

“เทคนิคการเชื่อมโยงข้อมูลภาครัฐ”

โครงการลดสำเนากระดาษเพื่อบริการประชาชน นำนํ่อง ๗ กระทรวง
Smart Service



สำนักงานรัฐบาลอิเล็กทรอนิกส์ (องค์การมหาชน) (สโร.)

Electronic Government Agency (Public Organization) (EGA)

ที่มา

การลดสำเนาเอกสารราชการเพื่ออำนวยความสะดวกให้แก่ประชาชน

หนังสือกระทรวงมหาดไทย เรื่อง การอำนวยความสะดวกให้แก่ประชาชนเกี่ยวกับการใช้สำเนาทะเบียนบ้านและบัตรประจำตัวประชาชน ลงวันที่ ๓ พ.ย. ๒๕๕๗ ระบุ "ห้ามมิให้เจ้าหน้าที่เรียกสำเนาภาพถ่ายเอกสารหรือแนะนำให้ผู้ร้องไปขอคัดสำเนารายการทะเบียนบ้านหรือบัตรฯ ... แต่ให้เจ้าหน้าที่คัดสำเนารายการทะเบียนบ้านหรือรายการบัตรประจำตัวประชาชนจากฐานข้อมูลทะเบียนและบัตรฯ ในระบบคอมพิวเตอร์ของสำนักทะเบียนนั้นแทนโดยไม่ต้องลงชื่อรับรองเอกสารก็ได้"

หากหน่วยงานผู้ให้บริการ สามารถเข้าถึงฐานข้อมูลทะเบียนและบัตรฯ ของกรมการปกครอง หรือฐานข้อมูลของหน่วยงานอื่นที่เกี่ยวข้องได้ โดยใช้บัตรประจำตัวประชาชนเอนกประสงค์ (Smart Card) ของผู้ร้องในการเข้าถึงข้อมูล ก็สามารถลดการขอสำเนาเอกสารราชการอื่นๆ ที่เกี่ยวข้องได้เช่นกัน

Smart e-Service : คำนิยาม

ประชาชนสามารถเข้าถึงบริการได้ โดยไม่ต้อง
ใช้สำเนาบัตรประชาชน สำเนาทะเบียนบ้าน หรือ
เอกสารราชการอื่นๆ ที่เกี่ยวข้อง / หรือ

ประชาชนสามารถเข้าถึงบริการด้วยตนเอง
(Self-Service) ผ่านช่องทางออนไลน์ โทรศัพท์
เคลื่อนที่ (Smart Phones) หรือตู้เอนกประสงค์
(Kiosk)

Smart e-Service : ประโยชน์ที่ได้รับ

ประชาชน / ผู้ขอรับบริการ

- อำนวยความสะดวกในการขอรับบริการ
- ลดค่าใช้จ่ายในการทำสำเนา

ภาครัฐ

- ลดการบันทึกข้อมูลของเจ้าหน้าที่
- ข้อมูลมีความถูกต้อง น่าเชื่อถือ
- ภาพลักษณ์ที่ดีขึ้นของภาครัฐในการให้บริการประชาชน

เป้าหมายโครงการ

- ปรับปรุงบริการภาครัฐไปสู่ Smart Service ที่ประชาชนสามารถขอรับบริการโดยไม่จำเป็นต้องยื่นสำเนาบัตรประจำตัวประชาชนหรือสำเนาทะเบียนบ้าน
- มีการใช้ประโยชน์จากเลขประจำตัวประชาชน ๑๓ หลักหรือบัตร Smart Card โดยข้อมูลบุคคลที่เกี่ยวข้องจะถูกบันทึกลงแบบฟอร์มขอใช้บริการ เพื่อลดการกรอกแบบฟอร์มกระดาษ
- มีช่องทางสำหรับเข้าถึงข้อมูลของรัฐในรูปแบบ Government API เพื่อนำไปใช้ในการสนับสนุนการให้บริการของภาครัฐ
- มีแอปพลิเคชันนำร่อง อาทิ แอปพลิเคชันให้ประชาชนสามารถตรวจสอบสิทธิในการเข้าถึงมาตรการ และนโยบายต่างๆ ของรัฐบาลได้

กลุ่มเป้าหมาย

โดยมีกลุ่มเป้าหมายในการดำเนินโครงการ ดังนี้
กลุ่มเป้าหมายหลัก ได้แก่ หน่วยงานราชการ ภายใต้กระทรวงนำร่องทั้ง ๗ กระทรวง

- กระทรวงเกษตรและสหกรณ์การเกษตร
- กระทรวงเทคโนโลยีสารสนเทศและการสื่อสาร
- กระทรวงการคลัง
- กระทรวงการท่องเที่ยวและการกีฬา
- กระทรวงพาณิชย์
- กระทรวงมหาดไทย
- กระทรวงอุตสาหกรรม

กลุ่มเป้าหมายรอง ได้แก่ ประชาชนที่ขอรับบริการภาครัฐ

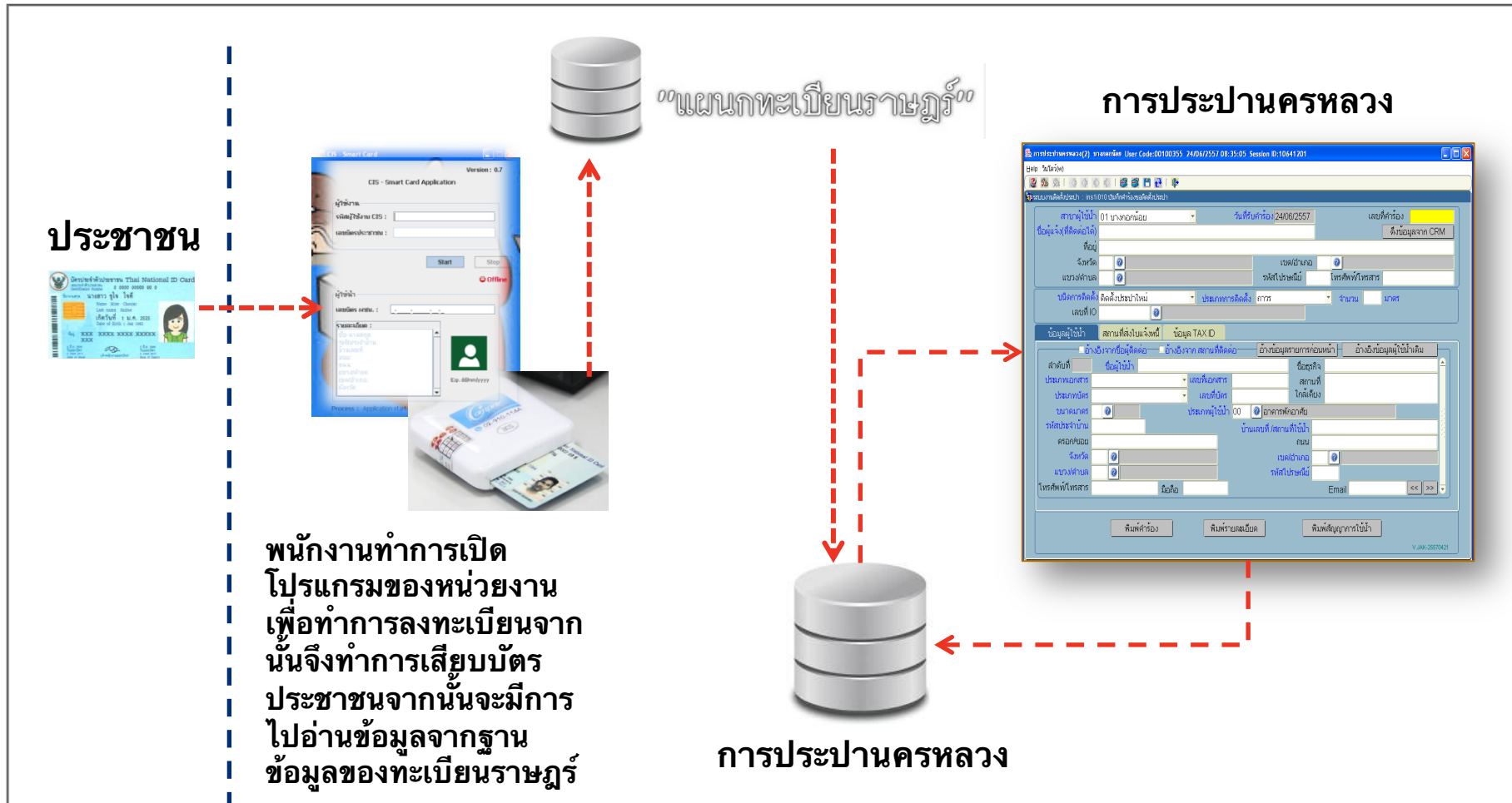
กลุ่มเป้าหมายที่มีส่วนเกี่ยวข้อง ได้แก่ สำนักงานพัฒนาระบบราชการ (กพร.) สำนักงานพัฒนาธุรกรรมทางอิเล็กทรอนิกส์ (องค์การมหาชน) และหน่วยงานที่กำกับนโยบายและมาตรการที่เกี่ยวข้อง

แนวทางในการดำเนินงาน

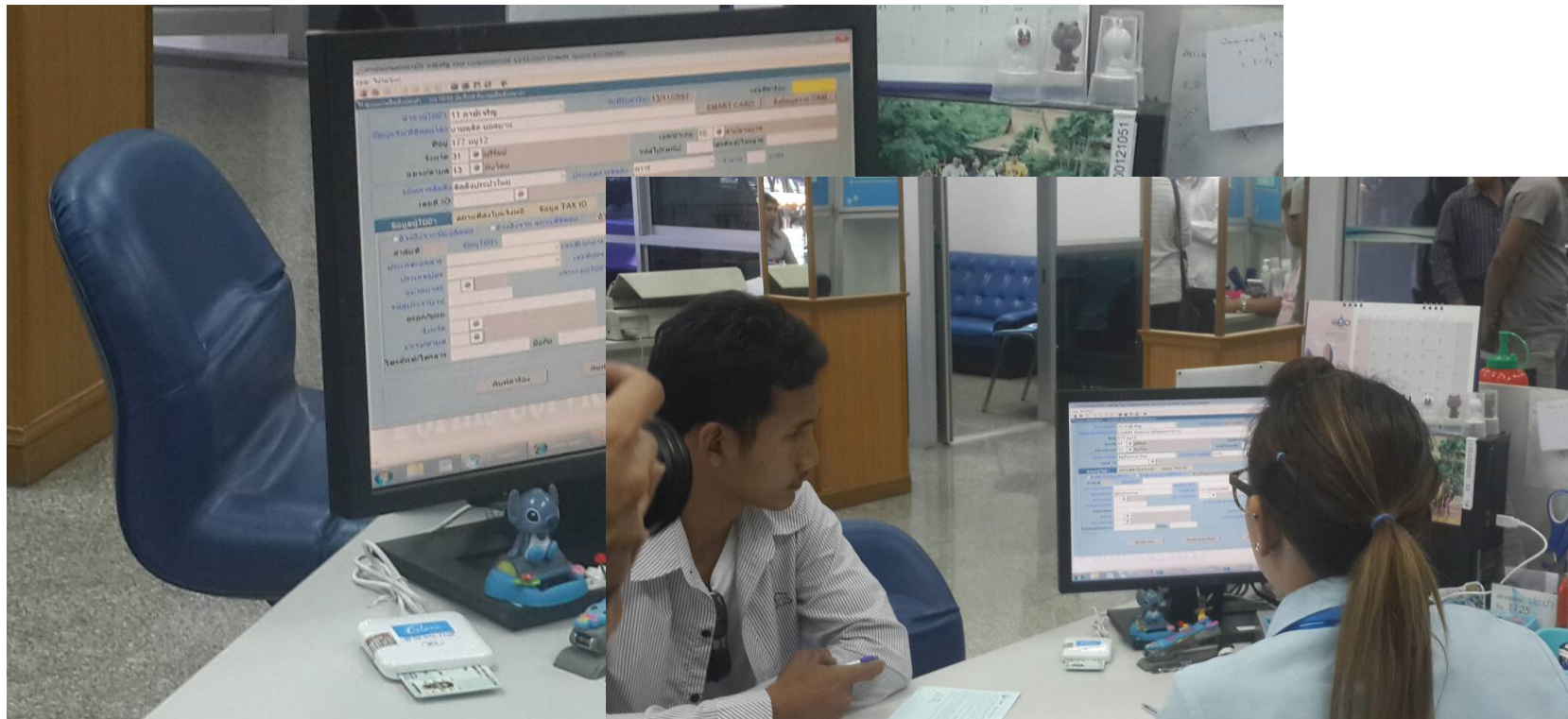
- . หน่วยงานผู้ให้บริการดำเนินการปรับขั้นตอน/ระบบบริการของราชการให้สามารถเชื่อมโยงข้อมูลกับฐานข้อมูลที่เกี่ยวข้องของหน่วยงานต้นทาง อาทิ ฐานข้อมูลทะเบียนราษฎร เพื่อลดการขอสำเนาเอกสารราชการจากผู้ขอรับบริการ โดยดำเนินการร่วมกับการปรับปรุงกฎหมาย และระเบียบต่างๆ ที่เกี่ยวข้อง
- . หน่วยงานผู้ให้บริการ เริ่มนำร่องการให้บริการแบบ Smart Service แก่ประชาชน ณ จุดให้บริการของแต่ละหน่วยงาน
- . หน่วยงานผู้ให้บริการและ สรอ. ร่วมกันพัฒนา Smart Service ให้ประชาชนสามารถเข้าถึงบริการบางเรื่องได้ด้วยตนเอง (Self Service) โดยนำร่องผ่านตู้เอนกประสงค์ (Kiosk) และพัฒนาต่อยอดผ่านช่องทางอื่นๆ ต่อไปในอนาคต

แนวทางในการดำเนินงาน

Process Flow ตัวอย่างของการประสานครหลวง



แนวทางในการดำเนินงาน



การให้บริการ
การประสานครหลวง

แนวทางในการดำเนินงาน



ตู้เอนกประสงค์ (Kiosk)
ณ G-Point Central World

เกณฑ์การคัดเลือกบริการ

- 1 หน่วยงานมีความพร้อมยินดีเข้าร่วมบริการ
- 2 มีระบบสารสนเทศ สำหรับสนับสนุนการดำเนินงาน
- 3 เป็นบริการประเภท ให้บริการแก่ประชาชน
- 4 บริการที่เข้าร่วมปัจจุบันมีการขอเรียกใช้สำเนาบัตรประชาชน และ/หรือ สำเนาทะเบียนบ้าน จากประชาชน

แนวทางการให้บริการแบบ Smart Service

1 อ่านข้อมูลจากหน้าบัตร และทำการเชื่อมโยงข้อมูลจากหน้าบัตรลงในระบบของหน่วยงานได้ (แบบ Offline)



CIS - Smart Card Application Version: 0.7

Offline

Process : Application start

การชำระเงินตรา(2) จำนวนเงิน User Code:00100355 24/06/2557 08:35:05 Session ID:10641201

ข้อมูลผู้ยื่นใบแจ้งหนี้

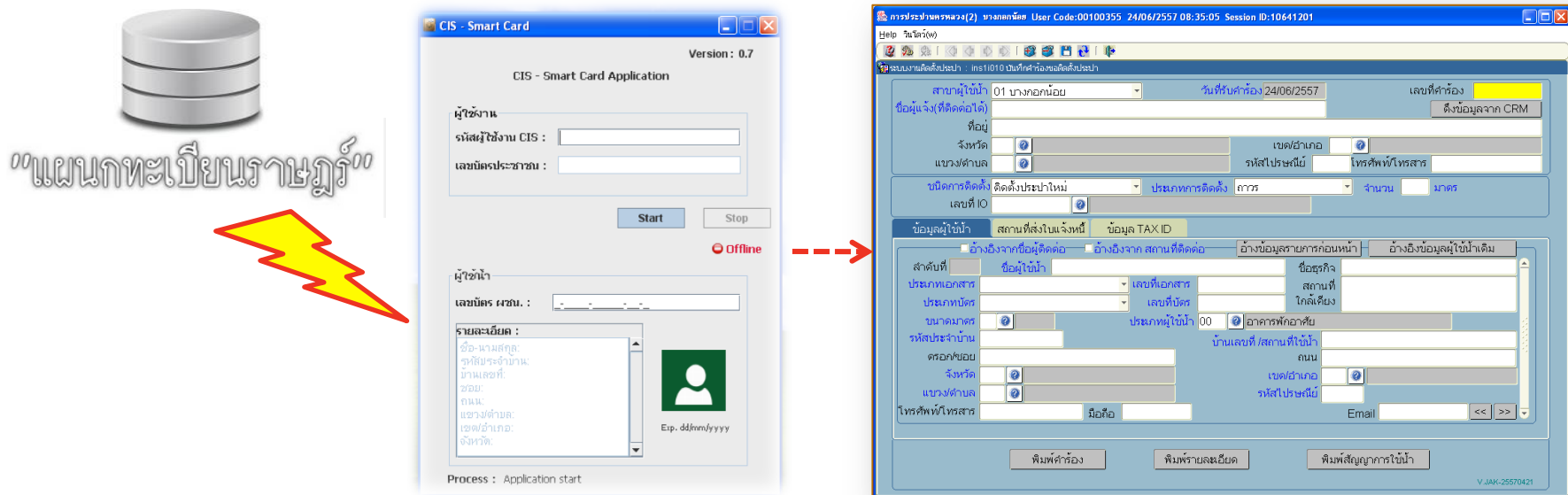
ข้อมูล TAX ID

ข้อมูลผู้ยื่นใบแจ้งหนี้



แนวทางการให้บริการแบบ Smart Service

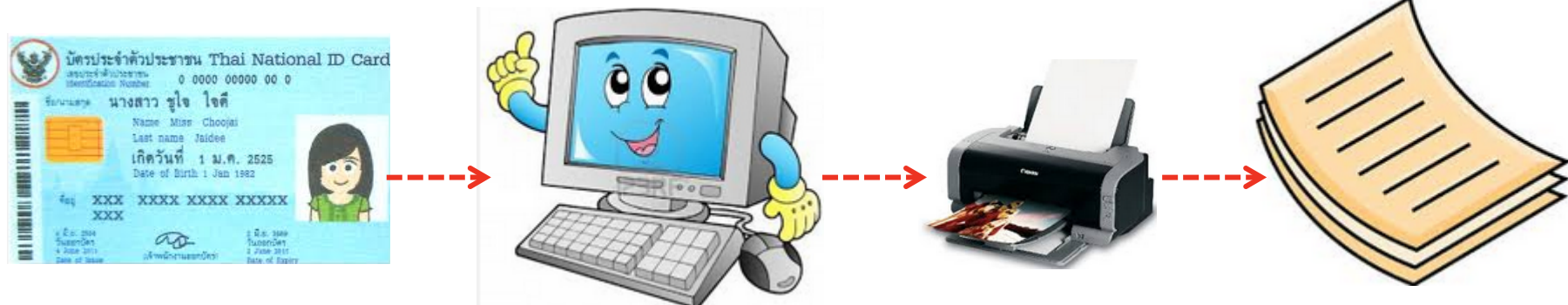
2 อ่านข้อมูลจากฐานข้อมูลทะเบียนราษฎร์ และทำการเชื่อมโยงข้อมูลจากหน้าบัตรลงในระบบของหน่วยงานได้ (แบบ Online)



