนดค่าตัวแปร api-key ของผู้ใช้ catalog
- บรรทัดที่ 5 จัดเตรียมค่า metadata ของชุดข้อมูลที่จะทำการสร้าง
- บรรทัดที่ 6 จัดเตรียมค่า headers ของการส่ง http-request
- บรรทัดที่ 7 ส่งค่า metadata และ header ผ่าน post request CKAN API เพื่อสร้างชุดข้อมูล โดยใช้ package_create
- บรรทัดที่ 8 แสดงผลค่า response เป็น log ประมวลผลการสร้างชุดข้อมูล

2) การเขียนโปรแกรมภาษา Python เรียกใช้ CKAN API แบบ GET เพื่อแสดงชุดข้อมูลที่ต้องการ

```
1 import json
2 import requests
3 ckan_url = 'http://xxx.go.th'
4 api_key = 'xxx'
5 data_dict = {'id': 'dataset_04'}
6 headers = {'Content-type': 'application/json', 'Authorization': api_key}
7 response = requests.get(url = ckan_url+'/api/3/action/package_show', data =
  json.dumps(data_dict), headers = headers)
8 response.json()
```

โค้ดโปรแกรมดังกล่าวสามารถอธิบายได้ดังนี้

- บรรทัดที่ 1 และบรรทัดที่ 2 เป็นการประกาศการใช้ python library json และ requests
- บรรทัดที่ 3 กำหนดค่าตัวแปร url ของ catalog
- บรรทัดที่ 4 กำหนดค่าตัวแปร api-key ของผู้ใช้ catalog (สำหรับ api ที่จำเป็นต้องส่ง api-key)
- บรรทัดที่ 5 จัดเตรียมค่า parameter ที่ต้องส่งให้กับ api
- บรรทัดที่ 6 จัดเตรียมค่า headers ของการส่ง http-request
- บรรทัดที่ 7 ส่งค่า parameter และ header ผ่าน get request CKAN API เพื่อแสดงชุดข้อมูลโดยใช้ package_show
- บรรทัดที่ 8 แสดงผลค่า response เป็น log ประมวลผลการแสดงชุดข้อมูล

10. Directory API

10.1. รายการ API GD Directory Service ที่เปิดให้บริการนั้นสามารถเรียกดูได้ที่ <https://api.gdcatalog.go.th/v1/api> มี Name Space ทางฟังก์ชันดังนี้

ฟังก์ชัน	รายละเอียด	Authorization
/dataset/list_all	เรียกดูรายละเอียดของชุดข้อมูล โดยมี 3 Parameters ที่ต้องกำหนด 1. option คือ การแสดงข้อมูล ซึ่งมีอยู่ 3 รูปแบบ - full = แสดงคำอธิบายชุดข้อมูลและทรัพยากรของชุดข้อมูล - mandatory = แสดงเฉพาะคำอธิบายชุดข้อมูล 14 ตัว - none = แสดงเฉพาะชื่อชุดข้อมูล) 2. rows คือ จำนวนชุดข้อมูล 3. start คือ หน้าที่ต้องการโดยเริ่มจาก 0	client_id, client_secret, username, password grant_type
/dataset/by_id	เรียกดูรายละเอียดของชุดข้อมูล โดยมี 2 Parameters ที่ต้องกำหนด 1. dataset_id คือ id ของชุดข้อมูล 2. option คือ การแสดงข้อมูล ซึ่งมีอยู่ 3 รูปแบบ full, mandatory, none	client_id, client_secret, username, password grant_type
/dataset/by_name	เรียกดูรายละเอียดของชุดข้อมูล โดยมี 2 Parameters ที่ต้องกำหนด 1. title คือ ชื่อของชุดข้อมูล 2. option คือ การแสดงข้อมูล ซึ่งมีอยู่ 3 รูปแบบ full, mandatory, none	client_id, client_secret, username, password grant_type
/dataset/by_meta	เรียกดูรายละเอียดของชุดข้อมูล โดยมี 5 Parameters ที่ต้องกำหนด 1. option คือ การแสดงข้อมูล ซึ่งมีอยู่ 3 รูปแบบ full, mandatory, none	client_id, client_secret, username, password grant_type

