



LABORATORY ON CLOUD

CONTENTS

1

Introduction

2

Architecture
Cloud

3

Solution

An aerial, grayscale photograph of a dense urban skyline, likely Hong Kong, featuring numerous skyscrapers and a harbor. A blue square is overlaid on the left side of the image, containing a white number '1'.

1

Introduction

บริการ คลาวด์ คือ

บริการที่ผู้ใช้งานสามารถเลือกใช้ซอฟต์แวร์ และทรัพยากรด้านคอมพิวเตอร์ของผู้ให้บริการ เพื่อนำมาใช้งาน โดยที่เราไม่ต้องลงทุนซื้อทั้ง Hardware และ Software ไม่ต้องวางระบบ network เอง เพราะ ผู้ให้บริการเป็นผู้ดูแลระบบ network ให้ผู้ใช้งาน เป็นการลดต้นทุนและลดความยุ่งยากในการอัปเดตระบบอีกด้วย และผู้ใช้งานสามารถจัดการข้อมูลบนคลาวด์ได้หลากหลายอุปกรณ์ที่สามารถเข้าถึงอินเทอร์เน็ตได้ เช่น โทรศัพท์มือถือ, Tablet, Notebook

บริการ คลาวด์ คือ

ในมุมมองของธุรกิจ บริการคลาวด์จะเป็นบริการที่ยืดหยุ่นต่อการใช้งาน รองรับการขยายตัวของธุรกิจ ค่าใช้จ่ายสำหรับระบบ cloud เป็นการจ่ายตามการใช้งานจริง ซึ่งตอบโจทย์สำหรับการใช้บริการในระยะเวลาสั้นๆ ในการใช้เท่าไร จ่ายเท่านั้น หากความต้องการใช้งานมีมากขึ้นก็สามารถซื้อเพิ่มได้โดยไม่ต้องอัปเดตระบบให้ยุ่งยาก และยังช่วยให้เกิดความยืดหยุ่นในการใช้จ่ายด้านไอทีที่ตอบโจทย์ความต้องการทางธุรกิจได้มากยิ่งขึ้น รวมถึงการตัดปัญหาค่าใช้จ่ายแฝง เช่น ค่าไฟฟ้าสำหรับระบบ hardware การจ้างผู้ดูแลระบบมาดูแลทั้งระบบ network และ hardware อีกด้วย ดังนั้น ประโยชน์ที่ธุรกิจขนาดเล็ก ขนาดกลาง ขนาดใหญ่ รวมถึงสถาบันการศึกษาจะได้รับ คือ ลดต้นทุนทั้งทางด้านงบประมาณ ทรัพยากรบุคคลทางด้านไอทีของผู้ใช้งานและลดความยุ่งยากทางด้านไอที จึงสมควรใช้บริการคลาวด์



2

Architecture Cloud

Architecture Cloud

Cloud Foundry, High Scalability and High Availability

Web UI

Web UI

View

Controller

Model

JSON

Node.js

Node.js

REST

Express.js

mongojs

JSON

MongoLab DBaaS

REST

DB

รูปแบบการทำงาน

- 1) Client-server architecture: Client ไม่จำเป็นต้องรู้อะไรเกี่ยวกับ Business logic ภายใน ไม่มีหน้าที่เกี่ยวกับการจัดเก็บข้อมูล ส่วน Server มีหน้าที่เก็บ Resource และไม่จำเป็นต้องรู้อะไรเกี่ยวกับ UI Frontend หรือสถานะของผู้เรียก
- 2) Statelessness: ส่ง Request รับ Response จาก Server แล้วเลิก
- 3) Cache-ability: สามารถ cache response ได้ การ Response จะต้องสามารถกำหนดได้ว่าจะ Cache หรือไม่ เพื่อป้องกันไม่ให้ User หรือ Client ได้รับข้อมูลเก่า
- 4) Layered system: ปกติ Client ไม่รู้ว่าที่ทำการเชื่อมต่อนั้น ได้เชื่อมต่อโดยตรงกับ Server ปลายทาง หรือไปยังตัวกลางอื่น ๆ ระหว่างทาง, Server ตัวกลางควรสามารถปรับปรุงความสามารถในการขยายระบบได้ โดยการใช้งานการทำ Load balance

รูปแบบการทำงาน

5) Uniform interface: ถือเป็นข้อสำคัญจะที่แยกระหว่าง REST API และ Non-REST API มันแสดงให้เห็นถึงวิธีการที่จะคุยกับ Server โดยไม่คำนึงถึงประเภทของอุปกรณ์ หรือประเภทของ application

Uniform interface ได้แยกออกไปอีก 4 อย่าง

- Resource-Based: เช่น API/users
- Manipulation of Resources Through Representations: เช่น User get user_id หรือ Request list of users แล้วทำการ Delete หรือ Modify user
- Self-descriptive Messages: แต่ละ Message มีข้อมูลเพียงพอที่จะนำมาอธิบายวิธีการ Process message เพื่อให้ Server ทำการวิเคราะห์ได้ง่าย
- Hypermedia as the Engine of Application State (HATEOAS): จำเป็นต้องมี Links สำหรับทุก ๆ Response เพื่อให้ Client สามารถค้นหาได้ง่าย