แนวทางการให้บริการแบบ Smart Service

3 ติดตั้งโปรแกรมคัดลอกสำเนา โปรแกรมอ่านข้อมูลจากหน้าบัตร แล้วพิมพ์ให้ผู้ให้บริการลงนามเอกสาร

งบประมาณสนับสนุนให้หน่วยงาน



งบประมาณสนับสนุนให้หน่วยงาน

งบประมาณที่คาดว่าจะใช้ในการดำเนินงาน **ค่าพัฒนาปรับปรุงระบบสารสนเทศ หรือระบบบริการ (e-Service)** ของหน่วยงานให้รองรับการให้บริการโดยไม่ต้องใช้สำเนาเอกสารจากประชาชน เป็นค่าใช้จ่ายที่ สรอ. จัดสรรให้หน่วยงานเพื่อดำเนินการปรับปรุงระบบฯ ของหน่วยงาน โดยครอบคลุมถึงค่าใช้จ่ายต่างๆ ดังนี้

- ❑ สำรวจและเก็บข้อมูลระบบงานบริการเพื่อพัฒนาปรับปรุง
- ❑ วิเคราะห์และออกแบบการพัฒนาปรับปรุงระบบ
- ❑ พัฒนาปรับปรุงระบบ
- ❑ ทดสอบระบบ
- ❑ อบรมการใช้งานระบบ
- ❑ ติดตั้ง และโอนย้ายข้อมูลในระบบ (Data Migration)
- ❑ อุปกรณ์ต่างๆ ที่เกี่ยวข้อง อาทิ เครื่องคอมพิวเตอร์ เครื่องพริ้นเตอร์ (Printer) เครื่องสแกนเนอร์ (Scanner)
- ❑ ค่าใช้จ่ายอื่นๆ

งบประมาณสนับสนุนให้หน่วยงาน

โดยขนาดของระบบสารสนเทศประเมินจากเงื่อนไขต่างๆ ดังนี้

- จำนวนแบบฟอร์มหรือหน้าจอที่จำเป็นที่จะต้องมีการปรับเพื่อให้สามารถเชื่อมโยงข้อมูลได้
- ปริมาณ transaction ที่ประชาชนขอใช้บริการ
- ปริมาณข้อมูลในระบบ
- ปริมาณจุดติดตั้ง ในกรณีที่เป็น client server
- จำนวนเจ้าหน้าที่ ที่ต้องได้รับการฝึกอบรมเพิ่มเติม
- ความซับซ้อน, ความเชื่อมโยงกับระบบอื่นที่เกี่ยวข้อง

การดำเนินการ ก่อนการพัฒนาระบบ

1 จัดทำ TOR พัฒนาระบบ

2 จัดทำ MOU กับกรมการปกครอง
ขอสัญญาจ้าง กับกรมการปกครอง
ขอ PIN ที่สำนักงานเขตหรืออำเภอ

3 ลงทะเบียนใช้งาน API ผ่าน สรอ.
กรอกฟอร์มลงทะเบียนใช้งาน API สรอ.

download ตัวอย่างเอกสาร และแบบฟอร์ม ได้ที่ dev.egov.go.th
(สมัครสมาชิกก่อนเข้าใช้งาน)

หัวข้อการอบรม

ช่วงเช้า - การพัฒนาระบบแบบ **offline**

องค์ประกอบที่เกี่ยวข้อง

ตัวอย่างการพัฒนาด้วย C#.Net และ Java

Offline Demo การเรียกใช้ข้อมูล

ช่วงบ่าย - การพัฒนาระบบโดยใช้ **API (online)**

การดำเนินการ ก่อนการพัฒนาระบบ

สถาปัตยกรรม และการทำงาน

ชุด Consumer Key

การเรียกใช้ API ตัวอย่างการพัฒนา

ตรวจสอบ log การเรียกใช้

การทดสอบเรียก API จากข้อมูล Dummy

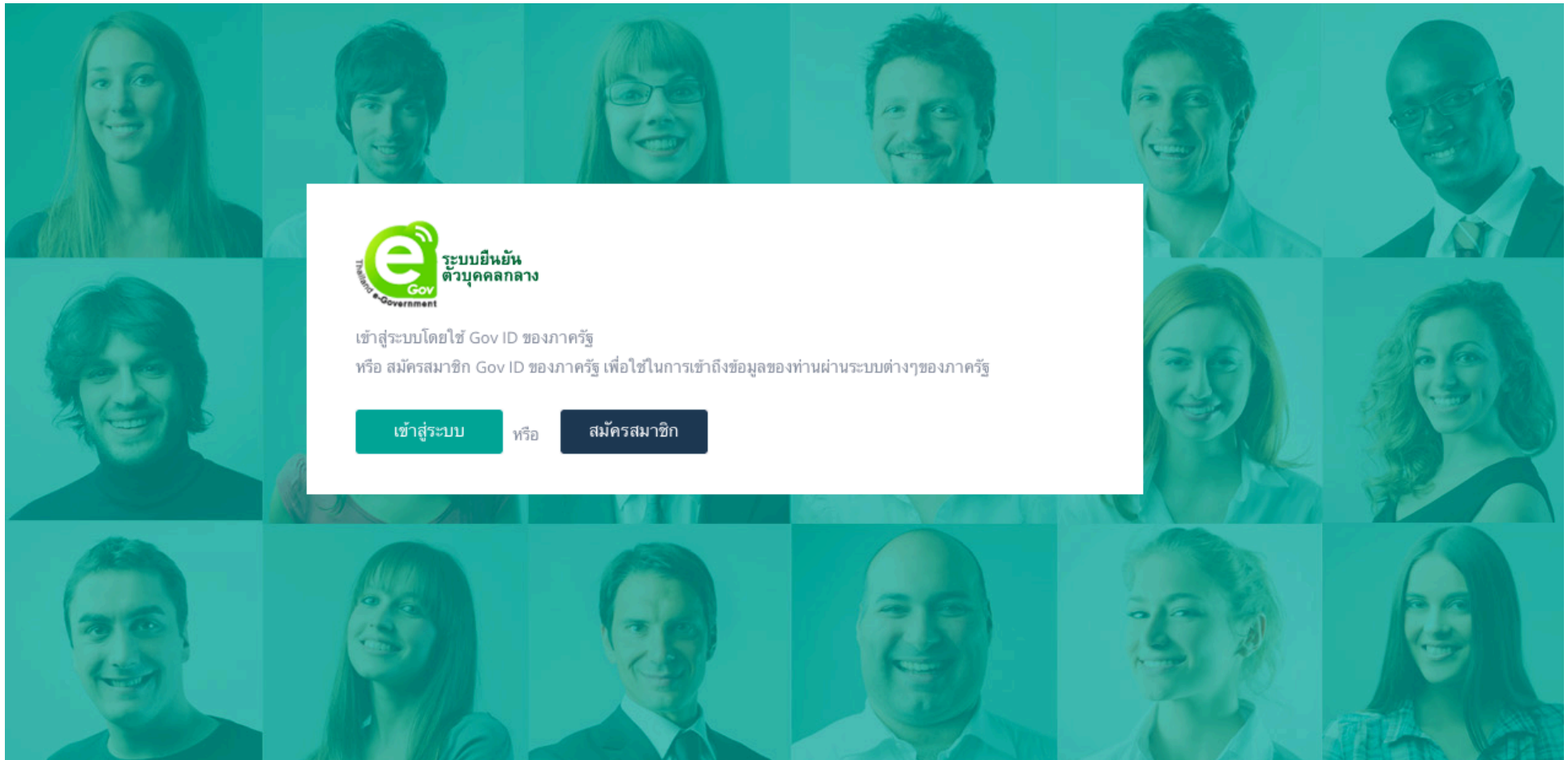
ขั้นตอนการใช้งาน และข้อควรระวังในการใช้งานข้อมูลบุคคล

Online Demo(แบบ Centralized Authentication และ Client Authentication)

การเรียกใช้ข้อมูล

การสมัครสมาชิก dev.egov.go.th

URL : dev.egov.go.th => egov.go.th



การสมัครสมาชิก dev.egov.go.th

หน้าหลัก



ระบบจัดการบริการจากบัตรประชาชนสมาร์ทการ์ด
Web Service Authorization and Management System

หน้าแรก ข้อมูลที่เกี่ยวข้อง API ในการพัฒนา ขอใช้งานเว็บเซอร์วิส



การพัฒนาระบบบริการ

การให้บริการ

ภาพรวมโครงการ

การสมัครสมาชิก sc.ega.or.th

URL : sc.ega.or.th

 Asis Unyapoth / Public Info Sign in

<

Project

Milestones

Issues 0

Merge Requests 0

Labels

Wiki

Home Pages Git Access

home Page History

ยินดีต้อนรับสู่ระบบ EGA Public Source Control ซึ่งจัดทำขึ้นเพื่อมีวัตถุประสงค์เพื่อให้เป็นช่องทางที่ ผู้พัฒนากายนอก สรอ. สามารถทำงานร่วมกับผู้พัฒนภายใน สรอ. ผู้ดูแลระบบ หรืออื่นๆ ได้อย่างมีประสิทธิภาพ โดยมีมาตรการในการรักษาความปลอดภัยของ source code ต่างๆ อย่างเหมาะสม

ความรู้เบื้องต้นเกี่ยวกับ Git

- Documentation ภาษาไทย
- คำแนะนำในการแตกกิ่ง (Git Branching)

การจัดการ (สรอ.)

- การสร้าง repo ใหม่

วิธีการใช้งาน

- วิธีการใช้งานบน Eclipse <http://www.vogella.com/tutorials/EclipseGit/article.html> (รายละเอียดเพิ่มเติม)

21

การสมัครสมาชิก sc.ega.or.th

Sign in /Sign up



GitLab Community Edition

Open source software to collaborate on code

Manage git repositories with fine grained access controls that keep your code secure. Perform code reviews and enhance collaboration with merge requests. Each project can also have an issue tracker and a wiki.

กรุณาส่งสมัครสมาชิกก่อนเข้าใช้งานนะครับ

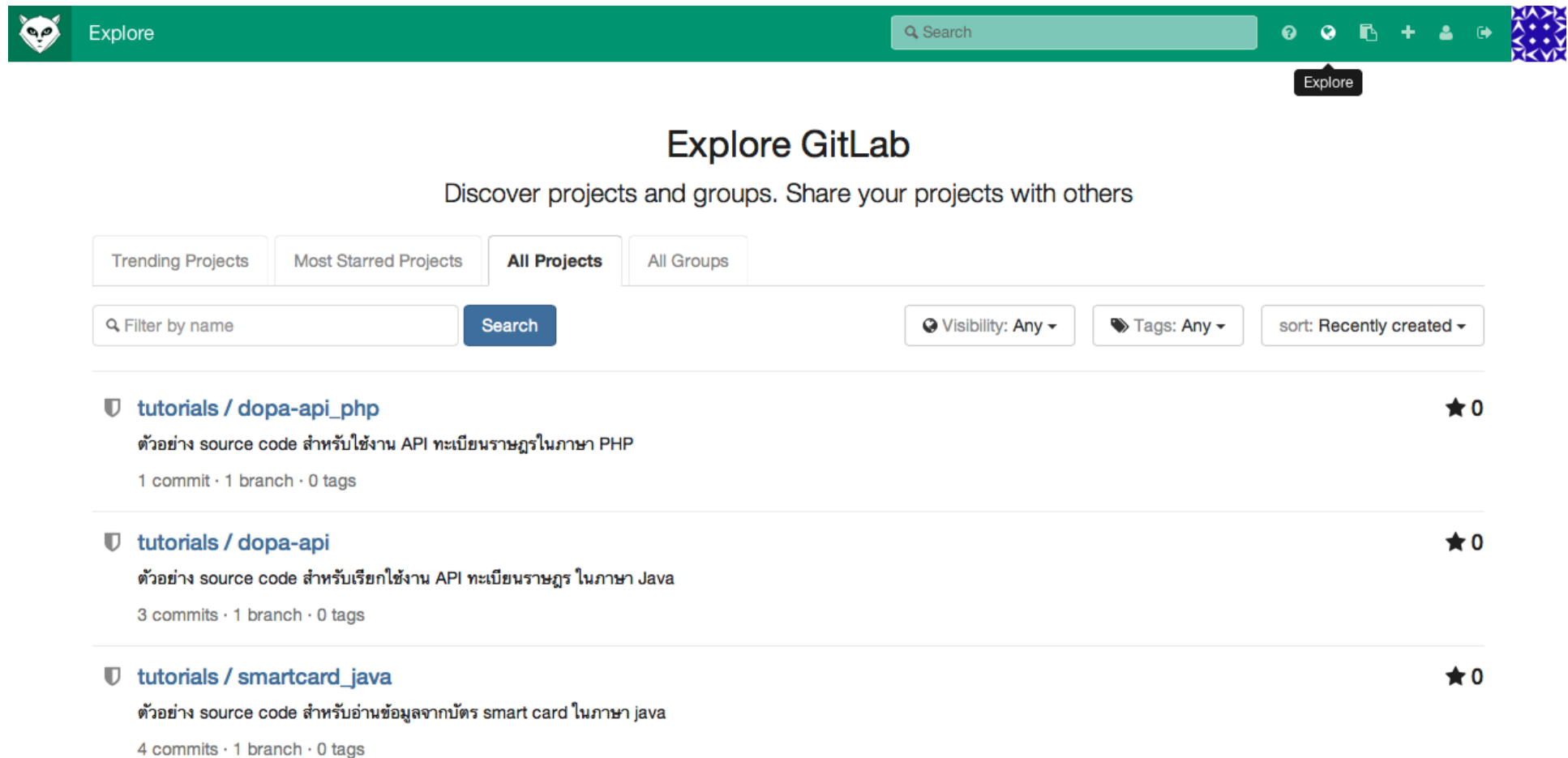
Existing user? Sign in

☐ Remember me[Forgot your password?](#)

New user? Create an account

การสมัครสมาชิก sc.ega.or.th

หลังจาก Login : Explore / All Project



The screenshot shows the 'Explore GitLab' page. At the top is a green navigation bar with the GitLab logo, the word 'Explore', a search bar, and several utility icons. Below the navigation bar, the page title 'Explore GitLab' is centered, followed by the subtitle 'Discover projects and groups. Share your projects with others'. A horizontal bar contains four tabs: 'Trending Projects', 'Most Starred Projects', 'All Projects' (which is selected), and 'All Groups'. Below this is a search section with a 'Filter by name' input, a 'Search' button, and three filters: 'Visibility: Any', 'Tags: Any', and 'sort: Recently created'. The main content area displays a list of three projects, each with a repository icon, the project name, a description, and star/commit/branch/tag counts.