ฟังก์ชัน	รายละเอียด	Authorization
	2. key meta_name คือ key ของ metadata ที่ต้องการ 3. value meta_name คือ ข้อมูลของ metadata ที่ต้องการ 4. rows คือ จำนวนชุดข้อมูล 5. start คือ หน้าที่ต้องการโดยเริ่มจาก 0	
/dataset/by_ministry	เรียกดูรายละเอียดของชุดข้อมูล โดยมี 4 Parameters ที่ต้องกำหนด 1. option คือ การแสดงข้อมูล ซึ่งมีอยู่ 3 รูปแบบ full, mandatory, none 2. ministry_id คือ id ของกระทรวง (ในที่นี้ได้ตั้งค่าเป็น Dropdown ไว้) 3. rows คือ จำนวนชุดข้อมูล 4. start คือ หน้าที่ต้องการโดยเริ่มจาก 0	client_id, client_secret, username, password grant_type
/dataset/by_department	เรียกดูรายละเอียดของชุดข้อมูล โดยมี 4 Parameters ที่ต้องกำหนด 1. department_id คือ id ของหน่วยงาน (ในที่นี้ได้ตั้งค่าเป็น Dropdown ไว้) 2. option คือ การแสดงข้อมูล ซึ่งมีอยู่ 3 รูปแบบ full, mandatory, none 3. rows คือ จำนวนชุดข้อมูล 4. start คือ หน้าที่ต้องการโดยเริ่มจาก 0	client_id, client_secret, username, password grant_type
/resource/list_all	เรียกดูรายละเอียดทรัพยากรของชุดข้อมูล โดยมี 1 Parameters ที่ต้องกำหนด dataset_id คือ id ของชุดข้อมูล	client_id, client_secret, username, password grant_type
/resource/by_id	เรียกดูรายละเอียดทรัพยากรของชุดข้อมูล โดยมี 1 Parameters ที่ต้องกำหนด resource_id คือ id ของทรัพยากรข้อมูล	client_id, client_secret, username, password grant_type

ฟังก์ชัน	รายละเอียด	Authorization
/resource/by_name	เรียกดูรายละเอียดทรัพยากรของชุดข้อมูล โดยมี 1 Parameters ที่ต้องกำหนด name คือ ชื่อของทรัพยากรข้อมูล	client_id, client_secret, username, password grant_type
/resource/by_ministry	เรียกดูรายละเอียดทรัพยากรของชุดข้อมูล โดยมี 1 Parameters ที่ต้องกำหนด ministry_id คือ id ของกระทรวง (ในที่นี้ได้ตั้งค่าเป็น Dropdown)	client_id, client_secret, username, password grant_type
/resource/by_department	เรียกดูรายละเอียดทรัพยากรของชุดข้อมูล โดยมี 1 Parameters ที่ต้องกำหนด department_id คือ id ของหน่วยงาน (ในที่นี้ได้ตั้งค่าเป็น Dropdown ไว้)	client_id, client_secret, username, password grant_type
/data/content	เรียกดูข้อมูลที่อยู่ในทรัพยากรของชุดข้อมูล โดยมี 1 Parameters ที่ต้องกำหนด resource_id คือ id ของทรัพยากรข้อมูล	client_id, client_secret, username, password grant_type
/data/dict	เรียกดูข้อมูลหัวตารางที่อยู่ในทรัพยากรของชุดข้อมูล โดยมี 1 Parameters ที่ต้องกำหนด resource_id คือ id ของทรัพยากรข้อมูล	client_id, client_secret, username, password grant_type
/report/totaldataset_all	เรียกดูจำนวนชุดข้อมูลทั้งหมด โดยจะแสดงข้อมูลของแต่ละหน่วยงาน โดยมี 1 Parameters ที่ต้องกำหนด ckan_url คือ url ของ ckan ที่ต้องการดูรายงาน	client_id, client_secret, username, password grant_type
/report/totaldataset_by_department	เรียกดูจำนวนชุดข้อมูลทั้งหมดของหน่วยงาน โดยมี 2 Parameters ที่ต้องกำหนด 1. department_id คือ id ของหน่วยงาน (ในที่นี้ได้ตั้งค่าเป็น Dropdown ไว้)	client_id, client_secret, username, password grant_type

