**เว็บแอปพลิเคชันสำหรับการจัดการทรัพยากรและค่าใช้จ่ายใน AWS**

**(WEB APPLICATION FOR RESOURCE AND COST MONITORING IN AWS)**

**โดย**

**นายชนายุส หนูเอียด**

**นายภูชิต รุ่งระวิ**

**อาจารย์ที่ปรึกษา**

**ผู้ช่วยศาสตราจารย์ ดร.ลภัส ประดิษฐ์ทัศนีย์**

**ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต**

**สาขาวิชาเทคโนโลยีสารสนเทศ**

**คณะเทคโนโลยีสารสนเทศ**

**สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง**

**ภาคเรียนที่ 1 ปีการศึกษา 2564**

**WEB APPLICATION FOR RESOURCE AND COST MONITORING IN AWS**

**CHANAYUS NUAIAD**

**POOHCID RUGRAWI**

**A PROJECT SUBMITTED IN PARTIAL FULFILLMENT**

**OF THE REQUIREMENT FOR THE DEGREE OF**

**BACHELOR OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY**

**FACULTY OF INFORMATION TECHNOLOGY**

**KING MONGKUT’S INSTITUTE OF TECHNOLOGY LADKRABANG**

**1/2021**

**COPYRIGHT 2021**

**FACULTY OF INFORMATION TECHNOLOGY**

**KING MONGKUT’S INSTITUTE OF TECHNOLOGY LADKRABANG**

**หัวข้อโครงงาน**       เว็บแอปพลิเคชันสำหรับการจัดการทรัพยากรและค่าใช้จ่ายใน AWS

**นักศึกษา** นายชนายุส หนูเอียด รหัสนักศึกษา 61070034

      นายภูชิต รุ่งระวิ รหัสนักศึกษา 61070166

**ปริญญา**       วิทยาศาสตรบัณฑิต

**สาขาวิชา** เทคโนโลยีสารสนเทศ

**ปีการศึกษา** 2562

**อาจารย์ที่ปรึกษา** ผู้ช่วยศาสตราจารย์ ดร.ลภัส ประดิษฐ์ทัศนีย์

**บทคัดย่อ**

การประมวลผลระบบคลาวด์เป็นเทคโนโลยีที่มีความสำคัญต่อการติดต่อสื่อสารผ่านอินเทอร์เน็ตในการกระจายข้อมูลไปยังส่วนต่าง ๆ ของโลกโดยที่มีการจัดการที่ง่ายและขยายตัวตามความต้องการได้รวดเร็ว ซึ่งจะมีผู้ให้บริการติดตั้งศูนย์กลางข้อมูลกระจายไปทั่วโลก การเรียนรู้โครงสร้างพื้นฐานของระบบสารสนเทศในรูปแบบคลาวด์เป็นการเรียนรู้ที่ง่าย ประหยัด และสามารถไปต่อยอดไปใช้งานในธุรกิจจริงได้ง่าย

Amazon Web Services เป็นหนึ่งในผู้ให้บริการคลาวด์และเป็นแพลตฟอร์มคลาวด์ที่นำมาใช้ในวิชาการพัฒนาคลาวด์แอปพลิเคชันระดับองค์กรและวิชาเทคโนโลยีกลุ่มเมฆเบื้องต้นของคณะเทคโนโลยีสารสนเทศ สจล. โดยมี ผศ. ดร.ลภัส ประดิษฐ์ทัศนีย์ เป็นอาจารย์ประจำวิชาที่ได้พบเจอปัญหาในการสอนวิชาคลาวด์ในส่วนที่เป็นข้อจำกัดของ AWS Management Console ที่ให้การใช้งาน IAM User ให้กับกลุ่มโปรเจคนักศึกษาไปใช้งาน โดยปัญหาหลัก ๆ ที่พบเจอคือการเข้าถึงและจัดการทรัพากรหรือการบริการต่างๆที่นักศึกษาสร้างไว้ทั้งหมดค่อนข้างใช้เวลานานและทรัพยากรที่สร้างขึ้นไม่มีการระบุว่าเป็นของ IAM User คนไหน จึงเกิดเป็นโปรเจคเว็บแอปพลิเคชันสำหรับการจัดการทรัพยากรที่จะมีการวางระบบเบื้องหลังในการติดตามการกระทำของ IAM User มีการระบุเจ้าของทรัพยากรที่สร้างขึ้น การลบทรัพยากรที่ง่าย และระบบเสริมที่มาชดเชยในส่วนที่ AWS Management Console ไม่มีที่จะช่วยอาจารย์ประจำวิชาและนักศึกษาในการลดปัญหาในการทำงานได้ดียิ่งขึ้น

**Project Title**      Web Application for Resource and Cost Monitoring in AWS

**Students** Mr. Chanayus Nuiad Student ID 61070034

      Mr. Poohcid Rugravi Student ID 61070166

**Degree**       Bachelor of Science

**Program**  Information Technology

**Academic Year** 2021

**Advisor** Assoc. Prof. Dr. Lapat Praditthasanee

**ABSTRACT**

Cloud computing is an essential technology for internet communication and distributing information to different parts of the world. Cloud computing easy management and rapid expansion on demand, which will have service providers spread around the world. The infrastructure of information systems in the cloud is easy to learn, economical, and easily extended to use in real business.