Project Name	Description	Commits	Branches	Tags	Stars
tutorials / dopa-api_php	ตัวอย่าง source code สำหรับใช้งาน API ทะเบียนราษฎรในภาษา PHP	1 commit	1 branch	0 tags	0
tutorials / dopa-api	ตัวอย่าง source code สำหรับเรียกใช้งาน API ทะเบียนราษฎร ในภาษา Java	3 commits	1 branch	0 tags	0
tutorials / smartcard_java	ตัวอย่าง source code สำหรับอ่านข้อมูลจากบัตร smart card ในภาษา java	4 commits	1 branch	0 tags	0

การพัฒนาระบบ

การพัฒนาระบบแบบ **offline**

การพัฒนาระบบโดยใช้ **API (online)**

รูปแบบในการเรียกใช้ข้อมูล

แบบ Offline

- เป็นการเรียกใช้ข้อมูลที่เก็บอยู่ในบัตรประจำตัวประชาชน
- ข้อจำกัดคือข้อมูลไม่ update ล่าสุด

แบบ Online

- เป็นการเรียกใช้ข้อมูลจาก API กรมการปกครอง
- ข้อมูล update ล่าสุด
- มีชุดข้อมูลหลายชุด

Government API (Online)

Government API เป็นระบบที่ทำให้ระบบสารสนเทศ(Information System) หรือแอปพลิเคชัน(Applications) สามารถเข้าถึงและเชื่อมโยงข้อมูลต่างๆ ของหน่วยงานภาครัฐได้ โดยมีการควบคุมสิทธิ์ในการเข้าถึงข้อมูล และมีการรักษาความปลอดภัยของข้อมูลที่เหมาะสม

โดยเป็นเครื่องมือสำคัญสำหรับขับเคลื่อนนโยบายของรัฐมนตรีว่าการกระทรวงเทคโนโลยีสารสนเทศและการสื่อสารที่ให้ "เร่งรัดการบูรณาการและแลกเปลี่ยนข้อมูลอิเล็กทรอนิกส์ระหว่างหน่วยงานภาครัฐ เพื่อให้การดำเนินการของรัฐบาลอิเล็กทรอนิกส์สัมฤทธิ์ผลด้วยความรวดเร็ว"

การพัฒนาแบบ Offline

องค์ประกอบที่เกี่ยวข้อง

ตัวอย่างการพัฒนา C#.Net

ตัวอย่างการพัฒนา Java

การพัฒนาแบบ Offline

C#.Net

การพัฒนาระบบแบบ Offline C#

Library ที่จำเป็นต่อการพัฒนาส่วนของการอ่านข้อมูลจากบัตร ประกอบด้วย ไฟล์ dll 2 ไฟล์

GemCard.dll

เป็น DLL ทัวไปสำหรับเชื่อมต่อระหว่างโปรแกรมกับเครื่องอ่านบัตร (Card Reader) เพื่ออ่านข้อมูลจากบัตรสมาร์ทประจำตัวประชาชนแบบสมาร์ทการ์ด ที่เป็นไปตามมาตรฐาน PC/SC API

ThaiSmartCardPlayer.dll

เป็น DLL ที่ สรอ.พัฒนาขึ้นสำหรับการเข้าถึงข้อมูลจากบัตรประจำตัวประชาชนแบบสมาร์ทการ์ด (จะต้อง update เมื่อมีการเปลี่ยนบัตรรุ่นใหม่ที่มีการเปลี่ยนตำแหน่งของข้อมูล แต่บัตรรุ่นหลังๆที่พบไม่มีการเปลี่ยนตำแหน่งของข้อมูลจึงไม่ต้อง update dll)

การพัฒนาแบบ Offline C#

Component ที่ต้องติดตั้งก่อนการพัฒนา หากต้องการเรียก CID จากบัตรประจำตัวประชาชน ประกอบด้วย (ต้อง Online)

License Manager (LM)

เป็นชุดโปรแกรม License ที่ออกโดยกรมการปกครอง และต้องมีการ update program ทุกปี ช่วงเดือนมกราคม

WinSock

เครื่องมือของโปรแกรมที่ สรอ.ปรับแต่ง ที่จะถูกใช้ในการในการส่ง และ รับ ข้อมูลผ่านทางหมายเลขพอร์ทของ TCP/IP ที่กำหนด

หมายเหตุ

- DLL และ Component ต่างๆ download ได้ที่ dev.egov.go.th(สมัครก่อนเข้าใช้งาน)
- หากไม่ต้องการดึงข้อมูล CID ไม่ต้องใช้ LM และ Winsock

การพัฒนาระบบแบบ Offline C#

การกำหนดค่าเริ่มแรก

กำหนดตัวแปรสำหรับคลาส ThaiSmartCardReader

```
private ThaiSmartCardReader _reader;
```

และวิธีการอ่านข้อมูลมี 2 แบบ คือ

การอ่านบัตรแบบอัตโนมัติ (Automatic) – เครื่องจะทำการอ่านข้อมูลหลังเสียบบัตรทันทีโดยอัตโนมัติ และควรกำหนดการหน่วงเวลาก่อนทำการอ่านข้อมูลจากบัตรฯ อย่างน้อย 1,000 - 3,000 ms เพื่อป้องกันการอ่านบัตรผิดพลาด

การอ่านบัตรแบบสั่งให้อ่าน (Manual) – เครื่องจะทำการอ่านข้อมูลตามคำสั่งที่ผู้พัฒนากำหนด

การพัฒนาระบบแบบ Offline C#

The screenshot shows a Windows application titled "SmartCard Sample Offline". The interface is divided into two main sections: a form for card data and a log window.

Thai National ID Card Section:

- Card Info:** Fields for Card No. (01, 0003, and a masked 15-digit number), Delay (ms) (3000), and a "Read Card" button.
- PID:** A masked 15-digit field.
- Name (TH):** Fields for Last Name (นาย), First Name (masked), and Middle Name (masked).
- Name (EN):** Fields for Title (Mr.), Last Name (masked), and First Name (masked).
- Birth Date:** Fields for Birth Date (masked) and Sex (ชาย).
- Address:** A large text area for the address.
- Issue Date:** Fields for Issue Date (5/9/2013) and Validity Date (3/30/2022).
- Issuer:** Fields for Issuer Name (ท้องถิ่นเขตราชเทวี) and Issuer Code (กรุงเทพมหานคร).
- Issuer Code:** A masked 15-digit field.
- Read Options:** Radio buttons for "Read Text" and "Read Field". Checkboxes for Name, Address, Issue, Version, Image, ImageNumber, CardNumber, Issuer, and IssuerNumber.

Log Section:

- Buttons for "Clear Log" and "Copy Log".
- Checkboxes for "Reader Connected", "Card Inserted", and "Card Supported", all of which are checked.
- A text area displaying a log of "TransmitCard: GetImage -> (Ascii)" commands and their corresponding data, including Thai and English names, birth date, sex, and address.

การพัฒนาแบบ Offline C#

ตัวอย่าง การอ่านข้อมูลบัตรฯ แบบอัตโนมัติ (Automatic)

```
_reader = ThaiSmartCardReader.GetInstance(ReadCard, ReadCard);  
_reader.TimeReadDelay = 1000;
```

ตัวอย่าง การอ่านข้อมูลบัตรฯ แบบสั่งให้อ่าน (Manual)

```
_reader = ThaiSmartCardReader.GetInstance();
```

กำหนดค่าก่อนปิดโปรแกรมเพื่อป้องกันไม่ให้เกิด Process ค้าง

```
_reader = ThaiSmartCardReader.GetInstance();  
_reader.ShutDown();
```

การพัฒนาระบบแบบ Offline C#

การอ่านข้อมูลจากบัตร
เลือกรูปแบบการอ่านบัตรได้ 2 แบบ คือ

- 1 การอ่านข้อมูลแบบข้อความทั้งหมด
- 2 การอ่านข้อมูลแบบ Field

การพัฒนาระบบแบบ Offline C#

การอ่านข้อมูลแบบข้อความทั้งหมด

โปรแกรมจะทำการอ่านแสดงข้อมูลในบัตรที่อยู่ในรูปแบบข้อความ(String)ทั้งหมด โดยที่การอ่านแบบนี้จะไม่สามารถอ่านข้อมูลรูปภาพได้

ตัวอย่าง การอ่านข้อมูลบัตรฯ แบบข้อความทั้งหมด

```
string text = _reader.ReadAllText();  
ThaiSmartCardInfo info = _reader.ConvertTextToInfo(text);
```

การพัฒนาระบบแบบ Offline C#

การอ่านข้อมูลแบบแบบฟิลด์

โปรแกรมจะทำการอ่านข้อมูลตามที่คุณใช้งานเลือกไว้ โดยไม่จำเป็นต้องอ่านข้อมูลในบัตรทั้งหมด การอ่านข้อมูลลักษณะนี้สามารถอ่านข้อมูลที่เป็นรูปภาพได้

ตัวอย่าง การอ่านข้อมูลบัตรฯ แบบฟิลด์ข้อมูล

```
_reader.IsReadAll = false;
_reader.IsReadName = true;
_reader.IsReadAddress = true;
_reader.IsReadIssue = true;
_reader.IsReadVersion = true;
_reader.IsReadImageNumber = true;
_reader.IsReadCardNumber = true;
_reader.IsReadIssuer = true;
_reader.IsReadIssuerNumber = true;
_reader.IsReadImage = true;
_reader.ReadCard();
ThaiSmartCardInfo info = _reader.CardInfo;
```

การพัฒนาระบบแบบ Offline C#

หลังจากที่โปรแกรมทำการอ่านข้อมูลเรียบร้อยแล้ว จะได้ข้อมูลออกมาในลักษณะ info.XXXXX

```
string CitizenID = info.CitizenID;
string NameTH_Prefix = info.NameTH_Prefix;
string NameTH_FirstName = info.NameTH_FirstName;
string NameTH_SurName = info.NameTH_SurName;
string NameEN_Prefix = info.NameEN_Prefix;
string NameEN_FirstName = info.NameEN_FirstName;
string NameEN_SurName = info.NameEN_SurName;
string Sex = info.Sex;
string BirthDateTH = info.BirthDateTH;
string Address = info.Address;
string IssueDateTH = info.IssueDateTH;
string ExpireDateTH = info.ExpireDateTH;
string CardType = info.CardType;
string Version = info.Version;
string ImageNumber = info.ImageNumber;
string CardNumber = info.CardNumber;
string RequestNumber = info.RequestNumber;
string IssuerPlace = info.IssuerPlace;
string IssuerAgency = info.IssuerAgency;
string IssuerID = info.IssuerID;
Image personImg = info.Image;
```

การพัฒนาระบบแบบ Offline C#

Field	Data Type	Description
<u>CitizenID</u>	String	หมายเลขบัตรประจำตัวประชาชน 13 หลัก
<u>NameTH_Prefix</u>	String	คำนำหน้าชื่อ (ภาษาไทย)
<u>NameTH_FirstName</u>	String	ชื่อจริง (ภาษาไทย)
<u>NameTH_SurName</u>	String	นามสกุล (ภาษาไทย)
<u>NameEN_Prefix</u>	String	คำนำหน้าชื่อ (ภาษาอังกฤษ)
<u>NameEN_FirstName</u>	String	ชื่อจริง (ภาษาอังกฤษ)
<u>NameEN_SurName</u>	String	นามสกุล (ภาษาอังกฤษ)
<u>SexID</u>	int	รหัสเพศ ได้แก่ • 1 คือ ชาย • 2 คือ หญิง
<u>Sex</u>	String	เพศ (ชาย หรือ หญิง)
<u>BirthDate</u>	<u>DateTime</u> e	วันเกิดเป็นข้อมูลชนิด <u>DateTime</u>
<u>BirthDateTH</u>	String	วัน/เดือน/ปี พ.ศ. เกิด เช่น 01/12/2530
<u>Address</u>	String	ที่อยู่ตามสำเนาทะเบียนบ้าน
<u>Address_No</u>	String	เลขที่บ้าน
<u>Address_Moo</u>	String	เลขหมู่
<u>Address_MooPrefix</u>	String	คำว่า "หมู่"
<u>Address_Alley</u>	String	ชื่อหมู่บ้าน
<u>Address_AlleyPrefix</u>	String	คำว่า "หมู่บ้าน"
<u>Address_Soi</u>	String	ชื่อซอย

การพัฒนาระบบแบบ Offline C#

การตรวจสอบการอ่านข้อมูล

ผู้พัฒนาสามารถกำหนดให้โปรแกรมทำการตรวจสอบสถานะต่าง ๆ ได้แก่

- 1 สถานะการเชื่อมต่อกับเครื่องอ่านบัตร
- 2 สถานะการเสียบบัตรอยู่หรือไม่
- 3 สถานะว่าบัตรดังกล่าวเป็นบัตรประจำตัวประชาชนแบบสมาร์ทการ์ดหรือไม่

โดยโปรแกรมจะทำการตรวจสอบด้วยการเขียนโปรแกรมดังนี้

การพัฒนาแบบ Offline C#

การตรวจสอบการอ่านข้อมูล

```
bool IsCardReaderConnected = _reader.IsCardReaderConnected;  
// ตรวจสอบว่าเชื่อมต่อไปยัง Card Reader หรือไม่  
  
bool IsCardInserted = _reader.IsCardInserted;  
// ตรวจสอบว่ามีบัตรเสียบอยู่หรือไม่  
  
bool IsCardSupported = _reader.IsCardSupported;  
// ตรวจสอบว่าบัตรดังกล่าวเป็นบัตรประจำตัวประชาชนแบบสมาร์ทการ์ดหรือไม่
```

การพัฒนาแบบ Offline C#

ตัวอย่าง **source code – Offline**

C#.Net

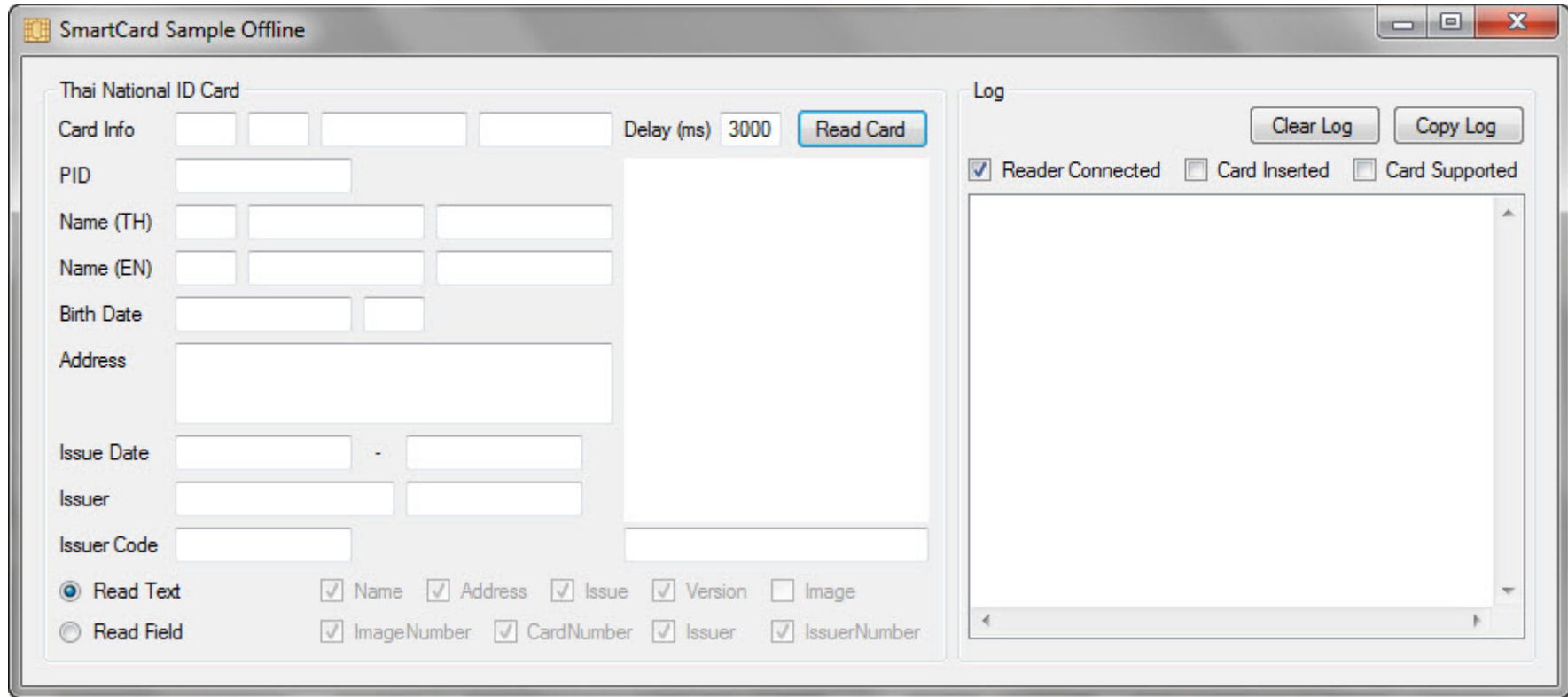
http://sc.ega.or.th/tutorials/smartservice_csharp/tree/master

(สมัครก่อนเข้าใช้งาน)

ตัวอย่าง

Offline Application C#

ดึงข้อมูลจากบัตร - Offline C#



The image shows a Windows application window titled "SmartCard Sample Offline". The window is divided into two main sections: "Thai National ID Card" on the left and "Log" on the right.

Thai National ID Card Section:

- Card Info:** Four text input fields for card information.
- Delay (ms):** A text input field with the value "3000".
- Read Card:** A blue button to initiate the card reading process.
- PID:** A text input field.
- Name (TH):** Three text input fields for the Thai name.
- Name (EN):** Three text input fields for the English name.
- Birth Date:** Two text input fields for the birth date.
- Address:** A large text input field for the address.
- Issue Date:** Two text input fields separated by a hyphen for the issue date.
- Issuer:** Two text input fields for the issuer name.
- Issuer Code:** A text input field.
- Read Options:** Two radio buttons: "Read Text" (selected) and "Read Field".
- Field Selection:** A grid of checkboxes for selecting data fields to read:
 - Row 1: ☒ Name, ☒ Address, ☒ Issue, ☒ Version, ☐ Image
 - Row 2: ☒ ImageNumber, ☒ CardNumber, ☒ Issuer, ☒ IssuerNumber

Log Section:

- Buttons:** "Clear Log" and "Copy Log" buttons.
- Status Indicators:** Three checkboxes: ☒ Reader Connected, ☐ Card Inserted, and ☐ Card Supported.
- Log Area:** A large text area for displaying the log output.

ดึงข้อมูลจากบัตร - Offline C#

The screenshot displays the 'SmartCard Sample Offline' application window. The left pane is titled 'Thai National ID Card' and contains various input fields for card information, a 'Read Card' button, and a set of checkboxes for data selection. The right pane, titled 'Log', shows a sequence of events including reader connection, card insertion, and data retrieval.

Thai National ID Card

Card Info: 01 0003 [redacted] 00001001 Delay (ms) 3000 **Read Card**

PID: [redacted]

Name (TH): นาย [redacted] [redacted]

Name (EN): Mr. [redacted] [redacted]

Birth Date: [redacted] ชาย

Address: [redacted] หมู่ที่ ๕ ตำบล [redacted] อำเภอ [redacted] จังหวัด [redacted]

Issue Date: 5/9/2013 - 3/30/2022

Issuer: กองทะเบียนราษฎร กรุงเทพมหานคร

Issuer Code: [redacted] 10070100001001

☒ Read Text ☒ Name ☒ Address ☒ Issue ☒ Version ☐ Image
☐ Read Field ☒ ImageNumber ☒ CardNumber ☒ Issuer ☒ IssuerNumber

Log **Clear Log** **Copy Log**

☒ Reader Connected ☒ Card Inserted ☒ Card Supported

```
SetupReaderList: Reader -> ACS CCID USB Reader 0
ConnectICard: Complete
ConnectICard: ATR -> 3B781800005448204E49442037
TransmitICard: Get Text -> (Ascii) 0000100100001001นาย#ส
TransmitICard: Get Text -> (Ascii) ขตราชเทวี/กรุงเทพมหานคร
TransmitICard: Get Text -> (Ascii) 00/0#หมู่ที่ ๕####ตำบล
ReadAllText: Complete
```

ดึงข้อมูลจากบัตร - Offline C#

The image shows a software application titled "SmartCard Sample Offline". It is designed to interact with a Thai National ID Card. The interface is divided into two main sections: a data entry/form section on the left and a log section on the right.