ฟังก์ชัน	รายละเอียด	Authorization
	2. ckan_url คือ url ของ ckan ที่ต้องการดูรายงาน	
/report/totaldataset_by_ministry	เรียกดูจำนวนชุดข้อมูลทั้งหมดของกระทรวง โดยมี 2 Parameters ที่ต้องกำหนด 1. ministry_id คือ id ของกระทรวง (ในที่นี้ได้ตั้งค่าเป็น Dropdown) 2. ckan_url คือ url ของ ckan ที่ต้องการดูรายงาน	client_id, client_secret, username, password grant_type
/report/totaldataset_format_all	เรียกดูจำนวนทั้งหมดของไฟล์แต่ละชนิด โดยมี 1 Parameters ที่ต้องกำหนด ckan_url คือ url ของ ckan ที่ต้องการดูรายงาน	client_id, client_secret, username, password grant_type
/report/totaldataset_format_by_department	เรียกดูจำนวนทั้งหมดของไฟล์แต่ละชนิดของหน่วยงาน โดยมี 2 Parameters ที่ต้องกำหนด 1. department_id คือ id ของหน่วยงาน (ในที่นี้ได้ตั้งค่าเป็น Dropdown ไว้) 2. ckan_url คือ url ของ ckan ที่ต้องการดูรายงาน	client_id, client_secret, username, password grant_type
/report/totaldataset_format_by_ministry	เรียกดูจำนวนทั้งหมดของไฟล์แต่ละชนิดของกระทรวง โดยมี 2 Parameters ที่ต้องกำหนด 1. ministry_id คือ id ของกระทรวง (ในที่นี้ได้ตั้งค่าเป็น Dropdown) 2. ckan_url คือ url ของ ckan ที่ต้องการดูรายงาน	client_id, client_secret, username, password grant_type
/report/totaltag_all	เรียกดูจำนวนรายการคำสำคัญทั้งหมด โดยมี 1 Parameters ที่ต้องกำหนด ckan_url คือ url ของ ckan ที่ต้องการดูรายงาน	client_id, client_secret, username, password grant_type

ฟังก์ชัน	รายละเอียด	Authorization
/report/totaltag_by_department	เรียกดูจำนวนรายการคำสำคัญทั้งหมดของหน่วยงาน โดยมี 2 Parameters ที่ต้องกำหนด 1. department_id คือ id ของหน่วยงาน (ในที่นี้ได้ตั้งค่าเป็น Dropdown ไว้) 2. ckan_url คือ url ของ ckan ที่ต้องการดูรายงาน	client_id, client_secret, username, password grant_type
/report/totaltag_by_ministry	เรียกดูจำนวนรายการคำสำคัญทั้งหมดของกระทรวง โดยมี 2 Parameters ที่ต้องกำหนด 1. ministry_id คือ id ของกระทรวง (ในที่นี้ได้ตั้งค่าเป็น Dropdown) 2. ckan_url คือ url ของ ckan ที่ต้องการดูรายงาน	client_id, client_secret, username, password grant_type

GDS Api 1.0.0 OAS3

GDcatalog Directory Service API

[Contact the developer](#)

Servers

<https://api.gdcatalog.go.th/v1> - GDcatalog Directory Service API

Authorize

dataset

GET	/api/dataset/list_all	Get list of Ckan	▼	🔒
GET	/api/dataset/by_id/{dataset_id}	Get list by_id	▼	🔒
GET	/api/dataset/by_name/{name}	Get list by_name	▼	🔒
GET	/api/dataset/by_meta	Get list by_meta	▼	🔒
GET	/api/dataset/by_ministry	Get list by_ministry	▼	🔒
GET	/api/dataset/by_department/{department_id}	Get list by_department	▼	🔒

resource

GET	/api/resource/list_all/{dataset_id}	Get resource all	▼	🔒
GET	/api/resource/by_id/{resource_id}	Get resource by_id	▼	🔒
GET	/api/resource/by_name/{word}	Get resource by_name	▼	🔒
GET	/api/resource/by_ministry/{ministry_id}	Get resource by_ministry	▼	🔒
GET	/api/resource/by_department/{department_id}	Get resource by_department	▼	🔒

data

GET	/api/data/content/{resource_id}	Get data content	▼	🔒
GET	/api/data/dict/{resource_id}	Get data dict	▼	🔒