Amazon Web Services is one of the cloud providers and is a cloud platform used in enterprise cloud application development, which is an introductory cloud technology course of the Faculty of Information Technology, KMITL. by Asst. Prof. Dr. Lapat Praditthasanee is an instructor who has encountered difficulties teaching cloud subjects in the AWS Management Console limitations. The main problem encountered is that accessing and managing all student-created resources or services takes a long time and resources to create. It is not specified which IAM User belongs to, This is Why we develop a web application for resource management in AWS with have a background event tracking system for IAM Users. The owner of the created resource is specified. Easy resource removal and a complementary system that compensates for parts that the AWS Management Console does not have to help course teachers and students reduce problems in working better.

**บทที่ 1**

**บทนำ**

**1.1 ความเป็นมาของปัญหา**

การที่มีนักศึกษาในการเรียนวิชาที่เกี่ยวข้องกับระบบ Cloud Computing เป็นจำนวนมากจึงส่งผลให้อาจารย์ประจำวิชาเข้าถึงและจัดการ resource หรือ service ต่างๆที่นักศึกษาสร้างไว้ทั้งหมดค่อนข้างใช้เวลานานและยากลำบาก ทางกลุ่มของเราจึงต้องการทำโครงงานเกี่ยวกับเว็บแอปพลิเคชันสำหรับจัดการ Resource และ Service ต่างๆ ใน AWS เพื่อให้สามารถเข้าถึงและติดตามได้สะดวกมากยิ่งขึ้น

**1.2 วัตถุประสงค์**

1. **เ**พื่อเพิ่มความรวดเร็วและความสะดวกสะบายให้กับอาจารย์ ในจัดการ resource ต่างๆ ใน AWS ที่นักศึกษาสร้างขึ้นในวิชาเรียนของคณะเทคโนโลยีสารสนเทศสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
2. ลดข้อผิดพลาดในการบริหารจัดการระบบ ยกต้ออย่างเช่น กรณีที่นักศึกษาลืมใส่ tag ให้กับ resource เพื่อมั่นใจได้ว่า resource ที่ถูกสร้างขึ้นทุกที่เป็นของกลุ่มไหนและสามารถลบได้ตามกรอบ policy ที่กำหนด
3. เพิ่มฟังก์ชันในส่วนที่ AWS management console ไม่มี สำหรับอาจารย์ผู้รับผิดชอบวิชา เช่น ฟังก์ชันการดูค่าใช้จ่าย ฟังก์ชันในการติดตั้งและตั้งค่า resources ที่นอกเหนือในการเรียนในวิชา Cloud

**1.3 ขอบเขตของระบบ**

โปรเจคจัดทำขึ้นเพื่อใช้สำหรับวิชา Cloud-Based Enterprise Application Development หรือวิชาที่เรียนทางด้าน Cloud Computing โดยใช้บริการของ AWS เท่านั้น เพื่อแก้ไขปัญหาของอาจารย์ประจำวิชาและนักศึกษาที่เรียนในวิชานี้

**1.4 ประโยชน์ที่คาดว่าจะได้รับ**

1. อาจารย์ประจำวิชามีความสะดวกสบายในการจัดการ Resources
2. นักศึกษาที่เรียนในวิชา Cloud วางแผนการคำนวณค่าใช้จ่ายได้ง่ายขึ้น
3. อาจารย์ประจำวิชาที่เป็นเจ้าของ Account ไม่ต้องเสียค่าใช้จ่ายส่วนเกินที่ได้รับมา
4. นักศึกษาสามารถทดลองการใช้ resource ได้เต็มที่เพื่อให้โปรเจคสำเร็จลุร่วง
5. คณะผู้จัดทำได้ความรู้และประสบการณ์ในการทำ Infrastructure As Code ของระบบ AWS