Thai National ID Card Section:

- Card Info:** Fields for Card No. (01, 0003, and a masked 15-digit number), Delay (ms) (3000), and a "Read Card" button.
- PID:** A masked 15-digit field.
- Name (TH):** Fields for Last Name (นาย), First Name (masked), and Middle Name (masked).
- Name (EN):** Fields for Title (Mr.), Last Name (masked), and First Name (masked).
- Birth Date:** Fields for Year (masked), Month (masked), and Day (masked).
- Address:** A large text area for the full address.
- Issue Date:** Fields for Issue Date (5/9/2013) and Expiry Date (3/30/2022).
- Issuer:** Fields for Issuer Name (ท้องถิ่นเขตราชเทวี) and Issuer Code (กรุงเทพมหานคร).
- Issuer Code:** A masked 15-digit field.
- Read Options:** Two radio buttons: "Read Text" (selected) and "Read Field". Below them are checkboxes for: Name, Address, Issue, Version, Image, ImageNumber, CardNumber, Issuer, and IssuerNumber.

Log Section:

- Buttons: "Clear Log" and "Copy Log".
- Checkboxes: "Reader Connected" (checked), "Card Inserted" (checked), and "Card Supported" (checked).
- Log Text: A list of log entries, each starting with "TransmitICard: GetImage -> (Ascii)". The entries contain various data points, including card numbers, names, and addresses, some in Thai and some in English. The last entry is highlighted in blue.

ดึงข้อมูลจากบัตร - Offline C#

ตัวอย่าง โปรแกรมคัดลอกสำเนา

Form1

Read Card Print

แบบรับรองรายการบุคคลจากฐานข้อมูลการทะเบียนของสำนักทะเบียนกลาง

ก. รายการบัตรประจำตัว

ชื่อ นาย [REDACTED]

ชื่อ(อังกฤษ) Mr. [REDACTED]

เลขประจำตัวประชาชน [REDACTED]

เพศ [REDACTED]

เกิดเมื่อ [REDACTED]

[REDACTED]

ที่อยู่ [REDACTED]

หน่วยงานที่ออกบัตร ห้องถิ่นเขตราชเทวี

จังหวัด กรุงเทพมหานคร

เลขรหัสการทำบัตร 3720 [REDACTED]

CID 21682 [REDACTED]

วันออกบัตร 9 พฤษภาคม 2556

วันหมดอายุ 30 มีนาคม 2565

การพัฒนาแบบ Offline Java

Java

การพัฒนาแบบ Offline Java

ติดตั้ง Java Library

ega-common-smartcard.jar Library สรอ. พัฒนาขึ้นมาเพื่ออ่านบัตรประจำตัวประชาชน download ได้จาก dev.egov.go.th

และติดตั้ง External Library ที่จำเป็นต่อการเรียกใช้งาน Library จากโครงการ Apache Foundation จำนวน 2 Library ด้วยกัน คือ

Apache Commons Codec สามารถ download ได้จากเว็บไซต์ https://commons.apache.org/proper/commons-codec/download_codec.cgi

Apache Commons Lang สามารถ download ได้จากเว็บไซต์ https://commons.apache.org/proper/commons-lang/download_lang.cgi

การพัฒนาระบบแบบ Offline Java

ในการอ่านข้อมูลจากบัตรประจำตัวประชาชนแบบสมาร์ทการ์ดด้วย Java Library นั้น จะรองรับการอ่านข้อมูลแบบ Manual เท่านั้น ไม่รองรับการอ่านข้อมูลแบบ Automatic โดยผู้พัฒนาสามารถเขียน code ในการอ่านข้อมูลได้ดังต่อไปนี้

```
DOPAProfileBean profile = ThaiSmartCard.readDOPAProfile(true);
```

โดยในการเรียก method เพื่ออ่านข้อมูลนั้น จะต้องทำการส่ง boolean เข้าไปใน method ด้วย 1 ตัวแปร โดยค่าของ boolean ดังกล่าว มีความหมายดังนี้

true เพื่อกำหนดให้อ่านข้อมูลรูปภาพออกมาด้วย

false เพื่อกำหนดให้ไม่ทำการอ่านข้อมูลรูปภาพออกมาด้วย

เมื่ออ่านข้อมูลได้แล้ว method จะคืนค่ากลับมาให้เป็น DOPAProfileBean เพื่อนำไปใช้งานต่อไป

การพัฒนาระบบแบบ Offline Java

การนำไปใช้งาน

หลังจากที่ได้ DOPAProfileBean มาแล้ว เราสามารถเรียกใช้งานข้อมูลที่อ่านออกมาได้จากบัตรประจำตัวประชาชนแบบสแกนการ์ดผ่านทาง method ดังต่อไปนี้

หมายเหตุ

-ใช้กับ JDK7 ขึ้นไป

การพัฒนาระบบแบบ Offline Java

Type	Method	Description
String	getCitizenID()	หมายเลขประจำตัวประชาชน 13 หลัก
String	getNameEN_Prefix()	คำนำหน้าชื่อ (ภาษาอังกฤษ)
String	getNameEN_FirstName()	ชื่อจริง (ภาษาอังกฤษ)
String	getNameEN_MiddleNam()	ชื่อกลาง (ภาษาอังกฤษ)
String	getNameEN_SurName()	นามสกุล (ภาษาอังกฤษ)
String	getNameTH_Prefix()	คำนำหน้าชื่อ (ภาษาไทย)
String	getNameTH_FirstName()	ชื่อจริง (ภาษาไทย)
String	getNameTH_MiddleName()	ชื่อกลาง (ภาษาไทย)
String	getNameTH_SurName()	นามสกุล (ภาษาไทย)

การพัฒนาระบบแบบ Offline Java

Type	Method	Description
Int	getSex()	เพศ ได้แก่ • 1 คือ ชาย • 2 คือ หญิง
Date	getBirthDate()	วัน/ เดือน/ ปีเกิด
Int	getAge()	อายุ
String	getAddress_No()	บ้านเลขที่
String	getAddress_Moo()	หมู่
String	getAddress_Alley()	ตรอก
String	getAddress_Soi()	ซอย
String	getAddress_Road()	ถนน
String	getAddress_Tumbol()	ตำบล
String	getAddress_Amphur()	อำเภอ
String	getAddress_Province()	จังหวัด

การพัฒนาระบบแบบ Offline Java

Date	getIssueDate()	วันที่ทำบัตรประจำตัวประชาชน
Date	getExpireDate()	วันที่หมดอายุของบัตรประจำตัวประชาชน
String	getIssuerID()	รหัสสำนักงานที่ออกบัตรประจำตัวประชาชน
String	getIssuerPlace()	สถานที่ที่ทำการออกบัตรประจำตัวประชาชน
String	getIssuerAgency()	ชื่อของสำนักงานที่ออกบัตรประจำตัวประชาชน
Int	getVersion()	หมายเลขรุ่นของบัตรสมาร์ทการ์ด
String	getCardType()	ประเภทของบัตรสมาร์ทการ์ด
String	getCardNumber()	หมายเลขบัตรสมาร์ทการ์ด
String	getRequestNumber()	หมายเลขคำร้องการขอมีบัตรประจำตัวประชาชน
String	getImageNumber()	หมายเลขของภาพถ่ายที่ปรากฏอยู่บนบัตรประจำตัวประชาชน
byte[]	getImage()	ข้อมูลรูปภาพที่ปรากฏอยู่บนบัตรประจำตัวประชาชน

พัฒนาระบบแบบ Offline Java

ตัวอย่าง **source code** - **Offline**การ

Java

http://sc.ega.or.th/tutorials/smartcard_java/tree/master

(สมัครก่อนเข้าใช้งาน)

ตัวอย่าง

Offline Application Java

ดึงข้อมูลจากบัตร - Offline Java

Thai Smart Card Viewer

☐ Image

Citizen ID:

English Name

Prefix: First Name: Middle Name: Sur Name:

Thai Name

Prefix: First Name: Middle Name: Sur Name:

Sex: Birth Date: Age:

No.: Moo: Alley:

Soi: Road:

Tumbol:

Amphur:

Province:

Issue Date: Expire Date:

Card Number:

Request Number:

Issuer ID:

Issuer Place:

Issuer Agency:

no image

Electronic Government Agency (Public Organization)



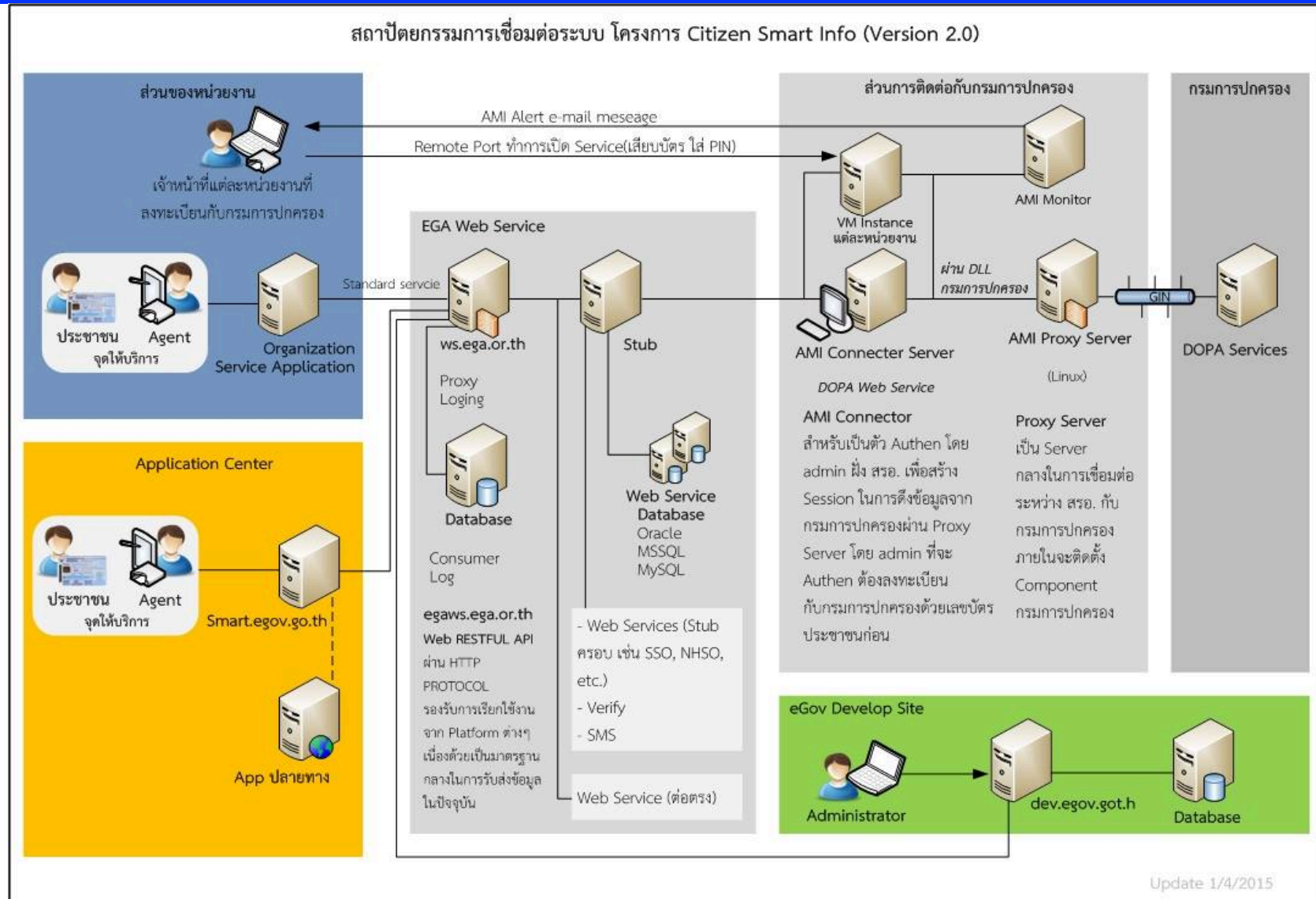
การพัฒนาระบบแบบ Online

การดำเนินการ ก่อนการพัฒนาระบบ
สถาปัตยกรรม และการทำงาน
ชุด Consumer Key
การเรียกใช้ API ตัวอย่างการพัฒนา C#, PHP, Java
ตรวจสอบ log การเรียกใช้
การทดสอบเรียก API จากข้อมูล Dummy
ขั้นตอนการใช้งาน และข้อควรระวังในการใช้งานข้อมูลบุคคล

การทำงานของระบบ

สถาปัตยกรรม และการทำงานของ
ws.ega.or.th

สถาปัตยกรรม Centralized Authen



สถาปัตยกรรม Centralized Authen

EGA Web Service – Web RESTFUL API ผ่าน HTTP PROTOCOL รองรับการทำงานจาก Platform ต่าง ๆ เนื่องจากเป็นมาตรฐานกลางในการรับ-ส่งข้อมูลในปัจจุบัน

AMI: Server Socket – สำหรับเป็นตัว Authenticate โดย Admin ฝั่ง สรอ. เพื่อสร้าง Session ในการดึงข้อมูลจากกรมการปกครองผ่าน Proxy Server โดย Admin ที่จะ Authenticate ต้องลงทะเบียนกับกรมการปกครองด้วยเลขบัตรประชาชนก่อน

AMI: Proxy Server – เป็น Server กลางในการเชื่อมต่อระหว่าง สรอ. กับ กรมการปกครอง ภายในจะติดตั้ง Component กรมการปกครอง

สถาปัตยกรรม Centralized Authenticate

กรมการปกครอง – สรอ. จะทำการเชื่อมต่อใช้ Service กับ Server ผ่าน DLL สำนักบริหารการทะเบียน กรมการปกครอง

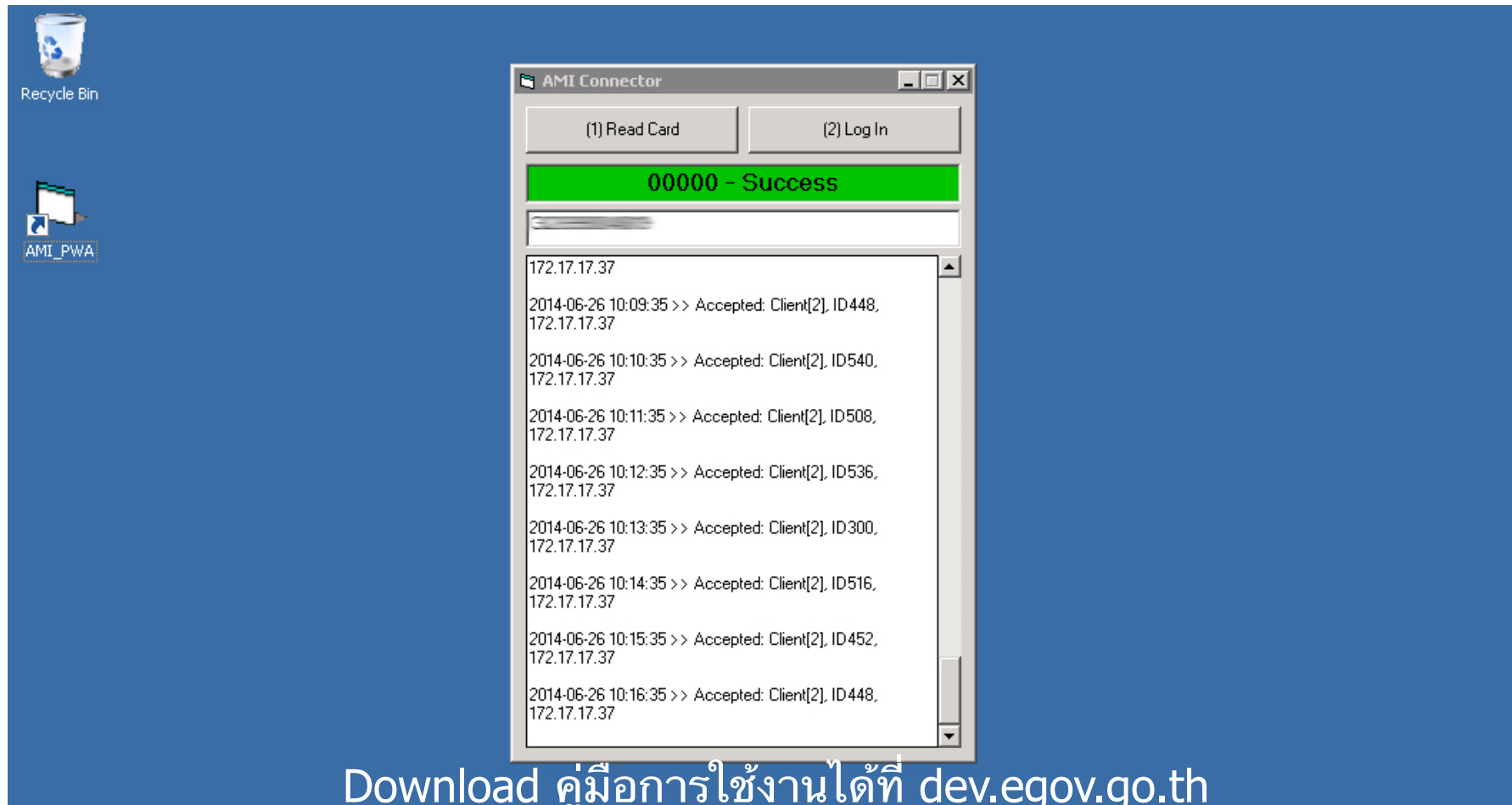
ส่วนของหน่วยงาน – เป็นการใช้ Service กับ Application ของแต่ละหน่วยงานพัฒนาขึ้นมา

