

บทที่ 2

ทฤษฎีและระบบงานที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

แนวคิดที่ได้ ได้มากจากการจัดกิจกรรม [1] Walk rally เป็นกิจกรรมที่มีมาตั้งแต่ยุค กรีก ต่อมาจนถึง ยุคเฟื่องฟูของโรมัน โดยเมื่อหลังจากสงครามได้สงบลง พระราชาก็ทรงให้เหล่าแม่ทัพหรือ ทหารระดับสูงมาประลองหรือทำกิจกรรม เพื่อทดสอบปฏิภาณไหวพริบของนายทหารผู้นั้น ซึ่งการ ประลองก็คือ จะให้เดินทางไปยังจุดต่างๆ โดยใช้ม้าหรือเดินเท้าแล้วจะกำหนด(ซึ่งก็เป็นลักษณะใน การจัดกิจกรรมวอล์คแรลลี่ในปัจจุบัน) และระหว่างรายทางนั้นก็ได้อุปสรรคมากมาย ที่ได้ทำขึ้น รวมไปถึงกลวงที่ลึกลับซับซ้อนต่างๆ และถ้าหากผู้ใดทำได้ตามที่กำหนดในกิจกรรมนั้นๆ ก็จะได้รับปูน บำเหน็จและได้เลื่อนตำแหน่งขึ้นไป

กิจกรรม Walk rally นี้ต่อมาประเทศญี่ปุ่นจึงมีแนวคิดมาปรับใช้วอล์คแรลลี่ในการทำงานขึ้นมา โดยนายชิเกรุ โคบายาชิ ซึ่งเป็นผู้จัดการบริษัท โซนี่ เมืองอัสฮิเกะ เนื่องด้วยในขณะนั้นบริษัทประสบ ภาวะขาดทุน เขาจึงศึกษาและตรวจสอบอย่างจริงจัง และเขาก็ได้พบว่า การบริหารงานของแต่ละ หน่วยงานขาดการประสานงานกันอย่างต่อเนื่อง และจิตสำนึกที่ดีในการทำงานและองค์กร หรืออาจ เรียกได้ว่าขาดการทำงานเป็นทีมนั่นเอง ซึ่งต่างกับทางด้านตะวันตกจะเน้นในเรื่องการทำงานเป็นทีม เขาจึงจึงจัดกิจกรรมวอล์คแรลลี่ขึ้น โดยเน้นการเรียนรู้และปฏิบัติด้วยตนเองและการทำงานร่วมกัน โดย การจัดการรูปแบบต่าง ๆ ที่เหมาะสมบนพื้นฐานสภาพความเป็นจริงและปรากฏว่าได้ผลดีเกินคาดบริษัท ก็ฟื้นตัวและมีกำไรอย่างต่อเนื่องต่อมา วอล์คแรลลี่ในประเทศไทย เกิดขึ้นมาเพราะได้เปิดโอกาสให้ บริษัทต่างชาติเข้ามาดำเนินธุรกิจเพิ่มมากขึ้น แนวคิดและวิธีการจัดการบริหารต่าง ๆ เข้ามามีบทบาท มากขึ้น ขณะที่ปรึกษาด้านเทคนิคชาวญี่ปุ่นของบริษัท ไทยบริดจสโตน จำกัด ก็ได้นำเอากิจกรรม Walk rally มาปรับใช้เพื่อให้เกิดการทำงานเป็นทีมเพิ่มมากขึ้น โดยเน้นที่กลุ่มระดับหัวหน้างานต้องเข้าร่วม กิจกรรมวอล์คแรลลี่ทุกคน และกลุ่มปฏิบัติการทุกคน ได้ผลตอบรับเป็นอย่างดีเกี่ยวกับกระบวนการ ทำงาน กลุ่มของเครือซีเมนต์ไทย ได้นำกิจกรรม Walk rally มาใช้ในหน่วยงาน และได้แพร่ขยายไปสู่ หน่วยงานในระดับชั้นนำอย่างรวดเร็วจนถึงปัจจุบัน

2.1.1 ความหมายของกิจกรรม Walk rally

กิจกรรมวอล์คแรลลี่ [1] Walk rally คือ กิจกรรมกลุ่มที่กระทำควบคู่ไปกับการแทรกสภาวะ เพื่อพัฒนาบุคลากรและองค์กร ให้เหมาะสมกับบุคลากร วัฒนธรรมองค์กร และ สภาพเศรษฐกิจสังคม ในภาวะการแข่งขันอย่างสูงในเชิงธุรกิจ ในยุคปัจจุบัน โดยเน้นให้ผู้เข้าร่วมกิจกรรมอาศัย ทักษะ และ ศักยภาพ เพื่อเพิ่มประสิทธิภาพในการทำงานร่วมกัน มีความคิดสร้างสรรค์ในการทำงาน หรือการสร้าง ผลงานเชิงสัญลักษณ์ ซึ่งจะแทรกเข้าไปในกิจกรรม