report

GET	/api/report/totaldataset_all	Get report totaldataset_all	▼	🔒
GET	/api/report/totaldataset_by_department	Get report totaldataset_by_department	▼	🔒
GET	/api/report/totaldataset_by_ministry	Get report totaldataset_by_ministry	▼	🔒
GET	/api/report/totaldataset_format_all	Get report totaldataset_format_all	▼	🔒
GET	/api/report/totaldataset_format_by_department	Get report totaldataset_format_by_department	▼	🔒
GET	/api/report/totaldataset_format_by_ministry	Get report totaldataset_format_by_ministry	▼	🔒
GET	/api/report/totaltag_all	Get report totaltag_all	▼	🔒
GET	/api/report/totaltag_by_department	Get report totaltag_by_department	▼	🔒
GET	/api/report/totaltag_by_ministry	Get report totaltag_by_ministry	▼	🔒

3) ตัวอย่างการใช้งานการ Request API โดยใช้ Library Axios ในการติดต่อสื่อสาร HTTP

```
Axios.get(`${process.env.NEXT_PUBLIC_APIURL}/*ลิ้ง API ตามด้วยพาทของ API*/}/ckan/all?`, {
  headers: {
    'Authorization': `Bearer ${Token}` /* แบนโทเค็นที่ได้จากการล็อกอินผ่านระบบ Keycloak*/
  },
}).then(({ data: { data } }) => {
  //หากการ request API สำเร็จจะเข้าทำงานที่ฟังก์ชันนี้ และจะรีเทรินค่าข้อมูลที่ส่งมาจาก API
  ผ่านตัวแปรที่ชื่อ data
  console.log('data :>> ', data);
  dispatch(SET_LOADING(false));

}).catch((error) => {
  //หากเกิดข้อผิดพลาดจะทำให้รีเทรินค่าเข้ามาที่ฟังก์ชัน catch และแสดงผล error ผ่านตัวแปรที่ชื่อ error
  console.log('error :>> ', error);
  dispatch(SET_LOADING(false));
})
```

4) ตัวอย่างการใช้งานการ Refresh API โดยใช้ Library Axios ในการติดต่อสื่อสาร HTTP

การ refresh token ใช้ param ดังนี้

```
'client_id': 'หมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้',
'client_secret': 'รหัสหมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้',
'grant_type': 'refresh_token',
'refresh_token': token,
```

```
import axios from 'axios';
import { Cookies } from 'react-cookie'
import { GetAPIkeyCloak } from '../service/API';
import { setAccessToken } from "../util/Utility";

const token = new Cookies().get('token');
const cookies = new Cookies();
```

```

export default axios.create({
  baseURL: process.env.NEXT_PUBLIC_APIURL,
  headers: {
    'Content-Type': 'application/json',
    // 'Authorization': Bearer ${token?.token}
  },
  transformRequest: [function (data, headers) {
    // Do whatever you want to transform the data
    const getopenid = cookies.get('openid');
    const token = cookies.get('token');
    if (token) {
      if (new Date() > new Date(new Date(token.time_stamps?.date).getTime() +
(token.refresh_expires_in-1000) * 1000).getTime() && new Date() < new Date(new
Date(token.time_stamps?.date).getTime() + (token.refresh_expires_in) * 1000).getTime())
      {
        console.log("หมดเวลา token")
        RefreshToken(token.refresh_token);
      } else {
        console.log("ยังไม่รีเฟรช");
        // RefreshToken(token.refresh_token)
      }
    }

    if (token) {
      if (getopenid) {
        headers.Authorization = "Bearer " + getopenid.token;
      } else {
        headers.Authorization = "Bearer " + token.token;
      }
    }
    return JSON.stringify(data);
  }],
});

```

```

const RefreshToken = async(refreshtokenval) => {
  let succesdata;
  const getapi = () => axios.post(process.env.NEXT_PUBLIC_APIURL+'/keycloak', {
    grant_type: "refresh_token",
    client_id: "หมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้",
    client_secret: "รหัสหมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้",
    refresh_token:refreshtokenval
  }).then(({ data }) => {
    console.log('datatoken :>> ', data);
    return data.data
  }).catch((error)=>{
    console.log('errortokenrefresh :>> ', error);
    GetAPIkeyCloak();
  })
  succesdata = getapi();
  let gettoken = await getapi().then((data) => data);
  console.log(gettoken)
  await cookies.set('token', gettoken);
  await setAccessToken(gettoken.token);
  location.reload();
  return succesdata;
}