Application Center– ระบบข้อมูลส่วนบุคคลโดยใช้บัตรประชาชนสมาร์ทการ์ด สำหรับประชาชน เป็น Application ที่ สรอ.พัฒนาขึ้นมาให้บริการ(เปรียบได้กับ Application ของหน่วยงาน)

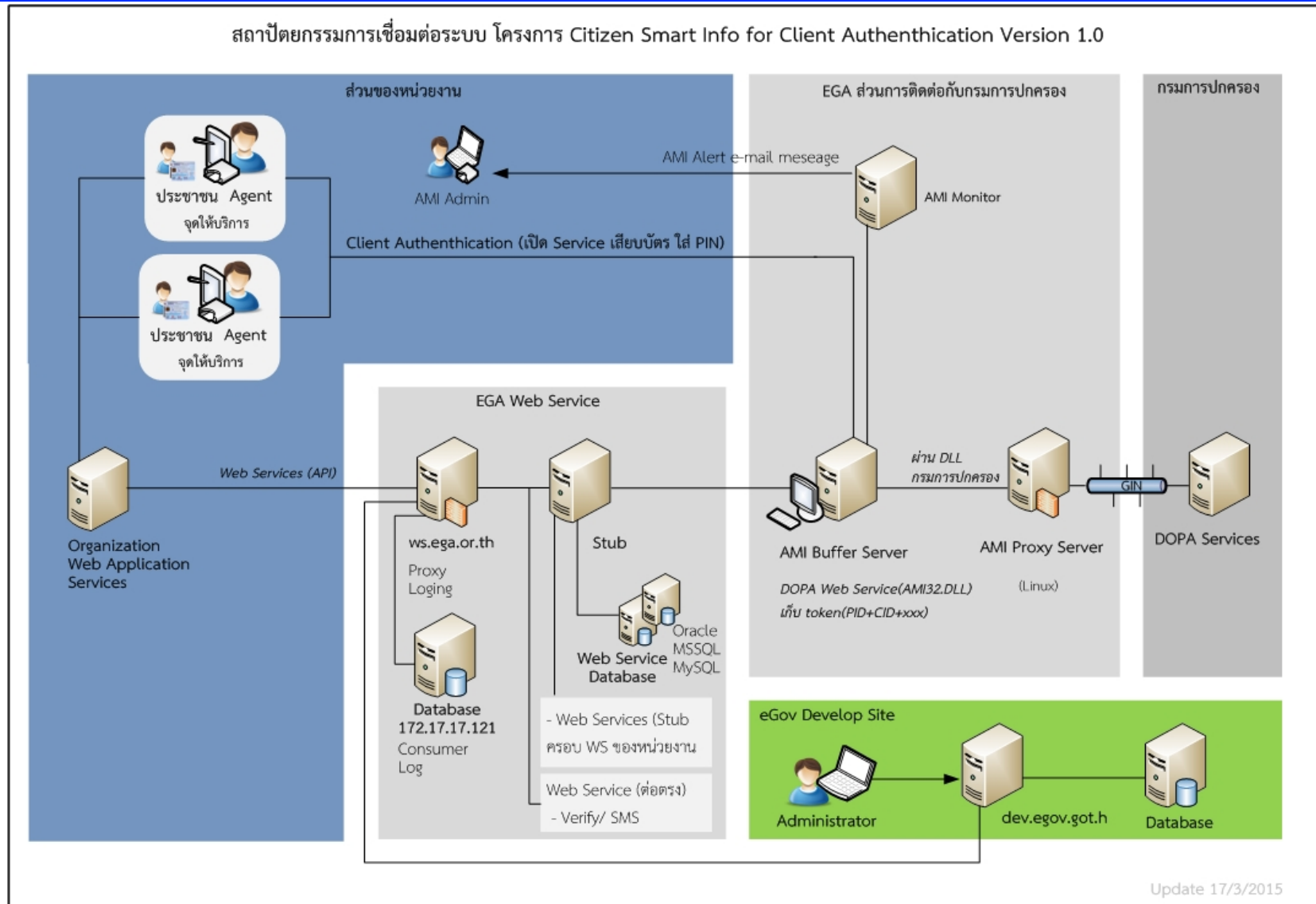
Development Site – ส่วนจัดการ Key, Account, Endpoint Services และ Log

สถาปัตยกรรม Centralized Authenticate

AMI Demo แบบ Centralized Authenticate



สถาปัตยกรรม Client Authenticate



สถาปัตยกรรม Client Authenticate

Client Authenticate Application – สำหรับการ Authenticate โดยเจ้าหน้าที่หน่วยงานที่ลงทะเบียนสิทธิ์กับการปกครอง(Agent) สามารถเรียกดูข้อมูลบุคคล ผ่านสิทธิ์ของตนเองได้(ไม่ต้องใช้สิทธิ์กลางแบบ Centralized Authenticate)

โดยใช้งานผ่าน **Client Authenticate Application** - ในโปรแกรมจะระบุ Port และ PassCode ของหน่วยงานไว้ สามารถใช้กับ Agent ก็คนก็ได้(ตามที่ขอสิทธิ์กับการปกครองไว้ โดยไม่ต้องแจ้งรายชื่อกับ สรอ.)

AMI: Buffer Server – สำหรับเป็นตัวแทนของ Client Authenticate Application ในการ Authenticate เพื่อสร้าง Session ในการดึงข้อมูลจากกรมการปกครองผ่าน Proxy Server

สถาปัตยกรรม Client Authenticate

AMI Demo แบบ Client Authenticate

The screenshot shows the 'AMI Buffer' application window. At the top, there are three buttons: '(1) Read Card', '(2) Log In' (which is highlighted with a dotted border), and '(3) Close'. Below these are input fields for 'Server IP' (104.110.9.100), 'Port' (8080), and 'Passcode' (10004040). A green status bar displays '00000 - Success'. The main area contains a log of transactions:

```
2558-07-27 17:07:01 >> PIN2 correct
2558-07-27 17:07:01 >> ..SUCCESS..

2558-07-27 17:07:01 >> <-SELECT APPLET : ADM->

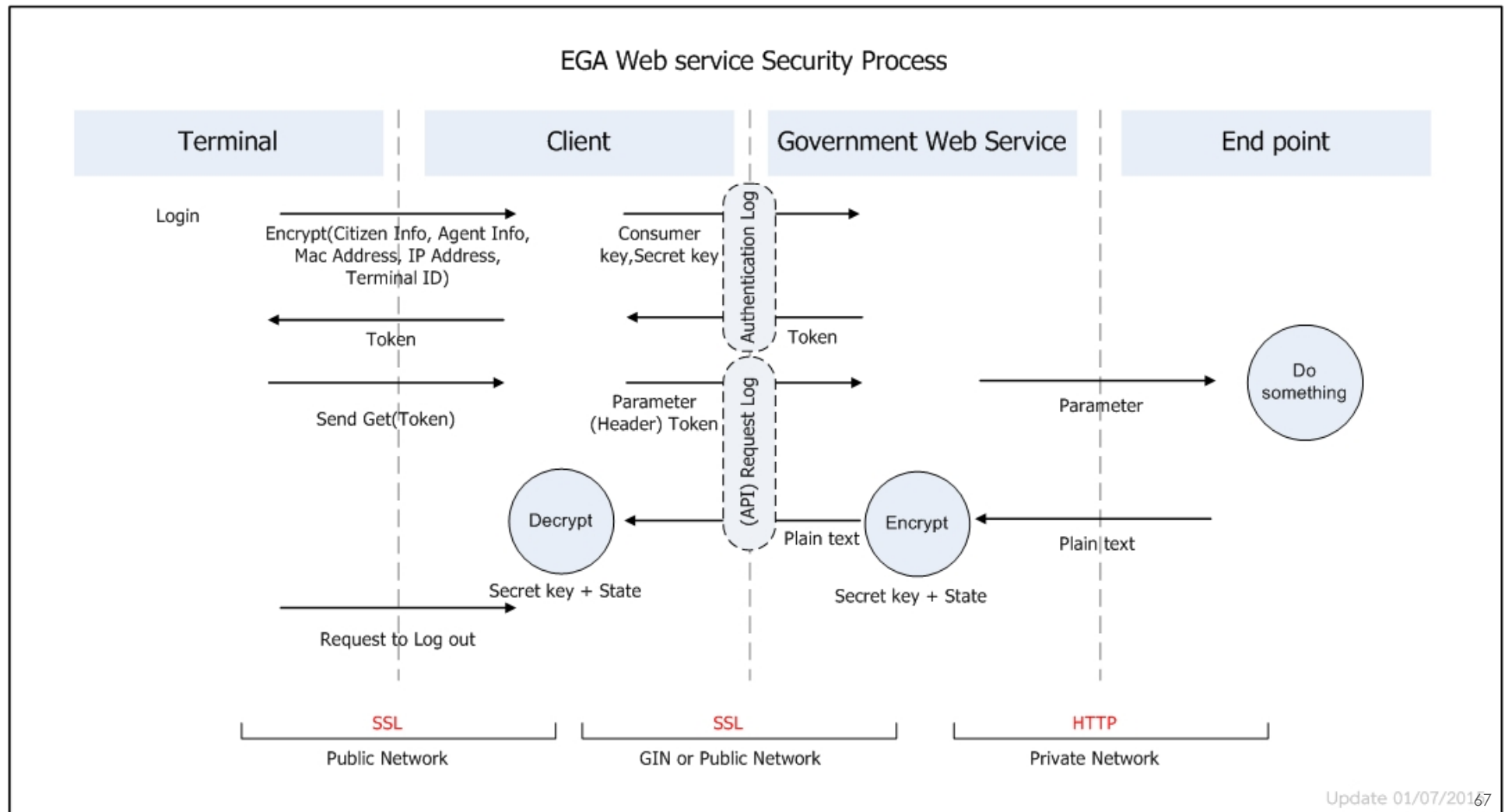
2558-07-27 17:07:01 >> <-GET MATCH STATUS->
2558-07-27 17:07:01 >> MATCH STATUS = MATCH
2558-07-27 17:07:01 >> CRYPTOGRAM
2558-07-27 17:07:01 >>
dcf69915bc8e4b4b48ba238b757ede3355
2558-07-27 17:07:01 >> ..SUCCESS..

2558-07-27 17:07:01 >> <-ENVELOPE GM->
2558-07-27 17:07:01 >> INT AUTH KEY ID=1
2558-07-27 17:07:01 >> ENVELOPE: 204
2558-07-27 17:07:01 >>
5341534d0001005e0c3b6800000073c84012
13018000489862423425230274253303b000
0120dcf69915bc8e4b4b48ba238b757ede33

2558-07-27 17:07:02 >> DataArrival: AP0102,00000
2558-07-27 17:07:02 >> LogIn: Complete
2558-07-27 17:07:02 >> ReturnCode: 00000
```

การทำงานของ ws.ega.or.th

ควบคุมการส่งค่าข้อมูลโดยใช้ Key ผ่าน SSL รวมกับการ Encrypt และการยืนยันค่าของผู้ส่ง-ผู้รับข้อมูล ตั้งแต่เครื่อง Terminal จนถึง Services End point ดังรูป



การทำงานของ ws.ega.or.th

Consumer Key : เป็นชุด...ที่ สรอ. ออกให้ เพื่อความปลอดภัยในการเรียกใช้งาน API (เปรียบกับ username)

Consumer Secret : เป็นชุดรหัสที่ สรอ. ออกให้ เพื่อความปลอดภัยในการเรียกใช้งาน API (เปรียบกับ password)

Token : ระบบของ สรอ. สร้างให้และส่งกลับมา เพื่อนำไปใช้ในการพัฒนา API เรียกใช้ข้อมูล
(หมายเหตุ หลังจากที่ได้ Validate เรียบร้อยแล้ว กรณีที่ไม่มีการเรียกใช้งาน API ใด ๆ ภายใน 60 นาที ระบบจะร้องขอให้ทำการ Validate ใหม่ (Token หมดอายุการใช้งาน))

การทำงานของ ws.ega.or.th

Terminal เป็นส่วนติดต่อกับประชาชน และเป็นเครื่องที่ติดตั้ง Smart App Center ไว้

Client/Application คือ Application ที่ติดตั้งไว้บน Terminal จะมีหน้าที่เป็นตัวกลางในการยืนยันตัวบุคคล และเป็นช่องทางในการเข้าไปใช้บริการ โดยเริ่มจากการ Request Token จากการ Validate ชุด Consumer Key ไปยัง ws.ega.or.th (จะมี Demo การเรียกใช้ตอนท้าย)

การทำงานของ ws.ega.or.th

Web Service คือ ส่วนที่ทำหน้าที่ในการจัดการ Token ในการใช้งานต่างๆ และตรวจสอบความผิดปกติของการเข้าถึงบริการ

Endpoint เป็นบริการปลายทางของแต่ละหน่วยงาน ที่ใช้ในการแสดงผลข้อมูลให้ประชาชนทราบข้อมูลที่เกี่ยวข้องกับบริการนั้นๆ
Endpoint เป็น Service ปลายทางของแต่ละหน่วยงานที่แสดงข้อมูล

ข้อกำหนดในการเรียกใช้บริการข้อมูล

1 ปริมาณการใช้ข้อมูล

เช่น 300 ครั้งต่อวัน หรือตามความเหมาะสมของภารกิจ

2 ช่วงระยะเวลาในการเข้าถึงข้อมูล

สามารถระบุช่วงเวลาในการเข้าถึง เช่น ช่วงเวลา 8.30 น. - 17.30 น. (วันธรรมดา) หรือตามสิทธิของภารกิจที่ขอใช้งาน(แต่ละหน่วยงานไม่จำเป็นต้องเป็นช่วงเวลาเดียวกัน)

3 สิทธิในการเข้าถึงข้อมูล

หน่วยงานจำเป็นต้องขออนุญาตจากกรมการปกครองก่อนเป็นอันดับแรก (หน่วยงานต้องทำ MOU ร่วมกับกรมการปกครองเพื่อเรียกใช้ข้อมูลนั้นๆ แล้วเท่านั้น)

ข้อกำหนดในการเรียกใช้บริการข้อมูล

4 การยืนยันตัวตนบุคคลเพื่อเข้าถึงข้อมูล

หน่วยงานผู้ร้องขอจำเป็นต้องทำการ Remote เพื่อเสียบบัตรประชาชน (ของผู้มีสิทธิที่ได้ลงทะเบียนไว้กับกรมการปกครองเรียบร้อยแล้ว) และใส่ PIN Code เปิดสิทธิ์การขอเรียกใช้ข้อมูล (Government Web Service) ที่เชื่อมต่อไปยังฐานข้อมูลทะเบียนราษฎร กรมการปกครอง

ข้อกำหนดในการเรียกใช้บริการข้อมูล

5 อื่น ๆ

เจ้าหน้าที่ผู้มีสิทธิจำเป็นจะต้องมี PIN Code 4 หลัก เพื่อใช้ในการ Remote เปิดบริการเรียกใช้ข้อมูล (Government Web Service)

หมายเหตุ:

-สามารถขอ PIN Code ได้ที่สำนักทะเบียนอำเภอ สำนักงานเขตทั่วประเทศ

-กรณีที่หน่วยงานใช้บัตรประชาชนของเจ้าของข้อมูลในการเรียกขอข้อมูลเพิ่มเติม(เรียกแบบ Centralized Authenticate) และต้องการจัดเก็บประวัติการเรียกใช้ข้อมูล (Log) การใช้งาน EGA Web Service ในทุก ๆ Transaction นั้น เจ้าหน้าที่ของหน่วยงานจำเป็นต้องสียบบัตรประชาชนของตนเองก่อนการสียบบัตรประชาชนของเจ้าของข้อมูล (ประชาชน) เพื่อสามารถตรวจสอบว่าเจ้าหน้าที่คนใดเป็นผู้เรียกใช้ข้อมูลของเจ้าของข้อมูลนั้น ๆ