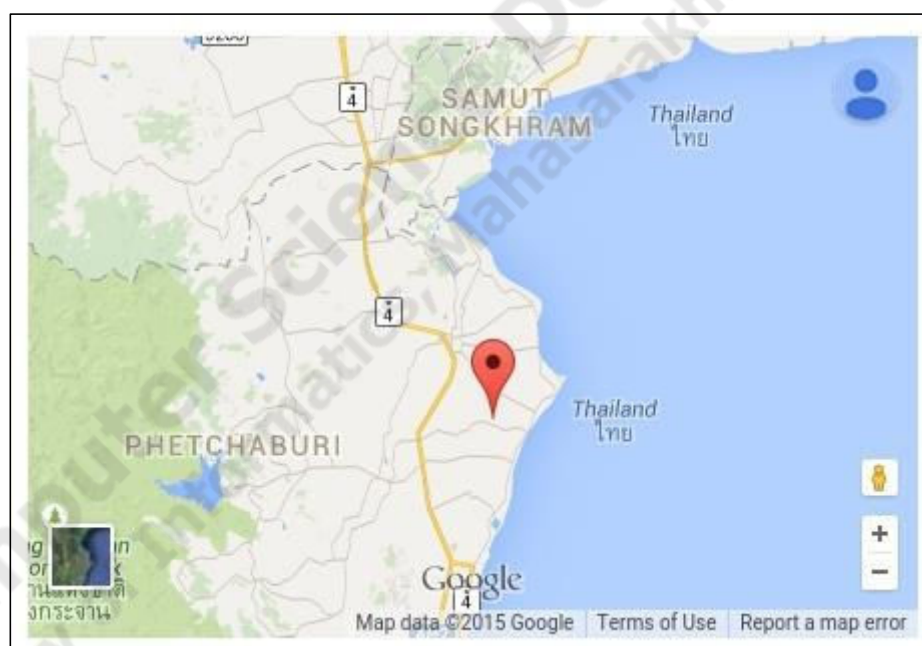
กิจกรรมวอล์คแรลลี่ ถือว่าเป็นกิจกรรมที่มีความสำคัญ ซึ่งจะสร้างประโยชน์ต่อองค์กรทำให้ พนักงานมีประสิทธิภาพเพิ่มขึ้น ลักษณะของกิจกรรม วอล์คแรลลี่ Walk rally แบ่งกว้างๆได้ 2 ลักษณะ

ทั้งแบบเชิงสาระ และเชิงกึ่งสาระ เน้นสั้นทนาการเป็นหลัก ซึ่ง Walk rally ทั้งสองแบบนี้ จะให้ผู้เข้าร่วมกิจกรรมได้ ทำภารกิจตามฐานต่างๆให้เป็นไปตามเป้าหมายของฐานนั้นๆ โดนการผสมผสาน ความ สนุกสนาน ตื่นเต้น ทำทาย ความสามัคคี และ มนุษย์สัมพันธ์ ไหวพริบ และสมรรถภาพทางร่างกายเข้าไว้ด้วยกัน การจัดกิจกรรมวอล์คแรลลี่ จำเป็นต้องมี ฐานกิจกรรม เพื่อสร้างทักษะและเพิ่มศักยภาพทั้งทางร่างกายและจิตใจไปพร้อมกัน

กิจกรรม Walk rally อาจจะใช้เพื่อเพื่อรองรับกลุ่มบุคคลต่างๆ ซึ่งอาจจะต่างที่มาซึ่งก็จะความหลากหลายทางด้านกายภาพและลักษณะบุคลิก ดังนั้นข้อสำคัญในการจัดกิจกรรม Walk rally ที่สำคัญก็คือ ควรเป็นไปอย่างสนุกสนานและไม่ควรเบื่อหน่าย เพราะเป็นเหตุอันจะทำให้กิจกรรมไม่น่าสนใจ โดยหลักการสำคัญของ กิจกรรมวอล์คแรลลี่ นั้น ต้องเป็นกิจกรรมที่สามารถสรุปความและตอบโจทย์ ในแง่ของหัวข้อที่ผู้เข้าร่วมกิจกรรมพึงได้รับนั่นเอง

2.2 ระบบงานที่เกี่ยวข้อง

2.2.1 Google Maps API



ภาพประกอบที่ 2.1 Google Maps API

ที่มา : <https://swiftlet.co.th/google-api/>

Google map API [2] ทำให้แอปพลิเคชันของผู้ใช้งานสามารถแสดงตำแหน่งของบาร์ ร้านอาหาร คาเฟ่ สนามบิน หรือร้านขายของชำที่อยู่ใกล้กับผู้ใช้ก็สามารถให้รายชื่อสถานที่ที่ผ่านการกรอง ซึ่งมีความเกี่ยวข้องกับผู้ใช้มากที่สุด Google Maps API เป็นชุด API ของ Google สำหรับพัฒนา Web application และ Mobile application (Android, iOS) ไว้สำหรับเรียกใช้แผนที่และชุด Service ต่างๆของ Google เพื่อพัฒนา Application ได้เหมือนกับที่ Google โดยแผนที่ยัง Features ต่างๆมากมายให้เรียกใช้ เช่น

- ชุดควบคุมแผนที่
- การปรับแต่งแผนที่ (Styled Map)
- การนำทางจากจุดหนึ่งไปยังอีกจุดหนึ่ง (Directions Service) Street View
- ชุดเครื่องมือวาดภาพบนแผนที่ (Drawing)
- การคำนวณความสูงของจุดพิกัด (Elevation Service)
- การดึงข้อมูล POI (Point of Interest) คือข้อมูลสถานที่ต่าง ๆ ที่ Google รวบรวมไว้ให้ เช่น โรงแรมห้างสรรพสินค้า

- การแปลงที่อยู่เป็นพิกัด Latitude และ Longitude (Geocoding Service)

โรงเรียน สถานที่ราชการต่างๆ และอื่นๆ อีกมากมาย (Places API) มาใช้งานใน Application เรา

2.2.1.1 การติดตั้ง Google Maps API

1. ขอ API Key [3] <https://mapsplatform.google.com/> เมื่อเข้ามาหน้าเว็บไซต์แล้วให้เข้าไปที่ Console