**1.5  ศัพท์นิยามเฉพาะที่ใช้ในโครงงาน**

1. SDK ( Software Development Kit ) คือ ชุดเครื่องมือที่ใช้สำหรับการพัฒนาซอฟต์แวร์ที่ถูกสร้างโดยผู้พัฒนา เพื่อให้ผู้ใช้ซอฟต์แวร์สามารถนำเครื่องมือนี้ไปสร้างหรือพัฒนางานของตนเองได้
2. AWS Resource คือ ทรัพยากรต่างๆที่มีให้บริการใน Amazon web service เช่น AWS Lambda, AWS CloudTrail และ AWS S3 เป็นต้น
3. IAM User คือ ผู้ใช้ในส่วนของนักศึกษาที่ใช้งาน AWS Management Console ของ Account

อาจารย์ผู้สอน

**บทที่ 2**

**การทบทวนวรรณกรรม**

**2.1 การบริการและทรัพยากรใน AWS เพื่อใช้ในการสร้างแท็กแบบอัตโนมัติ**

  
  
**รูปที่ 2.1** สัญลักษณ์ AWS CloudTrail

**2.1.1 AWS CloudTrail**

เป็น Service ที่จะเป็น log ทุกการกระทำของ IAM user ทั้งหมด โดยการเก็บ log จะขึ้นอยู่กับ policy ที่ตั้งไว้ เมื่อ user ใช้งานตาม policy ที่กำหนดจะมีการเก็บ log โดย CloudTrail ลง S3 bucket



**รูปที่ 2.2** สัญลักษณ์ AWS CloudWatch

**2.1.2 AWS CloudWatch**

เป็น Service ที่จะคอยตรวจสอบ รวบรวมข้อมูลและการดำเนินการต่างๆ เห็นภาพรวมของทรัพยากร แอปพลิเคชัน และบริการต่างๆ ของ AWS ใน 1 region

  
**รูปที่ 2.3** สัญลักษณ์ AWS Lambda

**2.1.3 AWS Lambda**

เป็น Service ที่ทำให้ผู้ใช้สามารถรันโค้ดโดยที่ไม่จำเป็นต้องมี Server เป็นของตัวเอง

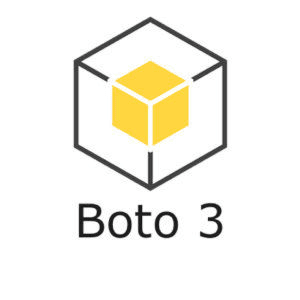
ซึ่งส่งผลให้ผู้ใช้ไม่ต้องทำการตั้งค่าต่างๆเกี่ยวกับ Server ผู้ใช้เพียงแค่นำโค้ดขึ้นไปที่ AWS Lambda

โค้ดจะสามารถทำงานได้ทันที

  
  
  
**รูปที่ 2.4** สัญลักษณ์ AWS IAM

**2.1.4 IAM**

เป็น Service ที่ช่วยจัดการสิทธิ์การเข้าถึงทรัพยากรและบริการของ AWS อย่างปลอดภัย สามารถสร้างและจัดการผู้ใช้และกลุ่ม AWS ได้โดยใช้ IAM และใช้สิทธิ์เพื่ออนุมัติหรือปฏิเสธไม่ให้ผู้ใช้เข้าถึงทรัพยากร AWS ได้

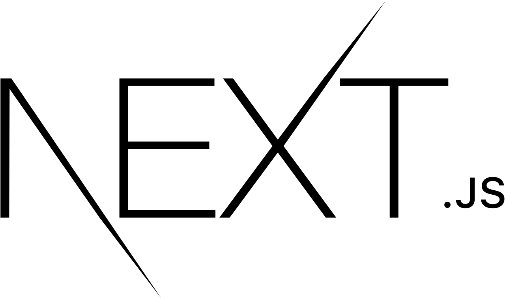


**รูปที่ 2.5** สัญลักษณ์ Boto 3

**2.1.3 Boto 3**

Boto 3 เป็น AWS SDK สำหรับภาษา Python ที่ช่วยให้นักพัฒนาสามารถพัฒนาซอฟต์แวร์ให้ทำงานร่วมกับ Amazon service ได้สะดวกมากขึ้น

**2.2 เครื่องมือที่ใช้ในการพัฒนาซอฟต์แวร์**

 **2.2.1 เครื่องมือสำหรับการพัฒนาเว็บแอปพลิเคชัน**

**รูปที่ 2.6** สัญลักษณ์ NEXT JS

**2.2.1.1 Next JS**

Next Js เป็น React Framework พัฒนาโดย Vercel ที่ใช้สำหรับการพัฒนาเว็บแอปพลิเคชันทั้งในฝั่ง Front-end โดยจุดเด่นหลักๆของ Next JS คือ ทาง Vercel ได้นำเครื่องมือต่างๆที่จำเป็นมาไว้ใน Next JS ให้แล้วเช่น การทำ Routing, การทำ API และมีการ Optimize ทำให้มี Performance ที่ดียิ่งขึ้น