การใช้งาน AMI

AMI

การใช้งาน AMI

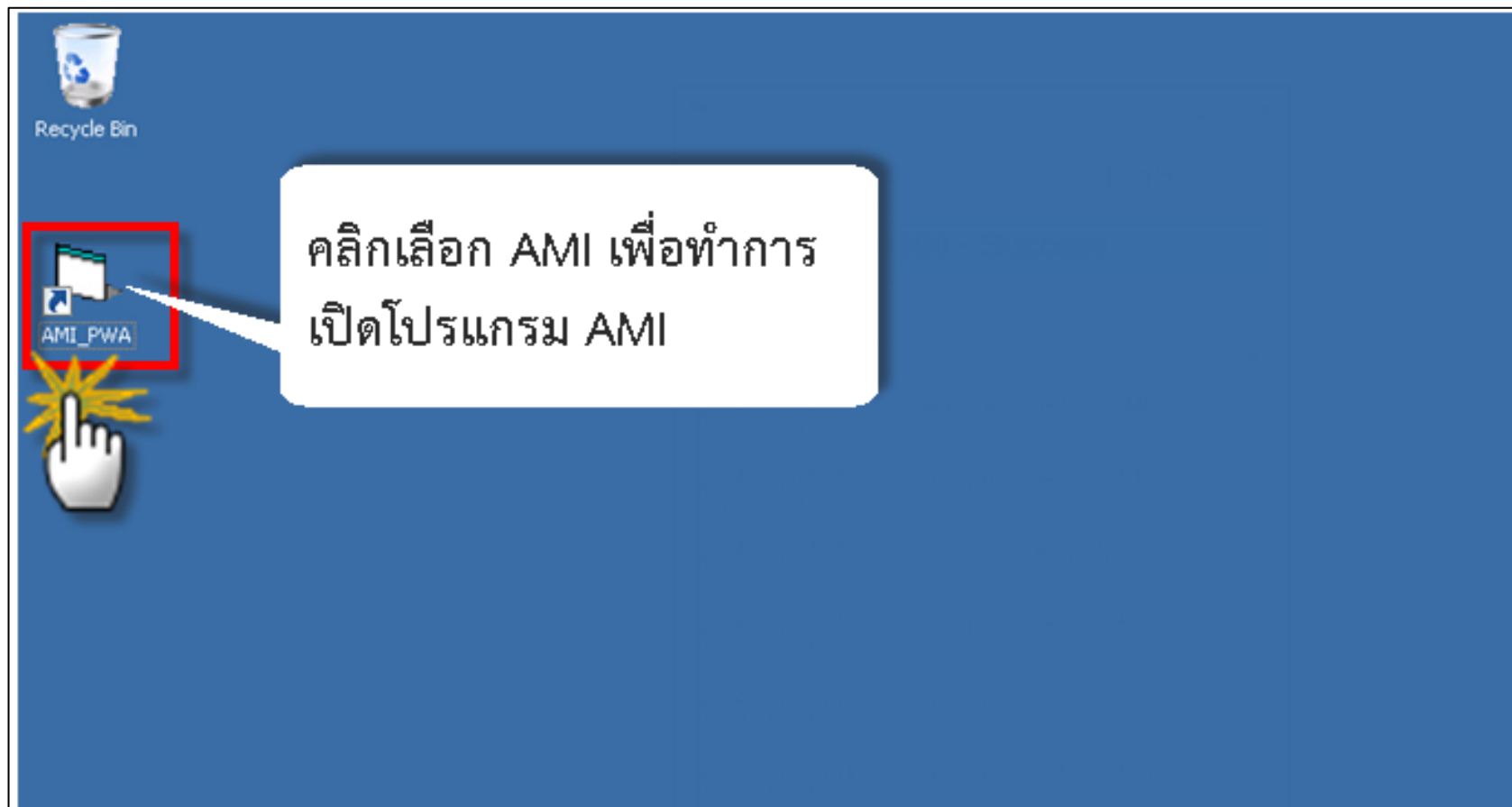
เตรียมการเพื่อเปิดใช้งาน AMI

- 1 เตรียมบัตรประจำตัวประชาชนที่ได้รับสิทธิ์ และมี PIN code แล้ว
- 2 เครื่อง Smart Card Reader และเสียบสาย USB ก่อนเปิดโปรแกรม
- 3 Username และ Password สำหรับ Remote Desktop
(ทาง สรอ.จะจัดส่งให้ทาง e-mail ที่ระบุไว้ในใบคำขอใช้บริการ API)



การใช้งาน AMI

เปิดโปรแกรม AMI_XXX



การใช้งาน AMI

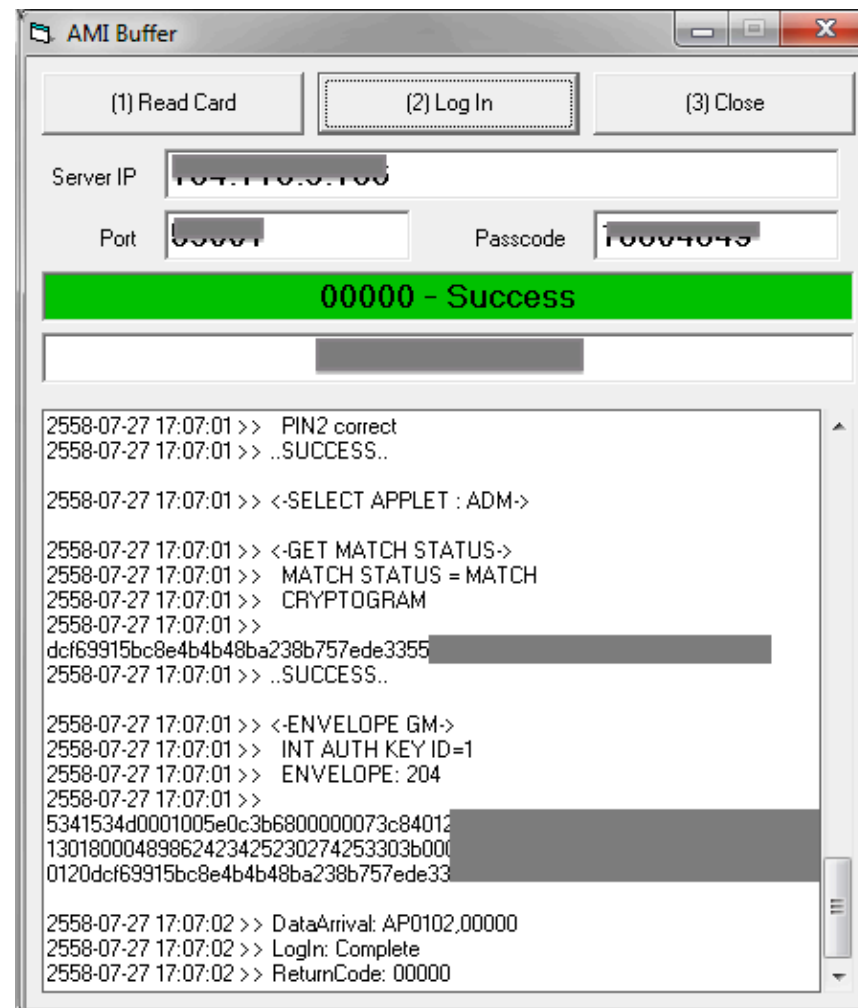
เมื่อทำการเปิดโปรแกรมแล้ว

1 ทำการเสียบบัตรประจำตัวประชาชน

2 กด Read Card

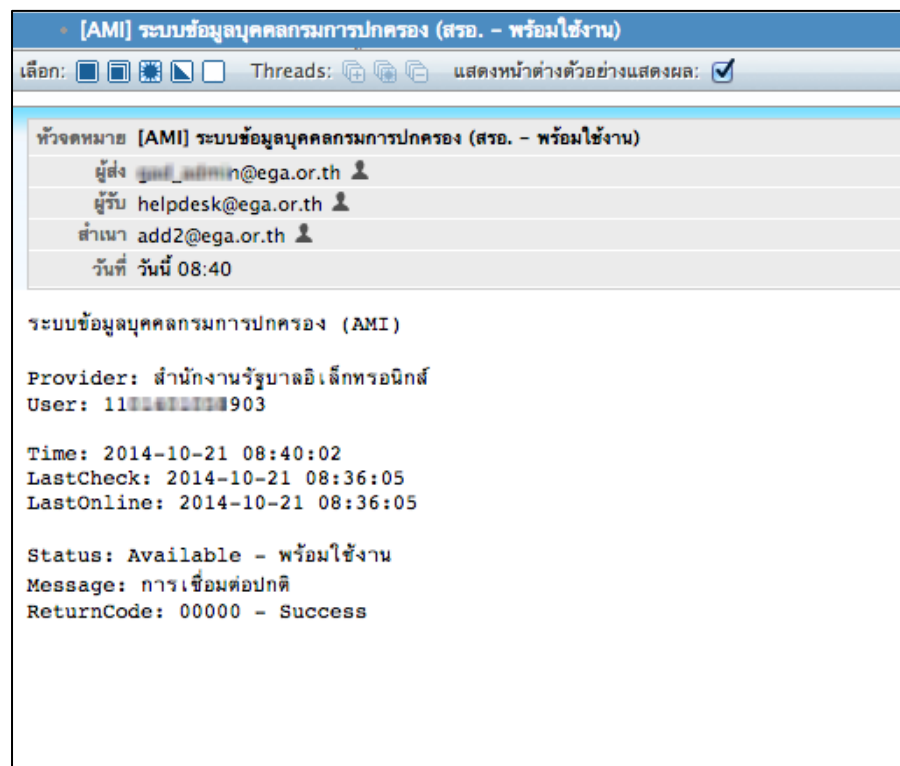
3 ใส่ PIN Code

หากสำเร็จ จะแสดงสถานะ Success (สีเขียว)
Return code "00000"



การใช้งาน AMI

ทั้งนี้ระบบจะมีการแจ้งเตือนสถานะการเข้าใช้งานของโปรแกรมทางอีเมลให้ผู้ใช้งานได้ทราบถึงสถานะการเข้าใช้งานทุกวัน (ถึงแม้ว่าจะไม่มี การเข้าใช้งานโปรแกรมในวันนั้นๆ) โดยเนื้อหาของอีเมลดังกล่าว เป็น ดังภาพ



การใช้งาน AMI

ข้อกำหนดทางเทคนิคเครื่องอ่านและเขียนบัตรประจำตัวประชาชน
เอนกประสงค์

- สามารถใช้งานได้ตามมาตรฐาน Microsoft Personal Computer / Smart Card(PC/SC)
- สามารถอ่านและเขียนบัตรประจำตัวประชาชนแบบเอนกประสงค์ (Smart Card)ได้ตามมาตรฐาน ISO/IEC 7816
- สามารถทำการอ่านและเขียนบัตรประจำตัวประชาชนแบบเอนกประสงค์ของกรมการปกครองได้อย่างมีประสิทธิภาพ
- สามารถเชื่อมต่อกับเครื่องคอมพิวเตอร์ที่ติดตั้งระบบปฏิบัติการ Microsoft Windows XP, VISTA และ Windows 7 ได้เป็นอย่างดี
- สามารถเชื่อมต่อกับเครื่องคอมพิวเตอร์ผ่านทาง USB Port
- เครื่องอ่านบัตรต้องได้มาตรฐาน FCC หรือ UL เป็นอย่างน้อย

การเรียกใช้ API

API

การเรียกใช้ API

1 ทำการ Validate

1.1 ส่วนของ Consumer Key และ Consumer Secret เพื่อขอ Token และเปิด Session ในการเรียกใช้ Service โดยการ HTTP Request โดยวิธี GET

```
https://ws.ega.or.th/ws/auth/validate?  
ConsumerSecret=<ConsumerSecret>&  
AgentID=<AgentID>
```

การเรียกใช้ API

Key/ Parameter	Required	Type	Description
ConsumerSecret	Required	String	ชุดรหัสที่ สรอ. ออกให้ เพื่อความปลอดภัยในการเรียกใช้งาน API
AgentID	Required	String	• เลขบัตรประชาชนของผู้ที่เข้ามาดูข้อมูลในระบบ • เป็นค่าที่สามารถบ่งบอกได้ว่า User ใดที่เข้ามาใช้งาน Service ใน Session นั้น ๆ • เป็นค่า Unique ID ของแต่ละ Session โดยทาง สรอ. กำหนดไว้ให้

การเรียกใช้ API

1.2 ส่วนของ HTTP HEADER

หากการทำงานไม่มีข้อผิดพลาด Service จะทำการตอบกลับด้วย http status code: 200 OK และ TOKEN ที่สามารถนำไปใช้ประกอบการพัฒนา API ในการเรียกขอข้อมูลต่าง ๆ ดังนี้

200 OK

```
{  
  "Result": "<TOKEN_KEY>"  
}
```

การเรียกใช้ API

โดย Status Code ที่ระบบจะ แจ้งกลับ (Response) มายัง
ต้นทางทราบ มีดังนี้

Status Code	Status Phrase	Description
200	OK	Service ทำการคืนค่าได้เป็นปกติ
204	No Content	ไม่พบข้อมูลที่ต้องการ
400	Bad Request	มีการกำหนด Parameter มาไม่ครบหรือไม่ถูกต้อง
401	Unauthorized	ยัง <u>ไม่</u> ได้ทำการ Validate Consumer Key/ Token หมดอายุการใช้งาน
403	Forbidden	<u>ไม่</u> ได้รับอนุญาตให้ใช้งาน Service ที่กำลังเรียกใช้
404	Not Found	ไม่พบ Service ที่เรียกใช้
405	Method Not Allowed	เรียกใช้งาน Service ผ่าน http Method ที่ <u>ไม่</u> ถูกต้อง
500	Internal Server Error	มีข้อผิดพลาดเกิดขึ้นขณะที่กำลังทำงาน
503	Service Unavailable	Service ปลายทาง <u>ไม่</u> สามารถให้บริการได้

การเรียกใช้ API

หมายเหตุ: .หลังจากที่ Validate เรียบร้อยแล้ว กรณีที่ไม่มี การเรียกใช้งาน API ใด ๆ ภายใน 60 นาที ระบบจะร้องขอ ให้ทำการ Validate ใหม่ (Token หมดอายุการใช้งาน)

การเรียกใช้ API

2 ทำการเรียก HTTP Request โดยวิธี GET และต้องทำการใส่ HEADER ดังนี้

Parameter	Value
Content-Type	application/x-www-form-urlencoded
Consumer-Key	ชุดรหัสที่ สรอ. ออกให้
Token	TOKEN ที่ได้รับกลับมาจาก สรอ. ในการ Validate ข้างต้น

ตัวอย่าง Request

```
https://ws.ega.or.th/ws/dopa/personal/profile/  
simple?CitizenID=<CitizenID>
```

การเรียกใช้ API

เช่น

Content-Type : application/x-www-form-urlencoded

Consumer-Key : 8e2ed330-200d-4d7e-afea-8b240dfa986b

Token : b8743466-6894-4620-b173-cb99af7d72cd

[https://ws.ega.or.th/ws/dopa/personal/profile/
simple?CitizenID=1234567890123](https://ws.ega.or.th/ws/dopa/personal/profile/simple?CitizenID=1234567890123)

การเรียกใช้ API

หมายเหตุ

หลังจากที่มีการเรียกใช้งาน API ที่ต้องการแล้วนั้น กรณีที่มีการคืนค่า (Status Code) เป็น **"500"** (Internal Server Error) กลับมานั้น ระบบจะแสดง Content ของการข้อผิดพลาดที่เกิดขึ้นตาม Code ต่าง ๆ ดังนี้

ตัวอย่าง

```
{  
    "Message" : "[Status] Message"  
}
```

การเรียกใช้ API

ตัวอย่าง Status Code (ดูเพิ่มเติมตามคู่มือ)

Status Code	Message	Description
00000	Success	การเชื่อมต่อปกติ
00001	Next record not found	ไม่พบข้อมูลถัดไป
00101	(Unknown code)	(Unknown code)
00410	No input data	ไม่พบข้อมูลนำเข้า
00815	Record not found	ไม่พบข้อมูลที่ต้องการ
00870	Next record not found	ไม่พบข้อมูลถัดไปหรือก่อนหน้า
90005	Out of time	ปิดบริการ หรืออยู่นอกช่วงเวลาทำการ

การเรียกใช้ API

ตัวอย่าง Status Code (ดูเพิ่มเติมตามคู่มือ)

90008	Out of quota	ใช้สิทธิ์ในการตรวจสอบข้อมูลครบกำหนดแล้ว
90010	Can't connect to AMI Search server	ไม่สามารถติดต่อเซิร์ฟเวอร์ AMI Search ได้
90011	incorrect input	คำสั่งที่ส่งไปไม่ถูกต้อง
90021	Invalid password	ไม่สามารถแก้ไขรหัสลับได้
90028	No user logged in	เจ้าหน้าที่ไม่ได้ทำการเข้าระบบ
90046	No permission	เจ้าหน้าที่ไม่มีสิทธิ์ในการทำงาน
99702	Record not found	ไม่พบรายการการเปลี่ยนแปลงชื่อในฐานข้อมูล
99706	Record not found	ไม่พบข้อมูลชื่อกลาง
99707	Record not found	ไม่พบข้อมูลชื่อภาษาอังกฤษ
99999	Invalid RequestNo	เจ้าหน้าที่ไม่ได้รับอนุญาตในการใช้คำสั่งที่ระบุ
X0001	Can't connect to AMI Proxy server	ไม่สามารถติดต่อเซิร์ฟเวอร์ AMI Proxy ได้
X0002	Incorrect input	จำนวนข้อมูลนำเข้าไม่ถูกต้องตามคำสั่งที่ระบุ
X0003	Can't connect to AMI Connector	ไม่สามารถติดต่อโปรแกรม AMI Connector ได้

การเรียกใช้ API

ตัวอย่าง **API** ในการขอข้อมูลเกี่ยวกับบุคคล

การขอข้อมูลส่วนบุคคลแบบทั่วไปนั้น ระบบจะทำการตอบกลับข้อมูล (Response) ได้แก่

หมายเลขบัตรประชาชน 13 หลัก
คำนำหน้าชื่อ (ภาษาไทย)
ชื่อจริง-ชื่อกกลาง-นามสกุล (ภาษาไทย)
เพศ
วัน/เดือน/ปี พ.ศ. เกิด
อายุ
สถานะการมีชีวิต

การเรียกใช้ API

Request

การเรียกขอข้อมูลส่วนตัวแบบทั่วไปของบุคคลนั้น ๆ (HTTP Request) ในรูปแบบ API ต้องทำการเรียกผ่าน Url และ Parameter ดังนี้

วิธีการ	Url
GET	<a href="https://ws.ega.or.th/ws/dopa/personal/profile/simple?CitizenID=<CitizenID>">https://ws.ega.or.th/ws/dopa/personal/profile/simple?CitizenID=<CitizenID>

Parameter	Required	Type	Description
CitizenID	Required	String	หมายเลขบัตรประชาชน 13 หลัก ของบุคคลที่ต้องการทราบข้อมูล

การเรียกใช้ API

Response

หลังจากที่ส่ง Request ข้างต้นไปเรียบร้อยแล้ว
ระบบจะทำการส่ง Response กลับมา ดังนี้

การเรียกใช้ API

```
{  
  "CitizenID":"143XXXXXXXXXX",  
  "NameEN_Prefix":null,  
  "NameEN_FirstName":null,  
  "NameEN_MiddleName":null,  
  "NameEN_SurName":null,  
  "NameTH_Prefix":"นาย",  
  "NameTH_FirstName":"XXXXX",  
  "NameTH_MiddleName":"","  
  "NameTH_SurName":"XXXXXX",  
  "SexID":1,  
  "Sex":"ชาย",  
  "BirthDate":"25XXXXXX",  
  "Age":"29",  
  "PersonStatusID":1,  
  "PersonStatus":"มีชีวิตอยู่",  
  "AddressID":null,  
  "Address_No":null,  
  "Address_Moo":null,  
  "Address_Alley":null,  
  "Address_Soi":null,  
  "Address_Road":null,  
  
  "Address_Tumbol":null,  
  "Address_Amphur":null,  
  "Address_Province":null,  
  "IssueDate":null,  
  "ExpireDate":null,  
  "IssuerID":null,  
  "IssuerPlace":null,  
  "IssuerAgency":null,  
  "Nationality":null,  
  "Nationality_Old":null,  
  "Nationality_ChangeDate":null,  
  "Father_CitizenID":null,  
  "Father_Nationality":null,  
  "Father_NameTH_FirstName":null,  
  "Father_FirstName":null,  
  "Mother_CitizenID":null,  
  "Mother_Nationality":null,  
  "Mother_NameTH_FirstName":null,  
  "Mother_FirstName":null,  
  "DomicileType":null,  
  "DomicileStatus":null,  
  "DomicileDate":null,  
}
```