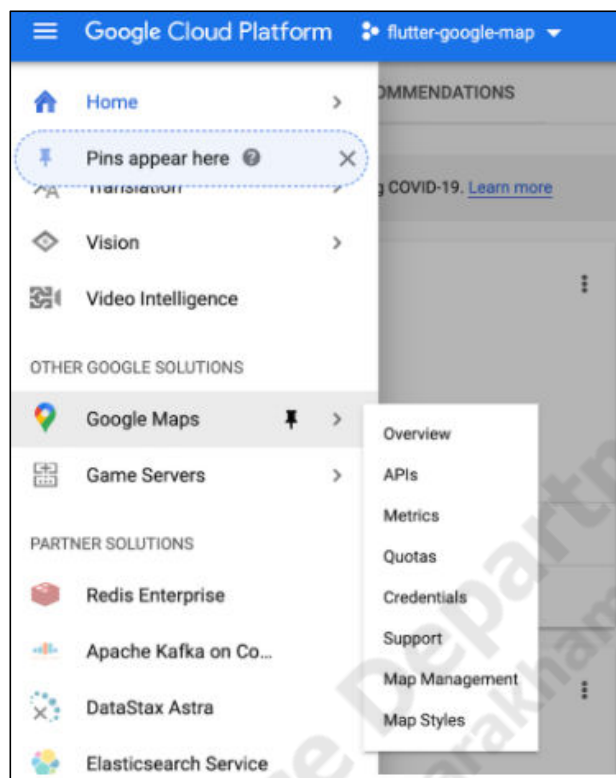
2. สร้าง Project ขึ้นมาก่อน เมื่อทำการสร้างเสร็จแล้วให้เปิดที่บด้านขวามือแล้วเลือกเมนูที่ชื่อว่า Google Maps

The screenshot shows the 'New Project' form in the Google Cloud console. At the top, there is a warning: 'You have 6 projects remaining in your quota. Request an increase or delete projects. Learn more' with a 'MANAGE QUOTAS' link. Below this, the 'Project name' field contains 'My Project 68529'. The 'Project ID' field shows 'vaulted-program-281304' with a note that it cannot be changed later and an 'EDIT' link. The 'Location' dropdown is set to 'No organisation' with a 'BROWSE' button. At the bottom, there are 'CREATE' and 'CANCEL' buttons.

ภาพประกอบที่ 2.2 การติดตั้ง Google Maps API

ที่มา : [How To] การติดตั้ง Google Maps for Flutter เบื้องต้น | Centillion Technology

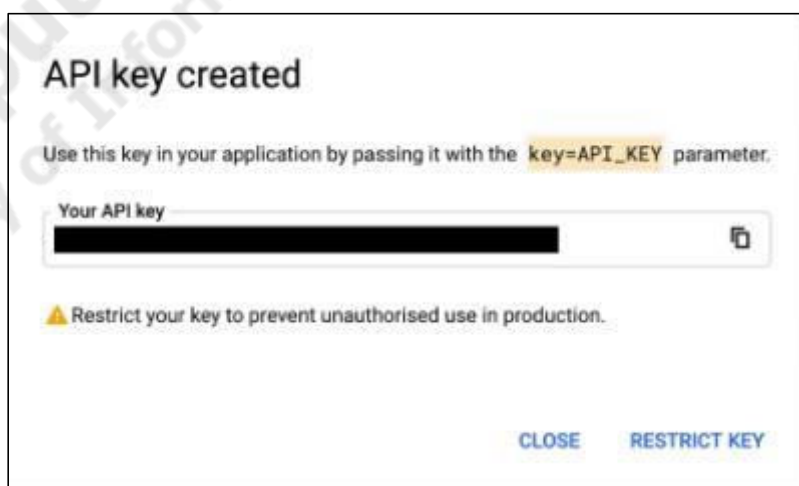
3. Google Maps จะมี API ให้ใช้หลายตัว เราสามารถเข้าไป Enable เพื่อที่จะเรียกใช้ API เหล่านั้นได้แต่ตอนนี้เราจะเลือก Enable Maps SDK for Android / IOS เพื่อที่จะเอา API Key ไปเรียกใช้งาน API



ภาพประกอบที่ 2.3 Google Maps มี API

ที่มา : [How To] การติดตั้ง Google Maps for Flutter เบื้องต้น | Centillion Technology

4. ต่อมาให้เข้าไปที่เมนู Credentials เพื่อที่จะสร้าง API Key เอาไปแปะในโค้ดให้เชื่อมต่อกับ Google Cloud Platform กดเลือก Create Credentials -> API Key เราก็จะได้ API Key มา เทำนี้ก็จะสามารถเชื่อมต่อกับ Google Map API ได้



ภาพประกอบที่ 2.4 API Key Created

ที่มา : [How To] การติดตั้ง Google Maps for Flutter เบื้องต้น | Centillion Technology

5. สำหรับ Android ให้เข้าไปเพิ่ม API Key ที่ไฟล์ AndroidManifest.xml ในโฟลเดอร์ Android (android/app/src/main/AndroidManifest.xml) และเพิ่ม เพื่อขอ permission เพื่อเพิ่ม API KEY เข้าไป

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.fluttergooglemap">

    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>

    <application
        android:name="io.flutter.app.FlutterApplication"
        android:label="fluttergooglemap"
        android:icon="@mipmap/ic_launcher">

        <meta-data android:name="com.google.android.geo.API_KEY"
            android:value="YOUR API KEY"/>

        <activity
            android:name=".MainActivity"
            ...
```

ภาพประกอบที่ 2.5 เพิ่ม API Key ที่ไฟล์ AndroidManifest.xml

ที่มา : [How To] การติดตั้ง Google Maps for Flutter เบื้องต้น | Centillion Technology

6. เข้าไปที่ไฟล์ pubspec.yaml ติดตั้ง google_maps_flutter: ^0.5.28+1 (เวอร์ชันล่าสุดเมื่อ May 21, 2020) และ import Library

7. สร้างตัวแปรประเภท GoogleMapController คือ class ที่เอาไว้สร้าง Future อื่นๆ ที่เอาไว้ควบคุมการทำงาน