รูปภาพประกอบด้วย ข้อความ, สัญลักษณ์

คำอธิบายที่สร้างโดยอัตโนมัติ

**รูปที่ 2.7** สัญลักษณ์ Mongodb

**2.2.1.2 MongoDB**

MongoDB เป็น database โดยมีพื้นฐานเป็น NoSQL จึงไม่มีความสัมพันธ์ของข้อมูล

เหมือนกับ database แบบ SQL ซึ่ง MongoDB จะบันทึกข้อมูลลงในไฟล์รูปแบบ JSON

ซึ่งต่างจาก database SQL ที่บันทึกข้อมูลลงตาราง

**บทที่ 3**

**วิธีการดำเนินการวิจัย**

**3.1 การวิเคราะห์และออกแบบระบบ**

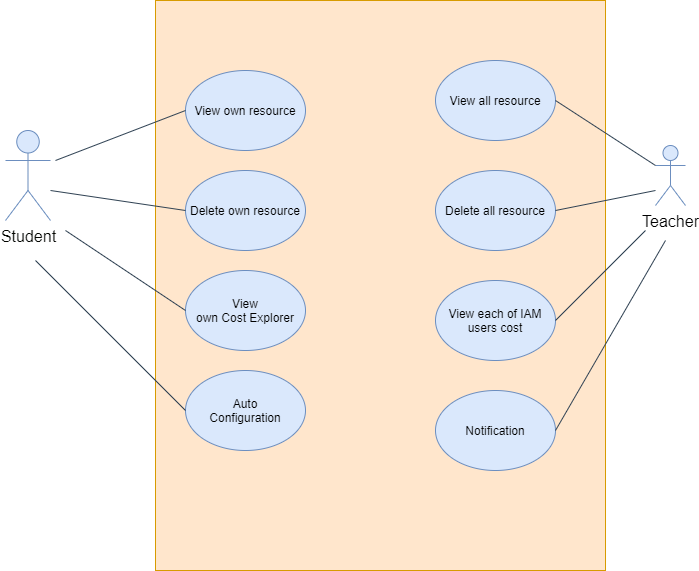
**3.1.1 วิเคราะห์ปัญหา**

* ทรัพยากรที่ถูกสร้างขึ้นจะไม่มีการระบุว่าเป็นของผู้ใช้งานคนไหน
* การดูแลทรัพยากรทุกอย่างใน AWS เป็นเรื่องที่ยุ่งยาก
* การจะลบทรัพยากรต้องใช้ระยะเวลานานถึงจะลบหมด
* IAM User ไม่มีหน้าการคำนวนค่าใช้จ่ายบน AWS Management Console

**3.1.2 ออกแบบความต้องการของระบบ**

* วางระบบสร้างแท็กอัตโนมัติเมื่อมีการสร้างทรัพยากรนั้น ๆ บน AWS ที่จะมีการระบุชื่อของกลุ่มที่สร้าง
* สร้างเว็บแอปพลิเคชันที่มีฟังก์ชันในการดูข้อมูลและลบทรัพยากรให้สะดวกมากยิ่งขึ้น และมีการคำนวนค่าใช้จ่ายของ IAM User นักศึกษา
* อาจารย์สามารถดูทรัพยากรของทุกกลุ่มที่สร้างขึ้นได้โดยจะแสดงผลให้เข้าใจได้ง่ายและสามารถสั่งลบทรัพยากรที่สร้างขึ้นจาก IAM User นักศึกษาได้ทั้งหมดได้ในขั้นตอนเดียว
* เว็บแอปพลิเคชันมีระบบสนับสนุนเพิ่มเติมจาก AWS Management Console เช่น การแจ้งเตือนการสร้าง VPC เกินจำนวน และ internet gateway เกินจำนวน เป็นต้น