```

```

export const GetAPIkeyCloak = async () => { //---ดึงข้อมูล token จาก KeyCloak
  // let gettoken = JSON.parse(localStorage.getItem ('KEYCLOAK'));
  let gettoken = cookies.get('token');
  let succesdata;
  console.log('gettoken :>> ', gettoken);
  let GETKEYCLOAK = async () => {
    const getapi = () => API.post('/keycloak', {
      grant_type: "password",
      client_id: "หมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้",
      client_secret: "รหัสหมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้",
      username: "ชื่อผู้ใช้ที่สำนักงานสถิติแห่งชาติกำหนดให้",
      password: "รหัสผ่านที่สำนักงานสถิติแห่งชาติกำหนดให้",
    }).then(({ data }) => data.data);
    succesdata = getapi();
    let gettokenkeycloak = await getapi().then((data) => data)
    await cookies.set('token', gettokenkeycloak);
    await setAccessToken(gettokenkeycloak.token);
    return succesdata;
  }
  if (gettoken !== null && gettoken !== undefined) {
    let checktokentime =
      new Date() > new Date(new Date(gettoken?.time_stamps?.date)
        .getTime() + gettoken.expires_in * 1000).getTime()
    // console.log('checktokentime :>> ', checktokentime);
    if (checktokentime) {
      GETKEYCLOAK()
    } else {
      setAccessToken(gettoken.token);
      succesdata = gettoken;
    }
  } else {
    GETKEYCLOAK()
  }
  return succesdata;
}
}

```

11. ตัวอย่างการใช้ชุดคำสั่งภาษา PHP

11.1. login เพื่อให้ได้ token ผ่าน <https://api.gdcatalog.go.th/v1/api/keycloak> โดยใช้ method POST และส่ง body เป็น form-data ดังนี้

- grant_type : required และต้องเป็น password หรือ refresh_token เท่านั้น password ใช้เมื่อ Client ยังไม่มี Token สำหรับเข้าถึงข้อมูลซึ่ง Token จะมีอายุประมาณ 30 นาที หลังจากนั้นระบบจะต้องทำการ Refresh token เพื่อให้ระบบยังสามารถเข้าถึงได้ต่อโดยการส่ง refresh_token ไปขอ Token ใหม่มา
- client_id : หมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้
- client_secret : รหัสหมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้
- username : ชื่อผู้ใช้ที่สำนักงานสถิติแห่งชาติกำหนดให้
- password : รหัสผ่านที่สำนักงานสถิติแห่งชาติกำหนดให้

11.2. ในการใช้งาน API จะใช้ Guzzle, PHP HTTP client (Guzzle เป็น HTTP Client ตัวหนึ่งของ PHP หน้าตาของมันก็เหมือน cURL คือเอาไปเชื่อมต่อไปยัง URL ต่างๆ (เช่นพวก REST API) รองรับ HTTP Transport หลายรูปแบบ ตั้งแต่ cURL, Socket, PHP Streams) ดูข้อมูลเพิ่มเติม <https://docs.guzzlephp.org> และ <https://wp63.co/2019/10/guzzle-php-http-client>

11.3. ตัวอย่างการใช้งาน

```
<?php
// token ที่ได้จากการ login key cloak
$token = "";
// ประการการใช้งาน GuzzleHttpClient
$client = new GuzzleHttp\Client(['base_uri' => 'https://api.gdcatalog.go.th/v1 ']);
// ส่ง header สองค่า Authoriaztion แบบ Bearer และ Accept
$headers = [
    'Authorization' => 'Bearer ' . $token,
    'Accept'       => 'application/json',
];
// parameter แรงคือ method จะใช้ method GET parameter ที่สอง คือ service ของ api
$response = $client->request('GET', '/api/dataset/list_all?option=full', [
    'headers' => $headers
]);
// จะ return แสดงค่า ของ รายการข้อมูล ถ้า token ไม่ผ่าน ก็จะ return 403 คือ Forbidden
return $response->getBody()->getContents();
```



```

#----- ระบุค่า Parameter ที่ต้องการส่งในฟังก์ชัน -----
$option = 'full';
$dataset_id = '08e14fb3-aaeb-4e79-9e03-ac251aaebdb9';
$word = 'คน';
$meta_name = 'unit_of_measure';
$ministry_id = 'กระทรวงดิจิทัลเพื่อเศรษฐกิจและสังคม';
$department_id = 'กรมการข้าว';
$resource_id = '7aaa6ffa-76f8-428e-8faa-6de690cdfcc7';
$start = 0;
$rows = 10;

#----- Dataset API function -----
function get_datset_all($option){
    $api = "api/dataset/list_all?option=".$option;
    return $api;
}
function get_dataset_by_id($dataset_id){
    $api = "api/dataset/by_id?dataset_id=".$dataset_id;
    return $api;
}
function get_dataset_by_name($word){
    $api = "api/dataset/by_name?word=".$word;
    return $api;
}
function get_dataset_by_meta($meta_name,$option,$word,$rows,$start){
    $api = "api/dataset/by_meta?meta_name=".$meta_name."&option=".$option."&w
ord=".$word."&rows=".$rows."&start=".$start;
    return $api;
}
function get_dataset_by_ministry($ministry_id){
    $api = "api/dataset/by_ministry?ministry_id=".$ministry_id;
    return $api;
}
function get_dataset_by_department($department_id){
    $api = "api/dataset/by_department?department_id=".$department_id;
    return $api;
}
}