```
class _MyHomePageState extends State<MyHomePage> {
  Completer<GoogleMapController> _controller = Completer();
```

ภาพประกอบที่ 2.6 สร้างตัวแปรประเภท GoogleMapController

ที่มา : [How To] การติดตั้ง Google Maps for Flutter เบื้องต้น | Centillion Technology

8. Library จะมี Widget มาให้ชื่อว่า Google Map () เริ่มต้น Google Map จะต้อง initialCameraPosition เสมอ โดยกำหนดพิกัดจุดเริ่มต้นสถานที่ที่จะให้ Google Map แสดงขึ้นมาแล้ว ลอง Run

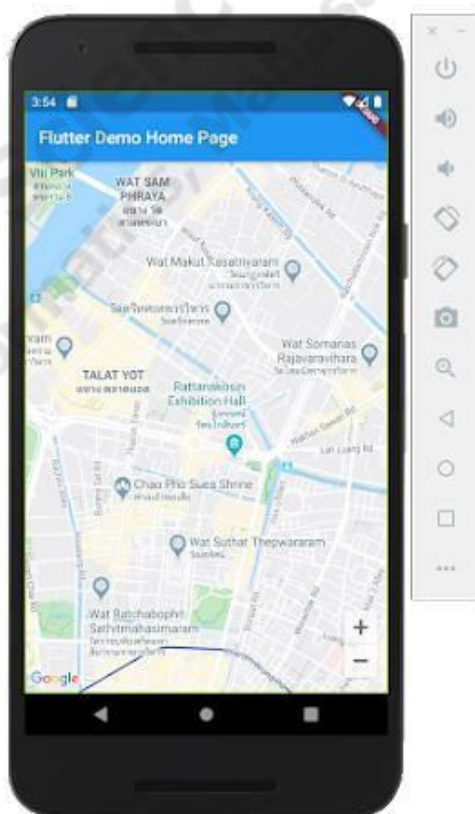
```

