

Q: Database Schema มีหน้าตาอย่างไร

A: เนื่องจากมีประสบการณ์ SQL มาก่อน จึงจะตอบคำถามนี้ในมุมมองที่ใช้ SQL

Table member ← สำหรับเก็บข้อมูล ส่วนตัวของคนที่มาวิ่ง

id : integer

title : varchar

th_first_name : varchar

th_last_name : varchar

en_first_name : varchar

en_last_name : varchar

birthday : date

email : varchar

idcard : varchar

address : varchar

phone_number : varchar

image_url : varchar

BIB : varchar(10)

Table emergency_contact ← สำหรับเก็บข้อมูล ผู้ติดต่อฉุกเฉิน

id : integer

full_name : varchar

relationship : varchar

phone_number : varchar

Table member_emergency_contact ← สำหรับเก็บข้อมูล ผู้ติดต่อกรณีฉุกเฉินนี้เป็นของใคร

id : integer

member_id : integer
emergency_contact_id : integer

Table member_medical_profile

id : integer
member_id : integer
medical_profile_id : integer

Table medical_profile ← สำหรับเก็บข้อมูล ข้อมูลทางการแพทย์

id : integer
blood_type : ENUM('A', 'AB', 'B', 'O')
allergies : varchar
congenital_disease : varchar
surgery : varchar
pregnant : ENUM('yes', 'no')
personal_medicine : varchar
running_accident : ENUM('yes', 'no')
exercise : ENUM('yes', 'no')
running_risk : ENUM('yes', 'no')

Table register ← สำหรับเก็บข้อมูล การลงทะเบียน ใครลงทะเบียนพร้อมกับ ข้อมูลเกี่ยวกับการวิ่ง

id : integer
member_id : integer
running_experience : ENUM('yes', 'no')
running_expect_finish_time : varchar
shirt_size : ENUM('XS', 'X', 'S', 'M', 'L', 'XL', 'XXL', 'XXXL')
created_at : datetime

Table savedraft

id : integer

user: varchar ← ใช้สำหรับยืนยันว่า ใครเป็นคน save draft นี้

step : integer ← บอกขั้นตอนว่า step ไหนแล้ว

draftjson : varchar ← เก็บข้อมูลที่ frontend ส่งมาเป็น json ให้เป็น string เมื่อไหร่ที่จะนำกลับไปใช้ ก็ดึง string นี้ แปลงเป็น json แล้วส่งกลับไปหา frontend

Q: จะใช้วิธีใดในการทำ Authentication

A: จะใช้ JWT ในการทำ Authentication โดยมีหลักการดังนี้

- ฝ่าย client จะส่งข้อมูลในการ login มายกตัวอย่างเช่น email, password มาที่ server
- server จะทำการสร้าง jwt ขึ้นมาและส่งกลับไปให้ client
 - ฝ่าย server เก็บ secret key ไว้เป็นความลับ
 - server จะทำการเอา header/payload ของข้อมูลที่จะส่ง มารวมเข้ากับ secret key จะได้ signature ขึ้นมา
 - server จะทำการรวม header/payload เข้ากับ signature จะได้เป็น jwt แล้วส่งกลับไปให้ client
- client จะเก็บ jwt เอาไว้
- เมื่อจำเป็นต้องเรียกใช้ protected route ใน server จะส่ง request พร้อมแนบ jwt นี้มาด้วย
 - server ได้รับ jwt จากฝั่ง client
 - server ทำการแยก jwt ออกนั้นออกมาเป็น header/payload และ original signature
 - server จะนำ header/payload ที่ client ส่งมา รวมเข้ากับ secret key ที่เก็บไว้ จะได้ test signature ออกมา

- ถ้า original signature == test signature ถือว่า jwt นั้น valid สามารถส่งข้อมูลไปได้
- แต่ถ้า original signature != test signature ถือว่า jwt นั้น invalid เพราะมีการแก้ไข jwt มาจากฝั่ง client แล้ว

Q: ต้องมี REST API Endpoint และ Method อะไรบ้าง (ไม่นับส่วนของการ Login / Register) ในกรณีที่ Endpoint ไหนมีการส่ง Body ต้องระบุ Body มาด้วยว่าต้องมีอะไรบ้าง (ไม่จำเป็นต้องละเอียด ในกรณีที่ field เยอะมากๆ) ใน API สำหรับการ Submit ส่งใบสมัคร ควรทำอะไรบ้างในขั้นตอนดังกล่าว สามารถเขียนเป็น Pseudo Code ได้

A: ตอบคำถามส่วน REST API Endpoint และ Method

1. POST /savedraft/:user/:step

สมมติว่า เป็นข้อมูลส่วนตัว step ที่ 1 และรหัสผู้ใช้เป็น sathirak

ก็จะยิงมาที่ /savedraft/sathirak/1

Body ที่ส่งจะประมาณนี้

```
{
  title : 'นาย'
  th_first_name : 'สธิระ'
  th_last_name : 'กิตติสุขมงคล'
  en_first_name : 'sathira'
  en_last_name : 'kittisukmongkol'
  birthday : '1997-12-03'
  email : 'sathira.k@ku.th'
  ...
  ...
  ...
}
```

2. GET /savedraft/:user/:step

สำหรับดึงข้อมูลที่เคย savedraft มาโดยใส่ user และ step ที่ต้องการจะดึง

3. POST /submit

Body ที่ส่งจะประมาณนี้

```
{
  member: {
    title : 'นาย'
    th_first_name : 'สธิระ'
    th_last_name : 'กิตติสุขมงคล'
    en_first_name : 'sathira'
    en_last_name : 'kittisukmongkol'
    birthday : '1997-12-03'
    email : 'sathira.k@ku.th'
    ...
    ...
    ...
  },
  running: {
    running_experience : true
    running_expect_finish_time : '01:30'
  },
  emergencyContacts: [
    {
      full_name: 'นางสาวมารดา ของสธิระ'
      relationship: 'มารดา'
      phone_number: '091-123-4567'
    },
    {
```

```

        full_name: 'นายบิชา ของสธิระ'
        relationship: 'บิดา'
        phone_number: '091-123-4567'
    }
],
medicalProfile: {
    blood_type : 'O'
    allergies : null
    congenital_disease : null
    surgery : null
    pregnant : false
    personal_medicine : null
    running_accident : false
    exercise : true
    running_risk : false
},
shirtSize: 'M'
}

```

ตอบคำถามขั้นตอน API สำหรับการ Submit ควรทำอะไรบ้าง

--- Controller ที่รับ POST /submit ---

```
let submitBody = req.body;
```

```
let checkContents = ['member', 'running', 'emergencyContacts',
'medicalProfile', 'shirtSize'];
```

```
let errors = [];
```

```
// validate the data
```

```
for(every key in checkContents) {  
    let result = validate(key, submitBody[key]);  
    if(result is not null) {  
        errors.push(result);  
    }  
}
```

```
// If errors is not empty means there is something invalid
```

```
if(errors is not empty) {
```

```
    send the response with error message
```

```
} else {
```

```
    let member = createMember(submitBody['member']);
```

```
    let emergencyContacts =
```

```
createEmergencyContact(submitBody["emergencyContacts"]);
```

```
    createMemberEmergencyContactRecord(member,
```

```
emergencyContacts );
```

```
    let medicalProfile =
```

```
createMedicalProfile(submitBody['medicalProfile']);
```

```
    createMemberMedicalProfileRecord(member, medicalProfile);
```

```
    createRegisterRecord(member, submitBody["running"],
```

```
submitBody['shirtSize']);
```

```
    send the success response
```

}