```

```

#----- Resource API function -----
function get_resource_list_all($dataset_id) {
    $api = "api/resource/list_all?dataset_id=".$dataset_id;
    return $api;
}
function get_resource_by_id($resource_id){
    $api = "api/resource/by_id?resource_id=".$resource_id;
    return $api;
}
function get_resource_by_name($word){
    $api = "api/resource/by_name?word=".$word;
    return $api;
}
function get_resource_by_ministry($ministry_id){
    $api = "api/resource/by_ministry?ministry_id=".$ministry_id;
    return $api;
}
function get_resource_by_department($department_id){
    $api = "api/resource/by_department?department_id=".$department_id;
    return $api;
}

#----- Choose function -----
//$api = get_datset_all($option);
//$api = get_dataset_by_id($dataset_id);
//$api = get_dataset_by_name($word);
//$api = get_dataset_by_meta($meta_name,$option,$word,$rows,$start);
//$api = get_dataset_by_ministry($ministry_id);
//$api = get_dataset_by_department($department_id);

//$api = get_resource_list_all($dataset_id)
//$api = get_resource_by_id($resource_id);
//$api = get_resource_by_name($word);
//$api = get_resource_by_ministry($ministry_id);
//$api = get_resource_by_department($department_id);

```

```
//parameter แรกคือ method จะใช้ method GET parameter ที่สอง คือ service ของ api
$response = $client->request('GET', $api, [
    'headers' => $headers
]);
// จะ return แสดงค่า ของ รายการข้อมูล ถ้า token ไม่ผ่าน ก็จะ return 403 คือ Forbidden
$body = json_decode($response->getBody());
print_r($body);
?>
```

12. ตัวอย่างการใช้ชุดคำสั่งภาษา Python

12.1. login เพื่อให้ได้ token ผ่าน <https://api.gdcatalog.go.th/v1/api/keycloak> โดยใช้ method POST และส่ง body เป็น form-data ดังนี้

- grant_type : required และต้องเป็น password หรือ refresh_token เท่านั้น password ใช้เมื่อ Client ยังไม่มี Token สำหรับเข้าถึงข้อมูลซึ่ง Token จะมีอายุประมาณ 30 นาที หลังจากนั้นระบบจะต้องทำการ Refresh token เพื่อให้ระบบยังสามารถเข้าถึงได้ต่อโดยการส่ง refresh_token ไปขอ Token ใหม่มา
- client_id : หมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้
- client_secret : รหัสหมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้

12.2. ตัวอย่างการใช้งาน

```
import json
import requests
# ระบุ API ที่ต้องการเลือกใช้งาน
url = https://api.gdcatalog.go.th/v1/api/dataset/list_all/
# กำหนด header
headers = {"X-API-KEY" : "e090c757-4998-494e-8ce6-db5c39b80e32", "accept": "application/json"}
# "X-API-KEY" : id ที่ได้จาก Keycloak,
# "accept" : application/json (ข้อมูลที่รับเข้ามาในรูปแบบ JSON)
payload = {"data_listall" : "none", "rows" : 10, "start" : 0}
# กำหนด Parameters ที่ได้มีการกำหนดไว้ในส่วนของ API ที่ใช้งาน
response = requests.get(url, headers=headers, params=payload)
# กำหนด Parameters ตามรูปแบบ requests.get ดูเพิ่มเติมจาก
# https://docs.python-requests.org/en/master/user/quickstart/#make-a-request
# จะ return แสดงค่า ของ รายการข้อมูล หาก ค่า "X-API-KEY" ไม่ถูกต้อง ก็จะ return 401 คือ
# Unauthorized (ไม่อนุญาตให้สามารถใช้ API)
return response
```