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text(widget.title),
    ),
    body: GoogleMap(
      initialCameraPosition: CameraPosition(
        target: LatLng(13.757429, 100.502465), //กำหนดพิกัดเริ่มต้นบนแผนที่
        zoom: 15, //กำหนดระยะการซูม สามารถกำหนดค่าได้ 0-20
      ),
      onMapCreated: (GoogleMapController controller) {
        _controller.complete(controller);
      },
    ),
  );
}

```

ภาพประกอบที่ 2.7 การกำหนดพิกัดจุดเริ่มต้นให้กับสถานที่

ที่มา : [How To] การติดตั้ง Google Maps for Flutter เบื้องต้น | Centillion Technology



ภาพประกอบที่ 2.8 ตัวอย่างหน้าแสดง Maps

ที่มา : [How To] การติดตั้ง Google Maps for Flutter เบื้องต้น | Centillion Technology

2.2.2 GOLANG

ภาษา Golang [4] หรือ ภาษา Go เป็นภาษา Programming แบบ Open-Source ถูกพัฒนาขึ้นโดยบริษัท Google ในปี 2007 และเป็นภาษาที่ได้รับความนิยมมากขึ้นเรื่อย ๆ ในยุคนี้ โดยภาษา Go นั้นจะมีจุดเด่นในเรื่องของ Performance ที่สามารถทำงานได้อย่างรวดเร็วเทียบกับภาษาอื่น ๆ อีกทั้งยังมีจุดเด่นในเรื่องของ Simplicity ที่เน้นความง่ายในการเขียนและการอ่าน และยังสามารถทำ Concurrent Programming ได้ง่าย เพราะภาษา Golang ถูกออกแบบมาเพื่อทำให้ Application ที่ต้องใช้ Multi-Threading หรือ Distributed Systems เป็นเรื่องที่ยั่งยืน

2.2.3 ติดต่อฐานข้อมูล MySQL

การพัฒนาโปรแกรมด้วยภาษา Go ที่มีการบันทึกข้อมูลบนไฟล์ฐานข้อมูล MySQL โดย ใช้ XAMPP จำลองเป็น WebServer และ phpMyAdmin เป็นตัวสร้าง Database เริ่มด้วยการติดตั้งและสร้าง Table สำหรับในการเก็บข้อมูล ในบทความนี้จะใช้ชื่อ Table เป็น member ที่มีการเก็บข้อมูล Username และ Password รวมทั้งสถานะว่าเป็น User หรือว่า Admin

2.2.3.1 โปรแกรม XMAPP

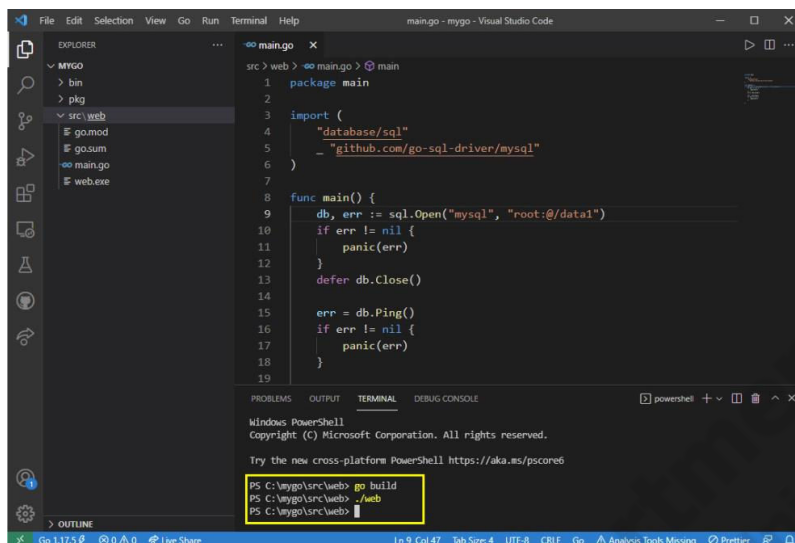
วิธีการจำลอง ให้คอมพิวเตอร์ของเราสามารถทำตัวเองเป็น WebServer เพื่อทดสอบการพัฒนาโปรแกรมทำการดาวน์โหลด XMAPPh^{https://www.apachefriends.org/index.html} เลือกตามระบบปฏิบัติการของคุณ (ในตัวอย่างใช้ระบบปฏิบัติการ Windows) แล้วติดตั้งตามขั้นตอนที่แนะนำ

2.2.3.2 ติดตั้ง GO-MySQL-Driver

เมื่อเราสร้าง Database เสร็จเราก็มาเขียน GO เชื่อมไปยัง MySQL กัน ในบทความนี้จะใช้ Library ของ GO ชื่อว่า go-sql-driver/mysql ถ้าเข้าไปดูที่ไฟล์ go.mod จะพบ require.com/go-sql-driver/mysqlv1.6.0 เพิ่มเข้ามา Go-MySQL Driver ^{https://pkg.go.dev/github.com/go-sql-driver/mysql}

2.2.3.3 ติดต่อฐานข้อมูล MySQL

ที่ไฟล์ main.go เขียนโค้ดดังนี้คอมไพล์ไฟล์ซอร์ส go build แล้ว เรียกใช้งาน โดยใช้คำสั่ง ./web ถ้าติดต่อฐานข้อมูลไม่ได้จะแสดงข้อผิดพลาดถ้าติดต่อฐานข้อมูลได้ จะไม่แสดงข้อผิดพลาดใดๆ



ภาพประกอบที่ 2.9 ติดต่อฐานข้อมูล MySQL

2.2.4 การสร้าง Go API โดยใช้ GIN, GORM และ MySQL

[5] Gin เป็น Web framework ที่เขียนด้วยภาษา golang ที่ถูกพัฒนาต่อมาจาก Martini ที่หยุดพัฒนาไปแล้ว โดย Gin จะใช้ Customized httprouter ทำให้มีประสิทธิภาพด้านความเร็วที่สูงมาก ถ้าคุณกำลังมองหา Framework ที่มี Performance กับ Productivity ที่ดี Gin เป็นเพียงไม่กี่เฟรมเวิร์กที่ไม่ได้ให้ฟีเจอร์มากมายเหมือนเฟรมเวิร์กอื่นๆ มีเพียงเครื่องมือที่จำเป็นในการสร้าง API เช่น การกำหนดเส้นทาง การตรวจสอบความถูกต้อง ฯลฯ ผู้เชี่ยวชาญเชื่อว่า Gin ดีที่สุดสำหรับการลงทุนระยะยาว เพื่อคว้าข้อได้เปรียบที่เหมาะสมจากประสิทธิภาพและความยืดหยุ่นสูง

2.2.4.1 สร้างโปรเจกใหม่

สร้างโปรเจกใหม่ตัวอย่างชื่อ first-api สร้างไฟล์ main.go เปิดเทอร์มินัลแล้วใช้คำสั่ง