การเรียกใช้ API

Name	Type	Description
CitizenID	String	หมายเลขบัตรประจำตัวประชาชน 13 หลัก
NameTH_Prefix	String	คำนำหน้าชื่อ (ภาษาไทย)
NameTH_FirstName	String	ชื่อจริง (ภาษาไทย)
NameTH_MiddleName	String	ชื่อกกลาง (ภาษาไทย)
NameTH_SurName	String	นามสกุล (ภาษาไทย)
SexID	String	รหัสเพศ ได้แก่ • 0 คือ ไม่ระบุ • 1 คือ ชาย • 2 คือ หญิง
Sex	String	เพศ
BirthDate	String	วัน-เดือน-ปี พ.ศ. เกิด โดยมีรูปแบบเป็น YYYYMMDD
Age	String	อายุ
PersonStatusID	String	รหัสสถานบุคคล ได้แก่ • 0 คือ ไม่ระบุ • 1 คือ มีชีวิตอยู่ • 2 คือ เสียชีวิต • 3 คือ จำหน่าย
PersonStatus	String	สถานบุคคล ได้แก่ ไม่ระบุ มีชีวิตอยู่ เสียชีวิต และจำหน่าย

API ในระบบ

ข้อมูลส่วนบุคคลแบบทั่วไป

```
https://ws.ega.or.th/ws/dopa/personal/profile/simple?  
CitizenID=<CitizenID>
```

การขอข้อมูลส่วนบุคคลแบบทั่วไปนั้น ระบบจะทำการตอบกลับข้อมูล (Response) อันได้แก่

- หมายเลขบัตรประชาชน 13 หลัก
- คำนำหน้าชื่อ (ภาษาไทย)
- ชื่อจริง-ชื่อกกลาง-นามสกุล (ภาษาไทย)
- เพศ
- วัน/เดือน/ปี พ.ศ. เกิด
- อายุ
- สถานะการมีชีวิตร

API ในระบบ

ข้อมูลส่วนบุคคลแบบธรรมดา

```
https://ws.ega.or.th/ws/dopa/personal/profile/normal?  
CitizenID=<CitizenID>
```

การขอข้อมูลส่วนบุคคลแบบธรรมดานั้น ระบบจะทำการตอบกลับข้อมูล (Response) อันได้แก่

- หมายเลขบัตรประชาชน 13 หลัก
- คำนำหน้าชื่อ (ภาษาไทย)
- ชื่อจริง-ชื่อกกลาง-นามสกุล (ภาษาไทย)
- เพศ
- วัน/เดือน/ปี พ.ศ. เกิด

API ในระบบ

ข้อมูลส่วนบุคคลแบบธรรมดา

- อายุ
- สถานะการมีชีวิต
- ข้อมูลที่อยู่
- ข้อมูลทำบัตรประชาชน (เช่น วัน/เดือน/ปี พ.ศ. ที่ทำบัตร วัน/เดือน/ปี พ.ศ. ที่บัตรหมดอายุ สถานที่ทำบัตร หน่วยงานที่ทำบัตร)
- ข้อมูลสัญชาติ (เช่น สัญชาติปัจจุบัน สัญชาติเดิม วัน/เดือน/ปี พ.ศ. ที่เปลี่ยนสัญชาติ)

API ในระบบ

ข้อมูลส่วนบุคคลแบบทั้งหมด

```
https://ws.ega.or.th/ws/dopa/personal/profile/full?  
CitizenID=<CitizenID>
```

การขอข้อมูลส่วนบุคคลแบบทั้งหมดนั้น ระบบจะทำการตอบกลับ
ข้อมูล (Response) อันได้แก่

- หมายเลขบัตรประชาชน 13 หลัก
- คำนำหน้าชื่อ (ภาษาไทย)
- ชื่อจริง-ชื่อกกลาง-นามสกุล (ภาษาไทย)
- เพศ
- วัน/เดือน/ปี พ.ศ. เกิด

API ในระบบ

ข้อมูลส่วนบุคคลแบบทั้งหมด

- อายุ
- สถานะการมีชีวิต
- ข้อมูลที่อยู่
- ข้อมูลทำบัตรประชาชน (เช่น วัน/เดือน/ปี พ.ศ. ที่ทำบัตร วัน/เดือน/ปี พ.ศ. ที่บัตรหมดอายุ สถานที่ทำบัตร หน่วยงานที่ทำบัตร)
- ข้อมูลสัญชาติ (เช่น สัญชาติปัจจุบัน สัญชาติเดิม วัน/เดือน/ปี พ.ศ. ที่เปลี่ยนสัญชาติ)
- ข้อมูลบิดา (เช่น หมายเลขบัตรประชาชน สัญชาติ ชื่อจริง)**
- ข้อมูลมารดา (เช่น หมายเลขบัตรประชาชน สัญชาติ ชื่อจริง)**
- ข้อมูลภูมิลำเนา (เช่น สถานะการอยู่อาศัย (ตามทะเบียนบ้าน) วัน/เดือน/ปี พ.ศ. ที่ย้ายเข้า)

API ในระบบ

ข้อมูลส่วนบุคคลแบบสมบูรณ์

```
https://ws.ega.or.th/ws/dopa/personal/profile/extra?  
CitizenID=<CitizenID>
```

การขอข้อมูลส่วนบุคคลแบบสมบูรณ์นั้น ระบบจะทำการตอบกลับข้อมูล (Response) อันได้แก่

- หมายเลขบัตรประชาชน 13 หลัก
- คำนำหน้าชื่อ (ภาษาไทย)
- คำนำหน้าชื่อ (ภาษาอังกฤษ)
- ชื่อจริง-ชื่อกกลาง-นามสกุล (ภาษาไทย)
- ชื่อจริง-ชื่อกกลาง-นามสกุล (ภาษาอังกฤษ)
- เพศ

API ในระบบ

ข้อมูลส่วนบุคคลแบบสมบูรณ์

- วัน/เดือน/ปี พ.ศ. เกิด
- อายุ
- สถานะการมีชีวิตร
- ข้อมูลที่อยู่
- ข้อมูลทำบัตรประชาชน (เช่น วัน/เดือน/ปี พ.ศ. ที่ทำบัตร วัน/เดือน/ปี พ.ศ. ที่บัตรหมดอายุ สถานที่ทำบัตร หน่วยงานที่ทำบัตร)
- ข้อมูลสัญชาติ (เช่น สัญชาติปัจจุบัน สัญชาติเดิม วัน/เดือน/ปี พ.ศ. ที่เปลี่ยนสัญชาติ)
- ข้อมูลบิดา (เช่น หมายเลขบัตรประชาชน สัญชาติ ชื่อจริง)
- ข้อมูลมารดา (เช่น หมายเลขบัตรประชาชน สัญชาติ ชื่อจริง)
- ข้อมูลภูมิลำเนา (เช่น สถานะการอยู่อาศัย (ตามทะเบียนบ้าน) วัน/เดือน/ปี พ.ศ. ที่ย้ายเข้า)

API ในระบบ

ข้อมูลที่อยู่

```
https://ws.ega.or.th/ws/dopa/house/profile?  
CitizenID=<CitizenID>
```

API ในระบบ

ข้อมูลที่อยู่

```
{
  "AddressID" : "00000000000000000000000000000000",
  "RegistryOffice" : "ท้องถิ่นเขตเมืองหลวง",
  "Address_No" : "78/15",
  "Address_Moo" : "",
  "Address_Alley" : "",
  "Address_Soi" : "ซอยสุขุมวิท 15",
  "Address_Road" : "",
  "Address_Tumbol" : "เมืองหลวง",
  "Address_Amphur" : "เขตเมืองหลวง",
  "Address_Province" : "กรุงเทพมหานคร",
  "HouseType" : "อาคารชุด",
  "PermitDate" : "2555-01-01",
  "DisposeDate" : "",
  "VillageName" : "",
  "BuildingName" : "ตึกทิมทิม",
  "Location" : "ขวา",
  "PostCode" : "10110",
  "HouseAttribute" : "",
  "HouseSize" : "",
  "HouseVolumn" : "",
  "OwnerFile" : "",
  "OwnerFileID" : "",
  "LandSize" : "",
  "Tel01" : "",
  "Tel02" : "",
  "Tel03" : "",
  "Tel04" : "",
  "Tel05" : "",
  "Tel06" : "",
  "Tel07" : "",
  "Tel08" : "",
  "Tel09" : "",
  "Tel10" : "",
  "ApproveID" : "00000000000000000000000000000000",
  "CommunityName" : ""
}
```

API ในระบบ

ข้อมูลสมาชิกในบ้าน

```
https://ws.ega.or.th/ws/dopa/house/person?  
CitizenID=<CitizenID>
```

API ในระบบ

ข้อมูลสมาชิกในบ้าน

```
{
    "Count": "4",
    "Person": [
        {
            "CitizenID": "9-๒๐๖๓๙๘๑๕๗๔๒",
            "NameTH_FirstName": "พิชญ์",
            "NameTH_SurName": "อินทนะ",
            "Sex": "ชาย",
            "BirthDate": "25360801",
            "AddressID": "๒๕๖-๘๘๘๘๘-๗๕"
        },
        {
            "CitizenID": "9-๒๐๖๓๙๘๑๕๗๔๒",
            "NameTH_FirstName": "พิชญ์",
            "NameTH_SurName": "อินทนะ",
            "Sex": "ชาย",
            "BirthDate": "2536๐7๒๒",
            "AddressID": "๒๕๖-๘๘๘๘๘-๗๕"
        },
        {
            "CitizenID": "9-๒๐๖๓๙๘๑๕๗๔๒",
            "NameTH_FirstName": "พิชญ์",
            "NameTH_SurName": "อินทนะ",
            "Sex": "หญิง",
            "BirthDate": "253๘๐7๒๒",
            "AddressID": "๒๕๖-๘๘๘๘๘-๗๕"
        },
        {
            "CitizenID": "9-๒๐๖๓๙๘๑๕๗๔๒",
            "NameTH_FirstName": "พิชญ์",
            "NameTH_SurName": "อินทนะ",
            "Sex": "หญิง",
            "BirthDate": "254๒0๘๑1",
            "AddressID": "๒๕๖-๘๘๘๘๘-๗๕"
        }
    ]
}
```

API ในระบบ

ข้อมูลการเปลี่ยนชื่อ-นามสกุล

```
https://ws.ega.or.th/ws/dopa/house/person?  
CitizenID=<CitizenID>
```

API ในระบบ

ข้อมูลการเปลี่ยนชื่อ-นามสกุล

```
{
  "Count":1,
  "Change":[
    {
      "ChangeDate":"25500309",
      "Place":"ท้องถิ่นเทศบาลเมืองหนองสำโรง",
      "TypeID":"103",
      "TypeName":"เปลี่ยนนามสกุลโดย ท.ร. 97",
      "TypeGroup":"นามสกุล",
      "DataOld":"XXXXXXXXXX",
      "DataNew":"XXXXXXXXX"
    }
  ]
}
```

ตัวอย่าง source code - Online

C#.Net

http://sc.ega.or.th/tutorials/smartservice_csharp/tree/master

Java

http://sc.ega.or.th/tutorials/dopa-api_java/tree/master

PHP

http://sc.ega.or.th/tutorials/dopa-api_php/tree/master

ทดลองใช้ API

KEY สำหรับทดสอบเรียกใช้ API เพื่อดึงข้อมูล Dummy

ConsumerKey: 8e2ed330-200d-4d7e-afea-8b240dfa986b

ConsumerSecret: Ea1hOqtJnG6

หมายเหตุ

- สามารถนำ key ไปทดสอบได้เลย

- วิธีการใช้งานเพิ่มเติมตาม “คู่มือประกอบการพัฒนาระบบการพัฒนา API” ในการเรียกใช้ข้อมูลบุคคล ข้อ6 (download คู่มือได้ที่ dev.egov.go.th)

ทดลองใช้ API

API ที่สามารถทดสอบกับข้อมูล Dummy ได้
รายการข้อมูลตาม คู่มือประกอบการพัฒนาระบบการพัฒนา API ใน
การเรียกใช้ข้อมูลบุคคล ข้อ 7

ทดลองเรียกใช้ได้ 4 แบบ

<https://ws.ega.or.th/ws/dopa/personal/profile/simple?CitizenID=<CitizenID>>

<https://ws.ega.or.th/ws/dopa/personal/profile/normal?CitizenID=<CitizenID>>

<https://ws.ega.or.th/ws/dopa/personal/profile/full?CitizenID=<CitizenID>>

<https://ws.ega.or.th/ws/dopa/personal/profile/extra?CitizenID=<CitizenID>>

CitizenID สามารถใส่เลข 13 หลักของท่านเอง

ทดลองใช้ API

Developer Tools by Google Chrome
DHC – REST/HTTP API Client



DHC - REST/HTTP API Client

offered by www.sprintapi.com

★★★★★ (1147)

[Developer Tools](#)

173,668 users

<https://chrome.google.com/webstore/detail/dhc-resthttp-api-client/aejoelaoggembcahagimdiliamlcdmfm>

ทดลองใช้ API

The screenshot shows the DHC 0.8.1 web interface, a tool for testing APIs. The browser address bar shows the URL: `chrome-extension://aejoelaoggembcahagimdiliamlcdmfm/dhc.html`. The interface is divided into several sections:

- Header:** Displays "DHC 0.8.1" and navigation links for "Settings", "Help", and "About".
- Left Sidebar:** Contains a "My Drive" section with the text "NO SAVED DATA". At the bottom, there are icons for "Export" and "Import", and a prominent orange "Donate" button.
- Main Content Area:**
 - USE CASE:** A text input field containing "name", with "Save", "2 Code", and "Reset" buttons.
 - REQUEST:** A section for configuring the request. It includes a dropdown for "HTTP", a URI input field with the placeholder "type an URI", a dropdown for the method (currently "GET"), and a "Send" button.
 - HEADERS:** A section with a "+ set an authorization" button.
 - BODY:** A section with a "form" dropdown and a message: "XHR does not allow an entity-body for GET request."
 - RESPONSE:** A section displaying the message: "Not available, a request has not been sent yet."
- Bottom Panel:** Contains tabs for "HISTORY", "ASSERTIONS", "HTTP", and "DOCS". The "HISTORY" tab is active, showing the message "History is empty". To the right of these tabs are icons for window management and a "Filter" input field.

ทดลองใช้ API

chrome-extension://aejoelaoggembcahagimdiliamlcdmfm/dhc.html

DHC 0.8.1 Settings Help About

My Drive
NO SAVED DATA

Get Token

Save 2 Code Reset

REQUEST

HTTPS `ws.ega.or.th/ws/auth/validate?ConsumerSecret=Ea1hOqtJnG6&AgentID=3191000` GET Send

HEADERS

☒ Consumer-Key : 8e2ed330-200d-4d7e-a

RESPONSE

200 OK

Access-Control-Allow...
Connection: close
Content-Length: 50 Bytes
Content-Type: application/json; charset=utf-8
Date: 2015 Jul 27 14:32:17 -2m 22s

Result: "b8743466-6894-4620-b173-cb99af7d72cd"

4 จะได้ Result คือ Token

HISTORY

☐	2015 Jul 27 14:32:32	GET	https://ws.ega.or.th/ws/auth/validate?ConsumerSecret=Ea1h...	200 OK 28ms
☐	2015 Jul 27 14:30:10	GET	https://ws.ega.or.th/ws/auth/validate?ConsumerSecret=Ea1h...	200 OK 57ms
☐	2015 Jul 27 14:31:20	GET	https://ws.ega.or.th/ws/dopa/personal/profile/simple?CitizenI...	401 Unauthorized 2x 53ms
☐	2015 Jul 27 14:26:55	GET	https://ws.ega.or.th/ws/auth/validate?ConsumerSecret=Ea1h...	200 OK

Export Import

Donate

เริ่มจาก
1 ใส่ ConsumerKey
2 ใส่ ConsumerSecret,
AgentID ตามวงสีแดง
ใน URL เพื่อ Validate Key
3 และกด Send

ทดลองใช้ API

chrome-extension://aejoelaoggembcahagimdiliamlcdmfm/dhc.html

DHC 0.8.1 Settings Help About

My Drive
NO SAVED DATA

Get API

Save 2 Code Reset

REQUEST

HTTPS ws.ega.or.th/ws/dopa/personal/profile/simple?CitizenID=19100001793 GET Send

HEADERS

☒ Consumer-Key : 8e2ed330-200d-4d7e-a

☒ Token : b8743466-6894-4620-b

set an authorization

BODY

XHR does not allow an entity-body for GET request.