12.3. ตัวอย่างการใช้งาน

```
import requests
from requests.structures import CaseInsensitiveDict
import ast

def get_token():
    url = 'https://api.gdcatalog.go.th/v1/api/keycloak'
    params = {
        'client_id': 'หมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้',
        'client_secret': 'รหัสหมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้',
        'grant_type': 'password',
        'username': 'ชื่อผู้ใช้ที่สำนักงานสถิติแห่งชาติกำหนดให้',
        'password': 'รหัสผ่านที่สำนักงานสถิติแห่งชาติกำหนดให้'
    }
    response = requests.post(url, params, verify=True).content.decode('utf-8')
    token = ast.literal_eval(response)['data']['token']
    print(" Get Token :", token)
    print('\n')
    return token
    #return requests.post(url, params, verify=False).content.decode('utf-8')

def refresh_token(token):
    #print(token)
    url = 'https://api.gdcatalog.go.th/v1/api/keycloak'
    params = {
        'client_id': 'หมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้',
        'client_secret': 'รหัสหมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้',
        'grant_type': 'refresh_token',
        'refresh_token': token
    }
    #print(params)
    response = requests.post(url, params, verify=True)
    print ( "Refresh Token : ",response)
```

```

# ----- dataset -----

def get_dataset_all(token):
    url = 'https://api.gdcatalog.go.th/v1/api/dataset/list_all'
    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'option': 'full',
        'start': 0,
        'rows': 10
    }
    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

def get_dataset_by_id(token):
    url = 'https://api.gdcatalog.go.th/v1/api/dataset/by_id'
    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'dataset_id': '08e14fb3-aaeb-4e79-9e03-ac251aaebdb9',
        'option': 'full'
    }
    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

```

```

def get_dataset_by_name(token):
    url = 'https://api.gdcatalog.go.th/v1/api/dataset/by_name'
    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'word': 'จำนวนสถานประกอบการที่มีการใช้คอมพิวเตอร์'
    }
    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

def get_dataset_by_meta(token):
    url = 'https://api.gdcatalog.go.th/v1/api/dataset/by_meta'
    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'option': 'full',
        'meta_name': 'unit_of_measure',
        'word': 'คน',
        'start': 0,
        'rows': 10
    }
    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

```

```

def get_dataset_by_ministry(token):
    url = 'https://api.gdcatalog.go.th/v1/api/dataset/by_ministry'
    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'option': 'full',
        'ministry_id': 'กระทรวงดิจิทัลเพื่อเศรษฐกิจและสังคม',
        'start': 0,
        'rows': 10
    }
    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

def get_dataset_by_department(token):
    url = 'https://api.gdcatalog.go.th/v1/api/dataset/by_department'
    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'option': 'full',
        'department_id': 'กรมการข้าว',
        'start': 0,
        'rows': 10
    }
    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

```

```

# ----- resource -----

def get_resource_list_all(token):
    url = 'https://api.gdcatalog.go.th/v1/api/resource/list_all'
    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'dataset_id': 'dc8ca65c-2bdc-4cd4-b127-0e263079c192'
    }

    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

def get_resource_by_id(token):
    url = 'https://api.gdcatalog.go.th/v1/api/resource/by_id'
    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'resource_id': '1e4b40fd-583e-4c22-8719-28b36d1af20a'
    }

    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

```

```

def get_resource_by_name(token):
    url = 'https://api.gdcatalog.go.th/v1/api/resource/by_name'
    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'word': 'สถิติการผลิตเมล็ดพันธุ์ข้าว290764'
    }
    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print('Out Put')
    print(api_response);

def get_resource_by_ministry(token):
    url = 'https://api.gdcatalog.go.th/v1/api/resource/by_ministry'

    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'option': 'full',
        'ministry_id': 'กระทรวงดิจิทัลเพื่อเศรษฐกิจและสังคม',
        'start': 0,
        'rows': 10
    }
    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