```
go mod init first-api
```

ภาพประกอบที่ 2.10 คำสั่งสร้างไฟล์ go.mod

2.2.4.2 ติดตั้งแพ็คเกจ

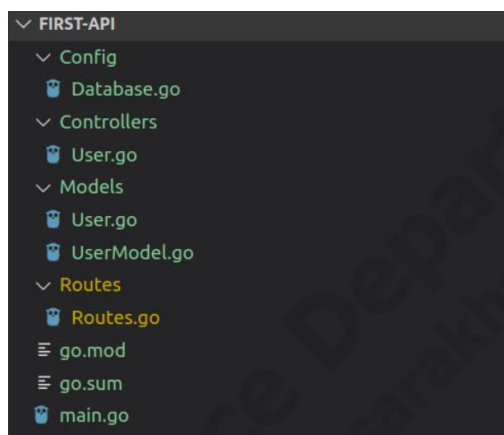
```
go get github.com/go-sql-driver/mysql
go get github.com/gin-gonic/gin
go get github.com/jinzhu/gorm
```

ภาพประกอบที่ 2.11 คำสั่งแพ็คเกจ mysql, gin และ gorm

แพ็คเกจภายนอกสำหรับโครงการ: ไดรเวอร์ sql สำหรับ mysql, gin และ gorm
 ป้อนคำสั่งต่อไปนี้เพื่อติดตั้งในโครงการ

2.2.4.3 โครงสร้างโครงการ

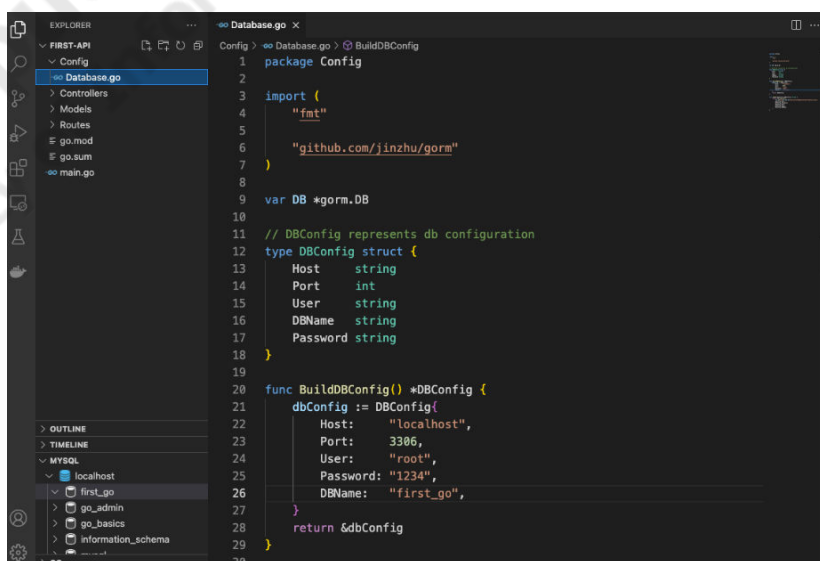
เราปฏิบัติตามรูปแบบ MVC สำหรับการสร้าง API ที่เป็น Model, View และ Controller แต่ 'V' ยังคงเจียบอยู่ที่นี้เพราะเรากำลังสร้าง API อยู่ในขณะนี้ เรามีการกำหนดค่าฐานข้อมูล โมเดล เส้นทาง ตัวควบคุม และแน่นอนว่าไฟล์หลักที่เราเขียนขอบเขตเวลาซึ่งเป็นจุดเริ่มต้นของโครงการของเรา เรามีสแน็ปช็อตสำหรับสร้างโครงสร้างของโครงการ



ภาพประกอบที่ 2.12 สร้าง API Model, View และ Controller

2.2.4.4 ตั้งค่าฐานข้อมูล

สร้างฐานข้อมูลใหม่ใน mysql ชื่อ first_go จากนั้นเราเพิ่มการกำหนดค่าฐานข้อมูลในไฟล์ Database.go ภายในไดเรกทอรี Config ฉันได้เพิ่มรายละเอียดการกำหนดค่าของฉันแล้ว และคุณต้องเพิ่มการกำหนดค่าของคุณเอง



ภาพประกอบที่ 2.13 กำหนดค่าฐานข้อมูล

2.2.4.5 สร้างแบบจำลอง Model

ตอนนี้ มาสร้างแบบจำลองสำหรับผู้ใช้ที่ประกอบด้วยฟิลด์ต่อไปนี้: ชื่อ, อีเมล, โทรศัพท์, ที่อยู่, name, email, phone, address, เราสร้างไฟล์ UserModel.go ภายในไดเรกทอรี Models ของเรา

2.2.4.6 กำหนดค่าการกำหนดเส้นทาง

ในการทำงานเดียวกัน เราสร้างการกำหนดเส้นทางสำหรับโครงการของเรา เราสร้างกลุ่มเพื่อปรับการกำหนดเส้นทางของเราที่เกี่ยวข้องกับผู้ใช้ และเราต้องดูแลกลุ่มในขณะที่ใช้งาน API มาสร้างไฟล์ Routes.go ในโฟลเดอร์ Routes กัน

2.2.4.7 สร้างตัวควบคุม

เราจัดการคำขอ http ที่มาจากส่วนหน้าในตัวควบคุม เราสร้างฟังก์ชันต่างๆ ที่จัดการคำขอเฉพาะของเราที่เราเตอร์ของเราส่งไปยังตัวควบคุม เรามี User.go ภายใน Models เพื่อโต้ตอบกับฐานข้อมูล เราตอบสนองต่อผู้ใช้ตามข้อมูลที่เราได้รับจากฐานข้อมูลของเรา หากเราไม่ได้รับข้อผิดพลาด เราจะตอบกลับเป็นสถานะตกลง และหากเราได้รับข้อผิดพลาด เราจะระบุสถานะข้อผิดพลาด เราสร้างไฟล์ User.go ภายในโฟลเดอร์ Controllers ไฟล์สำคัญที่ดึงข้อมูลและโต้ตอบกับฐานข้อมูลของเราโดยตรง เราสร้างไฟล์ User.go ภายในโฟลเดอร์ Models เพื่อจัดการกับคำขอฐานข้อมูลตอนนี้เราพร้อมแล้วที่จะทดสอบ API แรกของเราโดยใช้ Go และ Gin!

2.2.4.8 การสาธิตเรียกใช้คำสั่งต่อไปนี้เพื่อเริ่มต้นเซิร์ฟเวอร์ของคุณ