RESPONSE

200 OK elapsed time 406ms

Access-Control-Allow... *

Connection: close

Content-Length: 1 KBytes

Content-Type: application/json; charset=utf-8

HISTORY

☐	2015 Jul 27 14:53:52	GET	https://ws.ega.or.th/ws/dopa/personal/profile/simple?CitizenID...	200 OK 406ms
☐	2015 Jul 27 14:32:32	GET	https://ws.ega.or.th/ws/auth/validate?ConsumerSecret=Ea1hO...	200 OK 28ms
☐	2015 Jul 27 14:30:10	GET	https://ws.ega.or.th/ws/auth/validate?ConsumerSecret=Ea1hO...	200 OK 57ms
☐	2015 Jul 27 14:31:20	GET	https://ws.ega.or.th/ws/dopa/personal/profile/simple?CitizenID...	401 Unauthorized

Export Import

Donate

Filter

6 หลังจากได้ Token
นำมาใส่ตามรูป

7 และเลขประจำตัว
ประชาชนที่จะดึงข้อมูล
ใส่ใน URL Web Service

ทดลองใช้ API

chrome-extension://aejoelaoggembcahagimdiliamlcdmfm/dhc.html

DHC 0.8.1

Settings Help About

จะได้อข้อมูล Web Service ดังภาพ

RESPONSE

200 OK

HEADERS

Access-Control-Allow...*

Connection: close

Content-Length: 1 KBytes

Content-Type: application/json;charset=utf-8

Date: 2015 Jul 27 14:53:37 -15s

Nonce: 1437983617333

Server: GIIP/3.0

Via: HTTP/1.1 EGAWS

X-Powered-By: EGA/1.0

COMPLETE REQUEST HEADERS

form

BODY

elapsed time 406ms

formatted

```
{
  CitizenID: "000000000000",
  NameEN_Prefix: null,
  NameEN_FirstName: null,
  NameEN_MiddleName: null,
  NameEN_SurName: null,
  NameTH_Prefix: "นาย",
  NameTH_FirstName: "ชื่อจริง",
  NameTH_MiddleName: "",
  NameTH_SurName: "นามสกุลยาวมากมาย",
  SexID: 1,
  Sex: "ชาย",
  BirthDate: "25000101",
  PersonStatusID: 1,
}
```

Top Bottom Collapse Open 2Request Copy Download

HISTORY ASSERTIONS HTTP DOCS

2015 Jul 27 14:53:52	GET	https://ws.ega.or.th/ws/dopa/personal/profile/simple?CitizenID...	200 OK	406ms
2015 Jul 27 14:32:32	GET	https://ws.ega.or.th/ws/auth/validate?ConsumerSecret=Ea1hO...	200 OK	28ms
2015 Jul 27 14:30:10	GET	https://ws.ega.or.th/ws/auth/validate?ConsumerSecret=Ea1hO...	200 OK	57ms
2015 Jul 27 14:31:20	GET	https://ws.ega.or.th/ws/dopa/personal/profile/simple?CitizenID...	401 Unauthorized	

Export Import

Donate

Filter

ข้อควรระมัดระวังในการเรียกดูข้อมูล

“เรียกใช้ข้อมูลของบุคคลที่ได้รับอนุญาตเท่านั้น”

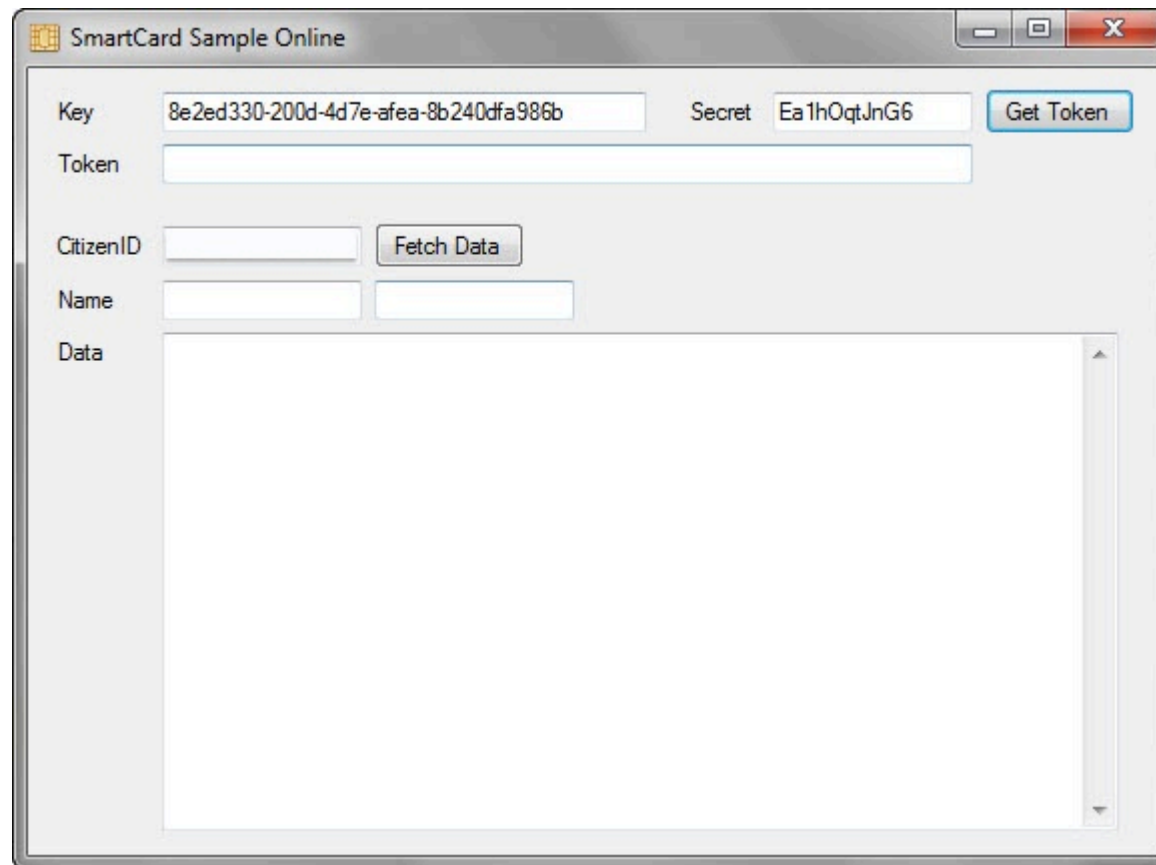
กรณีที่หน่วยงานใช้บัตรประชาชนของเจ้าของข้อมูลในการเรียกขอข้อมูลเพิ่มเติม ต้องมีการจัดเก็บประวัติการเรียกใช้ข้อมูล(Log) การใช้งาน EGA Web Service ในทุกๆ Transaction เช่น เจ้าหน้าที่ของหน่วยงานสืบบัตรประชาชนของตนเอง หรือทำการ login เข้าสู่ระบบ ก่อนการสืบบัตรประชาชนหรือกรอกเลข13หลักของเจ้าของข้อมูล(ประชาชน) เพื่อสามารถตรวจสอบว่าเจ้าหน้าที่คนใดเป็นผู้เรียกใช้ข้อมูลของเจ้าของข้อมูลนั้นๆ

ตัวอย่าง

Online Application

(แบบ Centralized Authentication และ Client Authentication)

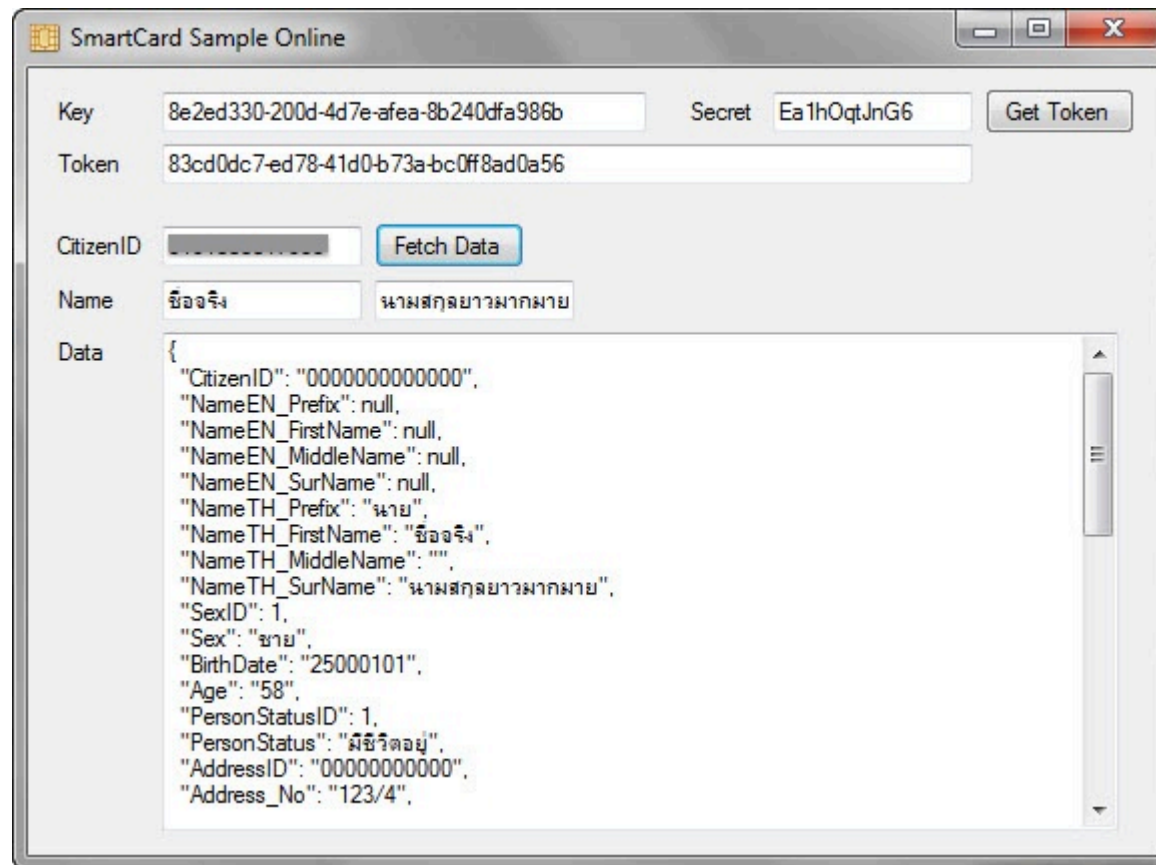
ดึงข้อมูลจาก API



The image shows a web application window titled "SmartCard Sample Online". It contains several input fields and buttons for interacting with an API. The "Key" field is pre-filled with "8e2ed330-200d-4d7e-afea-8b240dfa986b" and the "Secret" field is pre-filled with "Ea1hOqtJnG6". A "Get Token" button is located to the right of the secret field. Below these, there is a "Token" input field. Further down, there is a "CitizenID" input field and a "Fetch Data" button. Below the CitizenID field, there are two more input fields labeled "Name". At the bottom, there is a large "Data" output area with a vertical scrollbar.

Key	8e2ed330-200d-4d7e-afea-8b240dfa986b	Secret	Ea1hOqtJnG6	Get Token
Token				
CitizenID		Fetch Data		
Name				
Data				

ดึงข้อมูลจาก API



The image shows a web application window titled "SmartCard Sample Online". It contains several input fields and buttons for interacting with an API. The "Key" field is filled with "8e2ed330-200d-4d7e-afea-8b240dfa986b" and the "Secret" field with "Ea1hOqtJnG6". A "Get Token" button is next to the secret field. The "Token" field displays "83cd0dc7-ed78-41d0-b73a-bc0ff8ad0a56". Below this, there is a "CitizenID" field with a masked value and a "Fetch Data" button. The "Name" field is split into two parts: "ชื่อจริง" (First Name) and "นามสกุลยยาวมากมาย" (Long Surname). The "Data" section displays a JSON response from the API.

Key: 8e2ed330-200d-4d7e-afea-8b240dfa986b Secret: Ea1hOqtJnG6 Get Token

Token: 83cd0dc7-ed78-41d0-b73a-bc0ff8ad0a56

CitizenID: [Masked] Fetch Data

Name: ชื่อจริง นามสกุลยยาวมากมาย

Data:

```
{
  "CitizenID": "0000000000000",
  "NameEN_Prefix": null,
  "NameEN_FirstName": null,
  "NameEN_MiddleName": null,
  "NameEN_SurName": null,
  "NameTH_Prefix": "นาย",
  "NameTH_FirstName": "ชื่อจริง",
  "NameTH_MiddleName": "",
  "NameTH_SurName": "นามสกุลยยาวมากมาย",
  "SexID": 1,
  "Sex": "ชาย",
  "BirthDate": "25000101",
  "Age": "58",
  "PersonStatusID": 1,
  "PersonStatus": "มีสิทธิอยู่",
  "AddressID": "00000000000",
  "Address_No": "123/4",
}
```



Service and Support

Download

Contact



สำนักงานรัฐบาลอิเล็กทรอนิกส์ (องค์การมหาชน) (สโร.)
Electronic Government Agency (Public Organization) (EGA)

dev.egov.go.th

เป็น web สำหรับผู้ดูแลระบบ(Admin) และผู้พัฒนาของหน่วยงาน ประกอบด้วย

- ข้อมูลบริการ
- ขั้นตอนการพัฒนา และใช้งานระบบ
- การจัดการชุด Consumer Key ของหน่วยงาน
- ข้อมูลการใช้งาน API
- Log การใช้งาน API

ขั้นตอนการเข้าใช้งาน

1 ลงทะเบียนที่ egov.go.th

2 รอการอนุมัติ

dev.egov.go.th



ระบบจัดการบริการจากบัตรประชาชนสมาร์ตการ์ด

Web Service Authorization and Management System

หน้าแรก

ข้อมูลที่เกี่ยวข้อง

API ในการพัฒนา

ขอใช้งานเว็บเซอร์วิส



การพัฒนาระบบบริการ

การให้บริการ

ภาพรวมโครงการ

dev.egov.go.th



ระบบจัดการบริการจากบัตรประชาชนสมาร์ตการ์ด

Web Service Authorization and Management System

หน้าแรก

ข้อมูลที่เกี่ยวข้อง

API ในการพัฒนา

ขอใช้งานเว็บเซอร์วิส



API ในการพัฒนา

EGA App Center

Log

Download Key File

ega egov

Log

Download Key File

[DOPA] Proxy Available Checked

Key : [REDACTED]

CitizenVerification V2 :: Create Personal

Key : [REDACTED]

CitizenVerification V2 :: Get Public Personal

Key : [REDACTED]

EGA OTP Request

Key : [REDACTED]

EGA OTP Validate

Key : [REDACTED]

EGA Performance Test NET

Key : [REDACTED]

ega open id test

Log

Download Key File

dev.egov.go.th

EGA
e-Government Agency







หน้าแรก

ข้อมูลสมาชิก

สิทธิ์สมาชิก

รูปภาพหน้าแรก

Consumer Key

รายการคำร้อง 23

End-Point

เนื้อหาภายในเว็บไซต์ (CMS)

Log Monitoring

ค้นหาข้อมูล Consumer Key

ค้นหาข้อมูล Consumer Key

ชื่อ Consumer

กรอกเพื่อค้นหาด้วยชื่อ Consumer

องค์กร

[--- กรุณาเลือกองค์กร ---]

ค้นหา

ล้างผลการค้นหา

แสดง 10 รายการ

ก่อนหน้า 1 2 3 4 5 ... 18 ถัดไป

#	ชื่อ Consumer Key	องค์กร	ประเภทการใช้งาน	จำนวนการใช้งานปัจจุบัน	ระดับบริการ	คำสั่ง
1			ไม่จำกัด	0	ระดับที่ 0	 แก๊โซ  ขอใช้บริการ
2		สำนักงานรัฐบาลอิเล็กทรอนิกส์(องค์การมหาชน)	ไม่จำกัด	7843	ระดับที่ 0	 แก๊โซ  ขอใช้บริการ

dev.egov.go.th : API Usage Log

☐	Consumer Key	Service End-point	Date	Status	Source IP	คำสั่ง
☐	[REDACTED]	GET: [DOPA] House Profile Token: bf98511c-de49-4746-b229-664bd60404b1	10/07/2015 15:46		61.19.227.130	รายละเอียด
☐	[REDACTED]	GET: [DOPA] House Profile Token: 26e02d30-4f61-4724-a61a-3ff2df24f59f	10/07/2015 15:45		61.19.227.130	รายละเอียด
☐	[REDACTED]	GET: [DOPA] Personal Profile Full Token: 7a8ca54f-4a9b-4862-8700-60be6419caee	10/07/2015 15:45		61.19.227.130	รายละเอียด
☐	[REDACTED]	GET: [DOPA] Personal Profile Full Token: 7a8ca54f-4a9b-4862-8700-60be6419caee	10/07/2015 15:44		61.19.227.130	รายละเอียด

dev.egov.go.th : API Usage Log

ชื่อบริการ

[REDACTED]

Service End-point

GET: [DOPA] Personal Profile Full

TOKEN: 7a8ca54f-4a9b-4862-8700-60be6419caee

วันที่เข้าใช้งาน

10/07/2015 03:45

สถานะ

Passed

Source IP

61.19.227.130

Support and Download

Dev.egov.go.th/

เอกสารการอบรมด้านเทคนิค

เอกสารสำหรับวันอบรม
เอกสารการอบรม

เอกสารการดำเนินการก่อนการพัฒนาระบบ

1 TOR

[เอกสารตัวอย่าง TOR \(pdf\)](#)

[เอกสารตัวอย่าง TOR \(docx\)](#)

2 จัดทำ MOU กับกรมการปกครอง

[เอกสารแบบฟอร์ม MOU กับกรมการปกครอง](#)

[ตัวอย่างการเขียน เอกสารแบบฟอร์ม MOU กับกรมการปกครอง](#)

[เอกสารแบบฟอร์มการขอสิทธิ์ กับกรมการปกครอง \(pdf\)](#)

3 ลงทะเบียนใช้งาน API ผ่าน สรอ.

[เอกสารแบบฟอร์มลงทะเบียนใช้งาน API สรอ. \(pdf\)](#)

[เอกสารตัวอย่างการกรอก แบบฟอร์มลงทะเบียนใช้งาน API สรอ.](#)

Contact

สอบถามรายละเอียดเพิ่มเติมได้ท

contact@ega.or.th หรือ
โทร. **0-2612-6060**





สอบถามรายละเอียดเพิ่มเติมได้ที่
contact@ega.or.th หรือ โทร. **0-2612-6060**



สำนักงานรัฐบาลอิเล็กทรอนิกส์ (องค์การมหาชน) (สรอ.)
Electronic Government Agency (Public Organization) (EGA)