**3.1.3 Use Case Diagram**

****

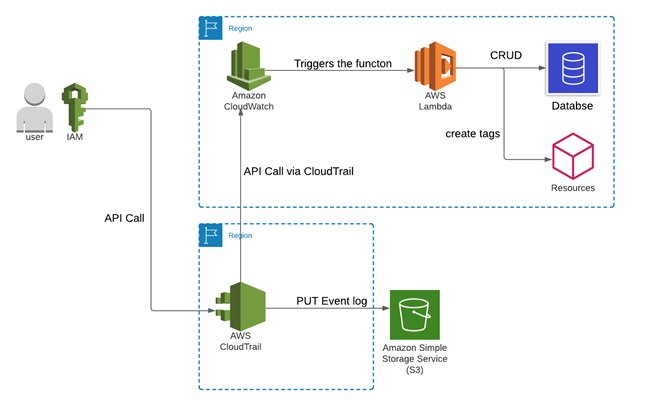
**รูปที่ 3.1** Use Case diagram

**3.2 แผนดำเนินงาน**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ขั้นตอนการดำเนินงาน | ระยะเวลาการดำเนินงาน | | | | | | | | | | | | | | | |
| สิงหาคม 2564 | | | | กันยายน 2564 | | | | ตุลาคม 2564 | | | | พฤศจิกายน 2564 | | | |
| 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| 1.ศึกษาและทดลองทำ auto-tagging สำหรับ AWS Resource |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2.ศึกษาและทดลองการทำ AWS SDK (ใส่ส่วนของการเรียกดูข้อมูล s3 ec2 และลบ instance) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3. ออกแบบ UI หรือ prototype ของเว็บ |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4. สร้าง API สำหรับการดึงข้อมูล Resource และลบ Resource |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5. เริ่มพัฒนาเว็บแอพ |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6.เริ่มทำระบบแจ้งเตือน เมื่อมีการสร้าง Resource เกินความจำเป็น |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7. เริ่มศึกษาเรื่องการคำนวนค่าใช้จ่ายในกรณีที่แยก Resource ตาม iAM User |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**3.3 หลักการทำงานของระบบ**

**3.3.1 หลักการทำงานของระบบ Auto Tag**



**รูปที่ 3.2** Auto-tagging solution workflow

เมื่อ IAM User ทำการสร้าง Resource หรือ ใช้งาน service ต่างๆ ระบบจะทำการเรียกใช้งาAWS CloudTrail เพื่อทำการตรวจสอบ Event ที่ IAM User ได้กระทำต่อระบบของ AWS  และเก็บข้อมูล Event นั้นลงใน AWS S3 และจากนั้น AWS CloudTrail จะทำการเรียกใช้งาน

**3.3.2 ลำดับกาวางระบบ Auto Tag เบื้องต้น**

**3.3.2.1 เพิ่มโค้ด Boto 3 ลงใน AWS Lambda**

รูปภาพประกอบด้วย ข้อความ

คำอธิบายที่สร้างโดยอัตโนมัติรูปภาพประกอบด้วย ข้อความ

คำอธิบายที่สร้างโดยอัตโนมัติ

**รูปที่ 3.4** โค้ด Boto 3

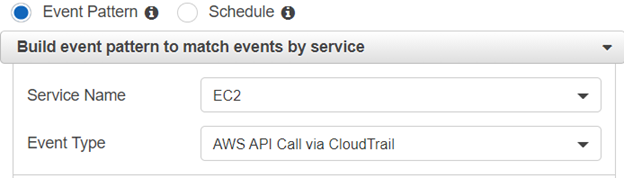
**รูปที่ 3.3** โค้ด Boto 3

**รูปภาพประกอบด้วย ข้อความ

คำอธิบายที่สร้างโดยอัตโนมัติ3.3.2.2 เปิดใช้งาน AWS CloudTrail**

**รูปที่ 3.5** อธิบายการเปิดใช้งาน AWS CloudTrail

**3.3.2.3 เพิ่ม Event Rule ใน AWS CloudWatch**

****

**รูปที่ 3.6** อธิบายการเพิ่ม Event Rule ใน AWS CloudWatch

**รูปภาพประกอบด้วย ข้อความ

คำอธิบายที่สร้างโดยอัตโนมัติ**เลือก service ที่จะดัก event เช่น EC2 และเลือก Event Type เป็น AWS API Call via CloudTrail หมายความว่าดัก event เมื่อ user กระทำต่อ service ผ่านทาง SDK หรือ AWS Management Console เช่น การสร้าง แก้ไข และ ลบ

**รูปที่ 3.7** อธิบายการเพิ่ม Event ที่ต้องการจะดักผ่าน JSON

ใส่ event ที่ต้องการจะดักใน key “eventName” โดย event ทั้งหมดสามารถหาข้อมูลเพิ่ม

**รูปภาพประกอบด้วย ข้อความ

คำอธิบายที่สร้างโดยอัตโนมัติ3.3.2.4 ใส่ Targets trigger function ใน CloudWatch**

**รูปที่ 3.8** อธิบายการใส่ Targets trigger function ใน CloudWatch

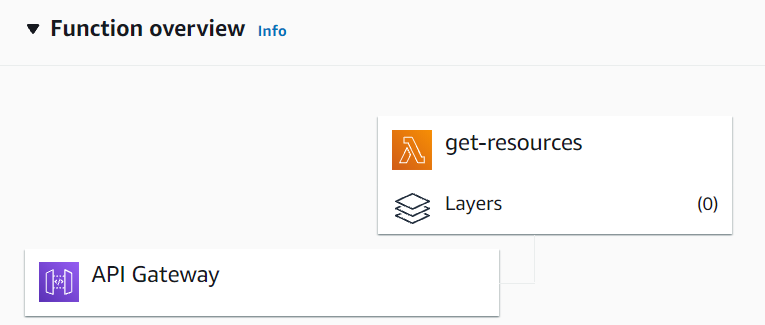
เลือก Lambda function ที่สร้างในขั้นตอนที่ 3.2.2.1