```
go run main.go
```

ภาพประกอบที่ 2.14 เรียกเซิร์ฟเวอร์

นี่คือปลายทางที่เราจะใช้เพื่อสร้าง อัปเดต อ่าน และลบข้อมูลผู้ใช้

GET user-api/user → ดึงข้อมูลผู้ใช้ทั้งหมด

POST user-api/user → เพิ่มข้อมูลผู้ใช้ใหม่

GET user-api/user/{id} → ดึงข้อมูลผู้ใช้คนเดียว

PUT user-api/user/{id} → อัปเดตข้อมูลผู้ใช้

DELETE user-api/user → ลบข้อมูลผู้ใช้

ตอนนี้ เปิดเครื่องมือทดสอบปลายทางที่คุณมี เราจะใช้ Postman ซึ่งยอดเยี่ยมมาก โปรดตรวจสอบให้แน่ใจว่าข้อมูลถูกแทรกในรูปแบบ JSON ผ่าน Postman เนื่องจากเราไม่ได้ทำงานกับข้อมูลแบบฟอร์มในโมดูลนี้

2.2.4.9 สร้างผู้ใช้ใหม่

เราจะสร้างผู้ใช้สองคนพร้อมกันที่ Postman เลือกเป็น POST ใส่ url:http://localhost:8080/user-api/user เลือกเป็น Body -> Raw -> JSON

สร้างผู้ใช้ที่1

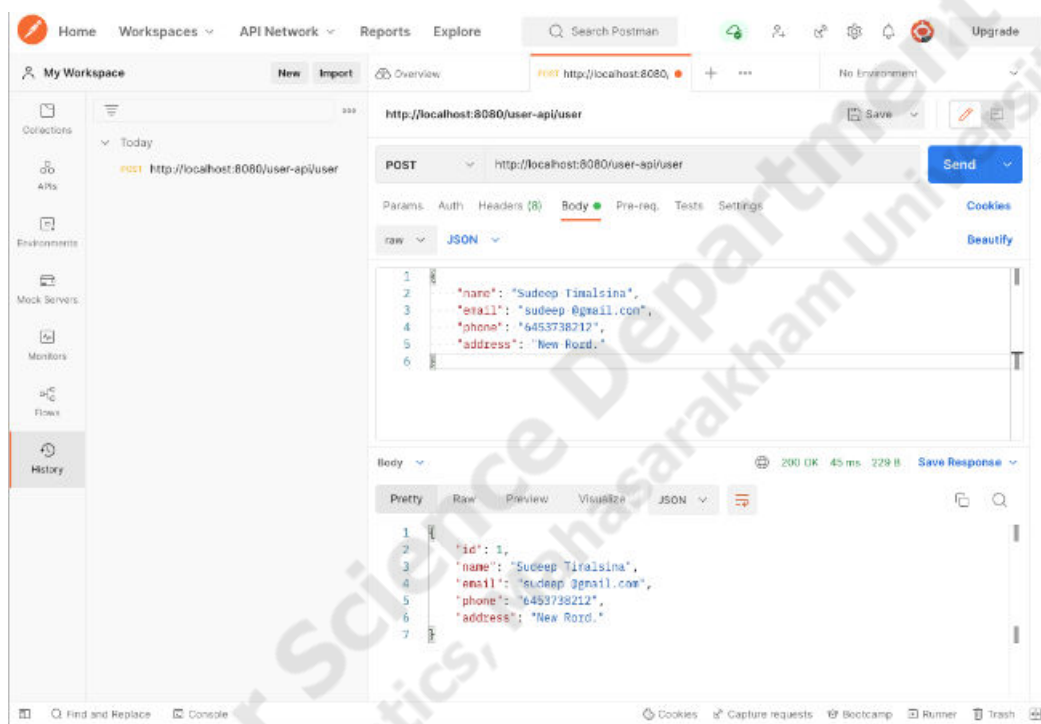
```
{
```

```

"name": "Sudeep Timalisina",
"email": "sudeep@gmail.com",
"phone": "6453738212",
"address": "New Rord."
}

```

คลิก Send ที่ web browser จะแสดงข้อมูลผู้ใช้ที่1 ที่เพิ่มเข้าไป



ภาพประกอบที่ 2.15 Postman

2.2.4.10 รับผู้ใช้โดย id

รับข้อมูลผู้ใช้สำหรับผู้ใช้ที่มี id=1 โดยการป้อน `http://localhost:8080/user-api/user/1`

2.2.4.11 อัปเดตผู้ใช้

อัปเดตข้อมูลผู้ใช้สำหรับผู้ใช้ที่มี Id=1

```

{
  "name": "Sudeep Timalisina",
  "email": "sudeep@gmail.com",
  "phone": "6453738212",
  "address": "New Baneshwor"
}