รูปแบบการทำงาน

- 6) สามารถควบคุมสิทธิการใช้งานในระบบผ่านกระบวนการ User Authenticate (ระบบ Log On) และกำหนดสิทธิได้มากถึง 12 สิทธิในแต่ละกลุ่มผู้ใช้งาน
- 7) มีระบบการรายงานผล Web-Based Reporting รองรับการรายงานผล
- 8) มีเครื่องมือในการบริหารจัดการ (Administration Tool) ผ่านทาง Web base
- 9) Worklist Entry สำหรับการตรวจอย่างน้อยที่สุดต้องประกอบด้วย ชื่อผู้ป่วย, รหัสผู้ป่วย, การตรวจ, วันเดือนปี และเวลาในการตรวจ
- 10) ระบบจะต้องสามารถเรียกการตรวจเก่าจาก Storage System และ Archive ได้
- 11) โปรแกรมระบบจัดการแบบ Web-Base (RIS : WEB-BASED RADIOLOGY WORKFLOW MANAGER)
- 12) สามารถกำหนดผู้รับผิดชอบแต่ละงานได้
- 13) รองรับมาตรฐาน HL7 สำหรับการแลกเปลี่ยนข้อมูล



3

Solution/Specification

คุณสมบัติ Application

1. ฐานข้อมูลเก็บบนระบบ cloud
2. ตัวโปรแกรม Run on web base สามารถเปิดได้จากอุปกรณ์ทุกแพลตฟอร์ม
3. แบ่งการรับส่งข้อมูลอย่างน้อย 3 Node โดยหากมี Node ใดล่มสามารถ เปลี่ยนไปใช้ Node ที่เหลือได้โดยอัตโนมัติ เพื่อป้องกันปัญหาหาคอขวดและระบบล่ม
4. สามารถรองรับการส่งข้อมูลผลตรวจและรายการตรวจใหม่ๆที่เพิ่มขึ้นได้
5. มีการกำหนดผู้ใช้งานระบบโดยให้มี Username Password และการยืนยันตัวตนทุกครั้งในการเข้าดูผลและมีการบันทึก log file ไว้ด้วยว่าใครเข้ามาดูผลของใครเวลาไหนจำนวนกี่ครั้งนอกจากนี้ควรมีการแบ่งระดับการเข้าถึงข้อมูลเป็นระดับต่างๆเช่น ผู้ใช้งานทั่วไป ผู้ใช้งานระดับผู้บริหารและผู้ดูแลรักษาระบบ

คุณสมบัติ Application

6. มีการกำหนด API กลางที่จะให้ LIS ของแต่ละโรงพยาบาลส่งข้อมูลเข้ามาเก็บไว้ในฐานข้อมูลโดยมีข้อมูลคือ ชื่อเทสต์ ผลตรวจของเทสต์นั้นๆ ค่าปกติหรือค่าอ้างอิงของแต่ละโรงพยาบาลเป็นอย่างน้อย

a.สามารถรองรับข้อมูลที่เป็นรูปภาพได้ด้วยเช่นผลที่มาจาก Out Lab เป็นต้น

7. ใช้ หมายเลขบัตรประจำตัวประชาชนเป็น Primary key ในการสืบค้นผลการตรวจและหากเป็นกรณีผู้ป่วยต่างตัวสามารถใช้เลขพาสปอร์ตแทนได้ หากไม่มีให้ใช้รหัสสถานพยาบาลตามด้วย HNของสถานพยาบาลนั้นๆแทน

a.หากเป็นการค้นผู้ป่วยใน โรงพยาบาลตนเองสามารถค้นได้ตลอดเวลา

b.หากต้องการค้นผลที่ตรวจโดยโรงพยาบาลอื่น ผู้ป่วยรายนั้นจะต้องมีสถานะกำลัง visit ที่โรงพยาบาลที่ต้องการค้นผลและได้รับการยินยอมให้เปิดเผยข้อมูลจึงจะสามารถค้นได้

คุณสมบัติ Application

- c. สามารถแสดงได้ว่าเคยไปตรวจที่โรงพยาบาลไหนบ้าง
8. สามารถ พิมพ์ผลการตรวจได้ โดยใบรายงานผลจะต้องระบุได้ว่าผลตรวจนั้น เป็นผลตรวจจากห้องปฏิบัติการใด
9. สามารถคำนวณสถิติเพื่อเปรียบเทียบปริมาณงานของแต่ละโรงพยาบาลได้โดยตรง เช่น การคำนวณ FTE
10. จะต้องสอนการใช้งาน/ฝึกอบรม ให้ผู้ใช้งานสามารถใช้งานได้และแนะนำการแก้ไขปัญหาเบื้องต้นพร้อมคู่มือการใช้งานภาษาไทยให้กับผู้ใช้งาน



HOSPITAL

Health Thank You

Mr.Akkarapol Utbuawong