**3.3.2 การดึงข้อมูลทรัพยากร**

**รูปภาพประกอบด้วย ข้อความ

คำอธิบายที่สร้างโดยอัตโนมัติ 3.3.2.1 ฟังก์ชันโค้ด AWS SDK (boto3)**

**รูปที่ 3.9** ฟังก์ชันโค้ด AWS SDK (boto3)

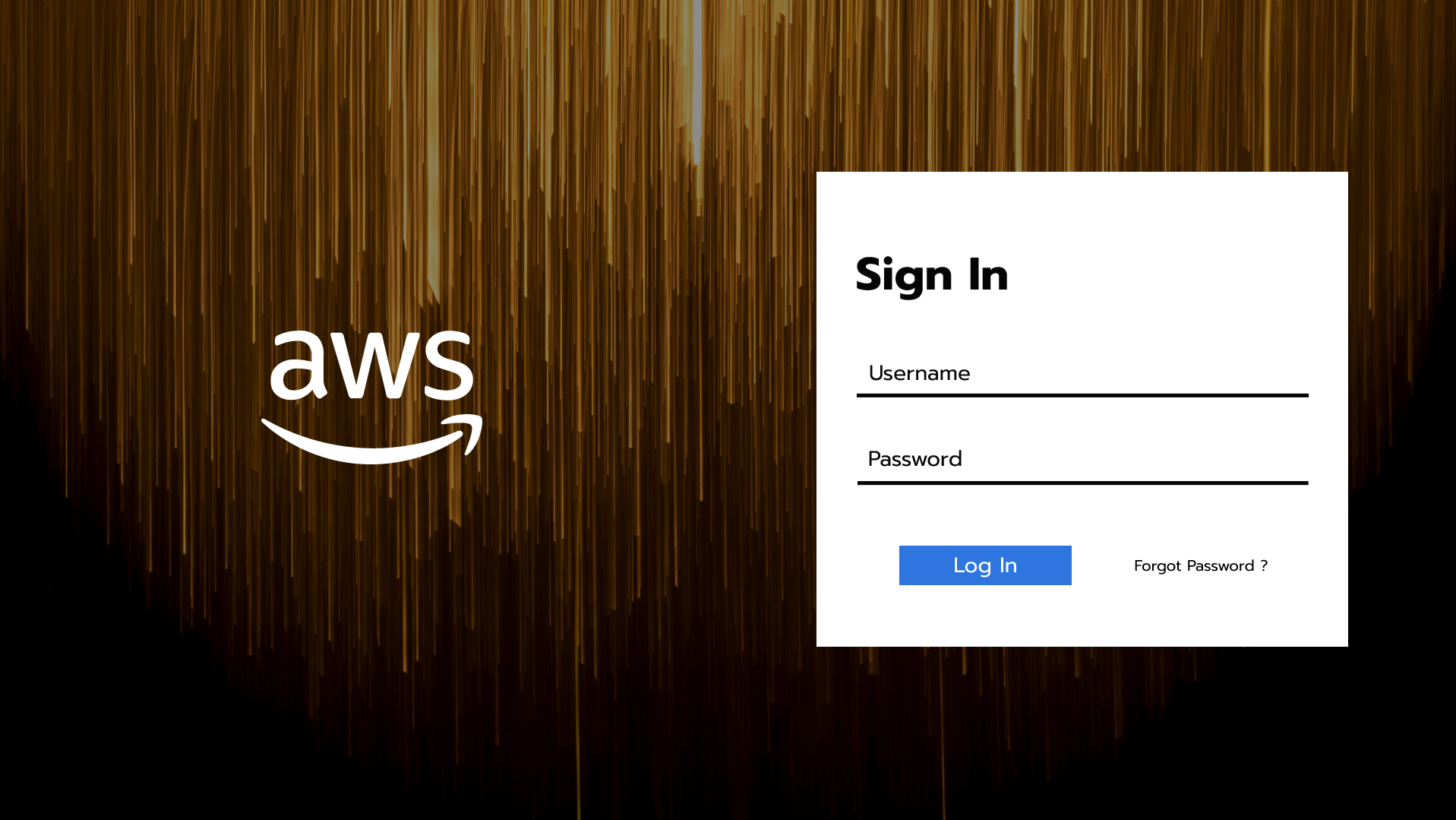
**3.3.2.2 สร้าง API Gateway เพื่อเรียกใช้งานฟังก์ชัน**