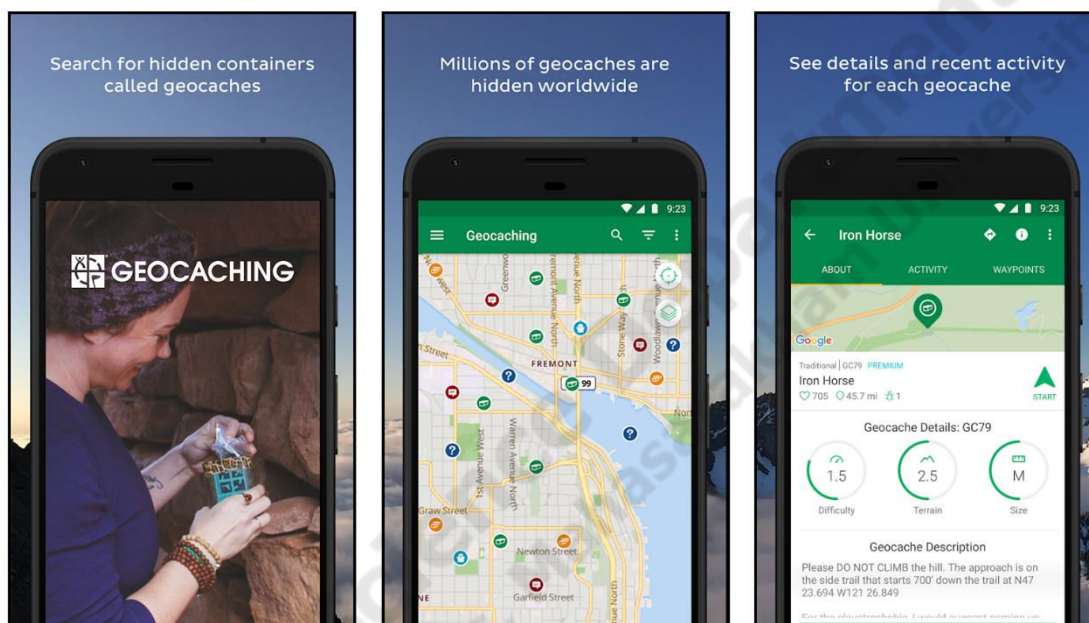
```

ที่ Postman เลือกเป็น PUT ใส่ url `http://localhost:8080/user-api/user/1` เลือกเป็น Body -> Raw -> JSON ที่ web browser จะแสดงข้อมูล ผู้ใช้ 1 ที่ แก้ไขแล้ว

2.3 ระบบงานที่เกี่ยวข้อง

2.3.1 Geocaching

Geocaching เกมซ่อนของและออกหากล่องระดับโลกที่ทุกคนสามารถร่วมเล่นได้ เพียงคุณมี GPS ที่สามารถค้นหาตำแหน่งได้ผ่านโทรศัพท์มือถือที่มีระบบ GPS หรือเครื่องรับสัญญาณ GPS ก็สามารถร่วมเล่นค้นหาสิ่งของกับคนทั้งโลกได้แล้ว



ภาพประกอบที่ 2.16 แอปพลิเคชัน GEOCACHING

ที่มา : <https://chiangmai.siamdot.com/geocaching-how-to-chiangmai/>

ข้อดี

- อินเทอร์เน็ตแสดงรายละเอียดเกี่ยวกับสมบัติแต่ละชิ้น
- เมื่อเจอจุดที่ต้องการแล้ว สามารถคลิกเข้าไป ระบบจะมาร์คจุดอัตโนมัติ แล้วบอกระยะห่างระหว่างเรากับสมบัติเป็นรูปไม้บรรทัด

ข้อเสีย

- เวอร์ชันที่ต้องชำระเงินมีข้อจำกัดในการทำงานหลายอย่าง
- ระยะเวลาในการตรวจสอบสมบัติใช้ระยะเวลานาน

2.3.2 Pokémon GO

[6] เป็นเกมพกพา อิงพิกัดภูมิศาสตร์ (location-based) ผสานกับเทคโนโลยีความเป็นจริงเสริม (Augmented reality) เล่นฟรี บนสมาร์ตโฟนที่ต้องมีกล้องหลัง, มีระบบตรวจจับพิกัดตำแหน่งบนโลก (GPS), ตัวจับระดับการหมุน และจอสัมผัส พัฒนาโดยบริษัทไนแอนติก ออกจำหน่ายใน

ค.ศ. 2016 บนระบบปฏิบัติการไอโอเอส และแอนดรอยด์ นอกจากเล่นบนมือถือแล้ว ยังสามารถใช้ร่วมกับอุปกรณ์สวมใส่ขนาดเล็กเรียกว่า โปเกมอนโกพลัส พัฒนาโดยนินเท็นโด ซึ่งเชื่อมต่อโดยใช้บลูทูธ เพื่อแจ้งผู้ใช้งานว่ามีโปเกมอนอยู่ใกล้ตัวผ่านแสงแอลอีดี https://th.wikipedia.org/wiki/โปเกมอน_โก และแจ้งเตือนเบาๆ เกมจะให้ผู้เล่นจับ ต่อสู้ ฝึกฝน และแลกเปลี่ยนโปเกมอนเสมือนจริงที่จะมีอยู่ทั่วโลก เกมจะไม่มีค่าใช้จ่าย แต่จะมีระบบสนับสนุนการซื้อขายในเกม



ภาพประกอบที่ 2.17 แอปพลิเคชัน Pokémon GO

ที่มา : <https://th.wikipedia.org/wiki/%>

ข้อดี

- รู้สึกเหมือนกำลังเล่นเกมนี้ในโลกแห่งความเป็นจริง
- ได้ออกกำลังกายจากการเก็บระยะทางการเดินของกิจกรรมในแอปพลิเคชัน

ข้อเสีย

- การระบายแบตเตอรี่เป็นข้อเสียที่สำคัญของเกมนี้
- ใช้ข้อมูลจำนวนมากจากโทรศัพท์มือถือของคุณ
- ไม่สามารถใช้เกมนี้ในโหมดออฟไลน์

ตารางที่ 2.1 ตารางการเปรียบเทียบการทำงานของระบบ

| การทำงานของระบบ | Geocaching | Pokémon GO | Mini World race |
|-----------------------------|------------|------------|-----------------|
| ฟังก์ชันการค้นหาผู้เข้าร่วม | / | / | / |
| ฟังก์ชันแสดงพิกัดสถานที่ | / | / | / |
| ฟังก์ชันการลงทะเบียน | / | / | / |

ตารางที่ 2.1 ตารางการเปรียบเทียบการทำงานของระบบ (ต่อ)

| การทำงานของระบบ | Geocaching | Pokémon GO | Mini World race |
|--|------------|------------|-----------------|
| ฟังก์ชันการทำภารกิจ | / | / | / |
| ฟังก์ชันการเข้าสู่ระบบด้วย Facebook | / | | / |
| ฟังก์ชันการสร้างกิจกรรม | / | | / |
| ฟังก์ชันดูสถิติการแข่งขัน | / | | / |
| ฟังก์ชันการสนทนา | / | | / |
| ฟังก์ชันส่งคะแนนความพึงพอใจหลังจบการแข่งขัน | / | | / |
| ฟังก์ชันเข็มทิศภายในแอปพลิเคชัน | / | | |
| บอกระดับความยากในการค้นหา สภาพภูมิประเทศโดยรอบ และขนาดสมบัติ | / | | |
| ฟังก์ชันARหรือความเป็นจริงเสมือน | | / | |
| ฟังก์ชันนับจำนวนที่ก้าวเดิน | | / | |
| ฟังก์ชันการเข้าสู่ระบบด้วย google | | / | |
| ฟังก์ชันรายงานลำดับหลังการแข่งขัน | | | / |
| ฟังก์ชันผู้ชม | | | / |