```

```

def get_resource_by_department(token):
    url = 'https://api.gdcatalog.go.th/v1/api/resource/by_department'

    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'option': 'full',
        'department_id': 'กรมการข้าว',
        'start': 0,
        'rows': 10
    }
    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

# ----- data -----

def get_data_content(token):
    url = 'https://api.gdcatalog.go.th/v1/api/data/content'
    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'resource_id': '1e4b40fd-583e-4c22-8719-28b36d1af20a'
    }
    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

```

```

# ----- report -----

def get_report_totaldataset_all(token):
    url = 'https://api.gdcatalog.go.th/v1/api/report/totaldataset_all'
    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'ckan_url': 'https://gdcatalog.go.th/api/3/action/'
    }
    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

def get_report_totaldataset_by_department(token):
    url =
'https://api.gdcatalog.go.th/v1/api/report/totaldataset_by_department'
    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'ckan_url': 'https://gdcatalog.go.th/api/3/action/',
        'depart' : 'สำนักงานสถิติแห่งชาติ'
    }
    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

```

```

def get_report_totaldataset_by_ministry(token):
    url = 'https://api.gdcatalog.go.th/v1/api/report/totaldataset_by_ministry'
    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'ckan_url': 'https://gdcatalog.go.th/api/3/action/',
        'ministry': 'กระทรวงดิจิทัลเพื่อเศรษฐกิจและสังคม'
    }
    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

def get_report_totaldataset_format_all(token):
    url = 'https://api.gdcatalog.go.th/v1/api/report/totaldataset_format_all'
    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'ckan_url': 'https://gdcatalog.go.th/api/3/action/'
    }
    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

```

```

def get_report_totaldataset_format_by_department(token):
    url =
'https://api.gdcatalog.go.th/v1/api/report/totaldataset_format_by_department'
    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'ckan_url': 'https://gdcatalog.go.th/api/3/action/',
        'depart' : 'สำนักงานสถิติแห่งชาติ'
    }
    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

def get_report_totaldataset_format_by_ministry(token):
    url =
'https://api.gdcatalog.go.th/v1/api/report/totaldataset_format_by_ministry'

    headersAPI = CaseInsensitiveDict();
    headersAPI = {
        'accept': 'application/json',
        'Authorization': 'Bearer ' + token
    }
    params = {
        'ckan_url': 'https://gdcatalog.go.th/api/3/action/',
        'ministry': 'กระทรวงดิจิทัลเพื่อเศรษฐกิจและสังคม'
    }
    response = requests.get(url, headers=headersAPI, params=params,
verify=True)
    api_response = response.json()
    print(api_response);

```

```
# -----Get Token-----
token = get_token()

# Get Data Dataset all
get_datset_all(token)

# ---- data set name space ----
get_dataset_by_id(token)
get_dataset_by_name(token)
get_dataset_by_meta(token)
get_dataset_by_ministry(token)
get_dataset_by_department(token)

# ---- resource name space ----
get_resource_list_all(token)
get_resource_by_id(token)
get_resource_by_name(token)
get_resource_by_ministry(token)
get_resource_by_department(token)

# ---- data space ----
get_data_content(token)
get_report_totaldataset_all(token)
get_report_totaldataset_by_department(token)
get_report_totaldataset_by_ministry(token)
get_report_totaldataset_format_all(token)
get_report_totaldataset_format_by_department(token)
get_report_totaldataset_format_by_ministry(token)
```

12.4 ตัวอย่างการใช้งานการ Refresh API โดยใช้การติดต่อสื่อสาร HTTPS

การ refresh token ใช้ param

```
'client_id': 'หมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้',  
'client_secret': 'รหัสหมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้',  
'grant_type': 'refresh_token',  
'refresh_token': token,
```

```
def refresh_token(token):  
    #print(token)  
    url = 'https://api.gdcatalog.go.th/v1/api/keycloak'  
    params = {  
        'client_id': 'หมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้',  
        'client_secret' : 'รหัสหมายเลขที่สำนักงานสถิติแห่งชาติกำหนดให้',  
        'grant_type': 'refresh_token',  
        'refresh_token': token  
    }  
    #print(params)  
    response = requests.post(url, params, verify=True)  
    print ( "Refresh Token : ",response)
```