**รูปที่ 3.10** การสร้าง API Gateway เพื่อเรียกใช้งานฟังก์ชัน**รูปภาพประกอบด้วย ข้อความ

คำอธิบายที่สร้างโดยอัตโนมัติ** (1)

**รูปที่ 3.11** การสร้าง API Gateway เพื่อเรียกใช้งานฟังก์ชัน (2)

**3.3.3 prototype**

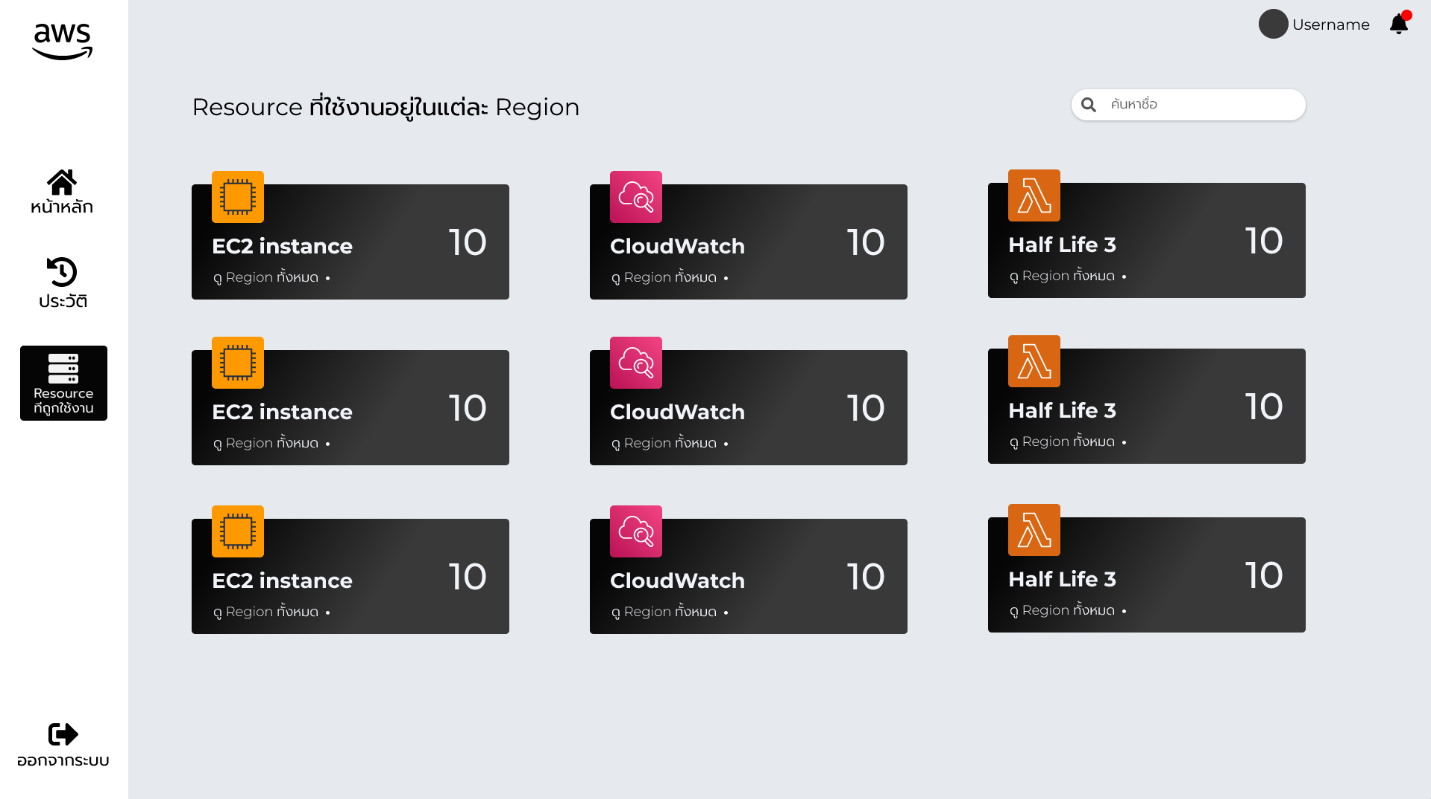
****

**รูปที่ 3.12 หน้า Log in**

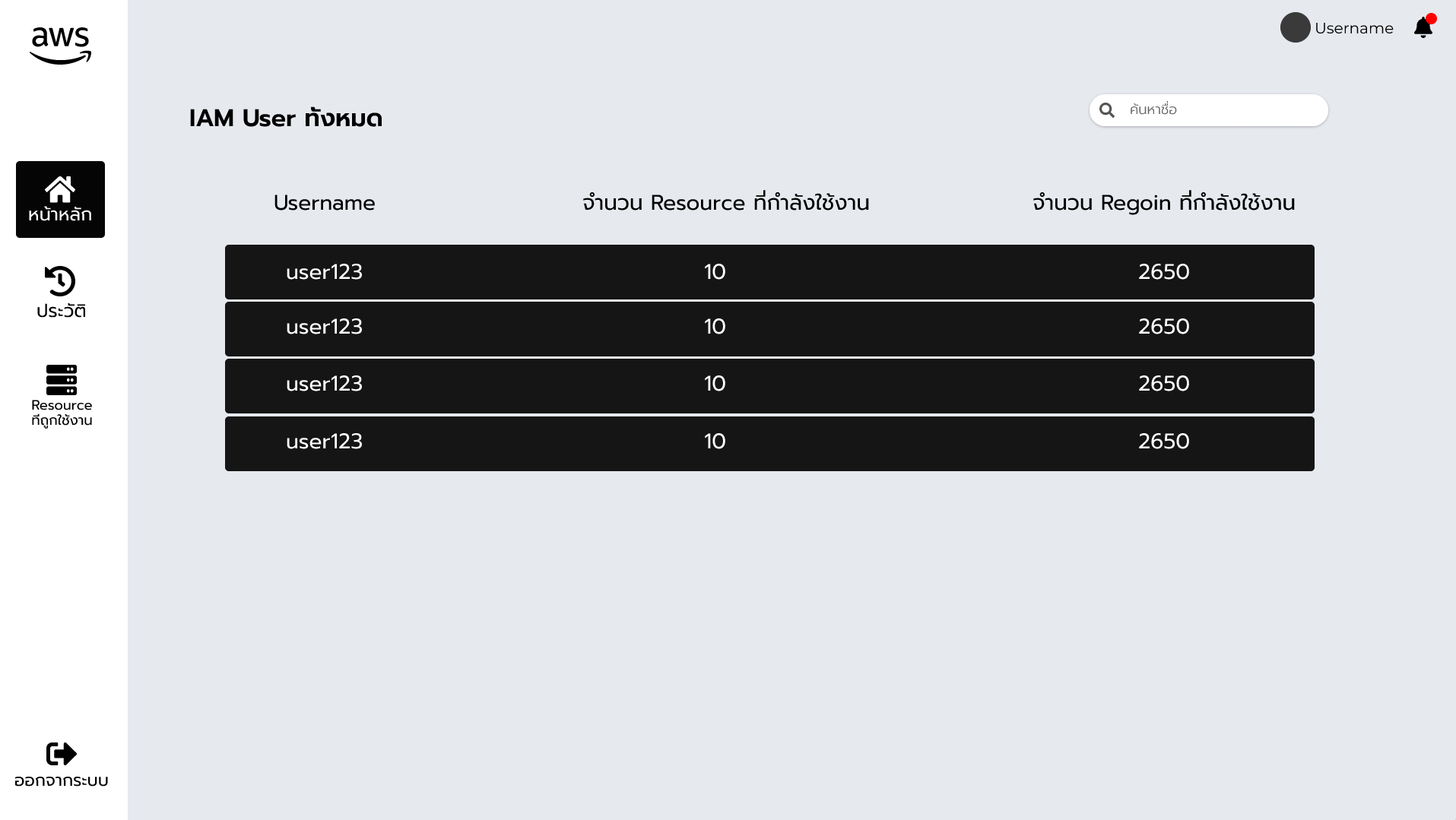
**รูปภาพประกอบด้วย ข้อความ

คำอธิบายที่สร้างโดยอัตโนมัติ**

**รูปที่ 3.**13 หน้า Dashboard ของอาจารย์ผู้สอน

****

**รูปที่ 3.14** หน้าแสดงผล Resource ที่กำลังใช้งานในแต่ละ Region

****

**รูปที่ 3.15** หน้าแสดง IAM User ทั้งหมด

****

**รูปที่ 3.16** หน้ารายละเอียดของ Resource

**รูปภาพประกอบด้วย ข้อความ

คำอธิบายที่สร้างโดยอัตโนมัติ**

**รูปที่ 3.17** หน้าแสดงผล IAM User ที่ใช้งาน Resource