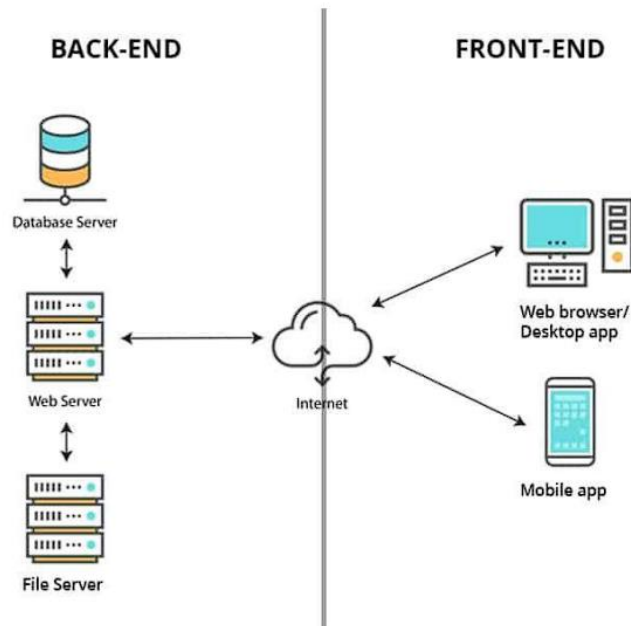


## ข้อสอบระดับ Junior Software Engineer [Frontend Developer]

### 1. จงอธิบายความหมายของโครงสร้าง ตามภาพด้านล่าง



จากภาพเป็นการทำงานของระบบหนึ่งๆ ที่แบ่งออกเป็น ส่วน Backend และ ส่วนของ Frontend ทาง ส่วนของ Backend ประกอบไปด้วย

- Database Server สำหรับเก็บข้อมูลผู้ใช้และที่อยู่ของไฟล์ที่เกี่ยวข้องใน File Server
- File Server ทำหน้าที่เก็บไฟล์ต่างๆ เช่น ภาพ เสียง
- Web Server เป็นส่วนที่จัดเก็บ API ให้ทาง Frontend สามารถเรียกใช้เพื่อขอข้อมูลใน Database และ File Server ไปแสดงผลต่อได้

ส่วนของ Frontend ประกอบไปด้วย

- Web Browser / Desktop App เป็นส่วนแสดงผลให้ผู้ใช้ผ่านเว็บแอปพลิเคชันหรือโปรแกรมบน Desktop
- Mobile App เป็นส่วนแสดงผลสำหรับผู้ใช้งานบนแอปพลิเคชันบนโทรศัพท์มือถือ

ระบบทั้งสองส่วน ถูกเชื่อมต่อด้วย Internet ซึ่งเป็นตัวกลางหลักในการส่งข้อมูล

## 2. จงอธิบายความหมายของ Bootstrap Frameworks, การออกแบบ CSS ด้วย BEM, การใช้ #Id, .Class, Ajax, Console, Debugger พร้อมทั้งตัวอย่าง Source Code

- **Bootstrap Frameworks** : เป็น CSS Framework ที่เมื่อนำมาใช้ในโปรเจก จะส่งผลให้การระบุชื่อ Class ที่เป็นคำเฉพาะของ Bootstrap Frameworks ใน HTML Tag จะส่งผลให้ Tag ดังกล่าวเป็นไปตามสไตล์ที่ Bootstrap กำหนดไว้ โดยไม่ต้องไปกำหนดสไตล์เองในไฟล์ CSS ตัวอย่างเช่น

```
<div class="container">
    <button class="btn btn-danger">ปุ่มนี้มีสีแดง</button>
</div>
```

ปุ่มนี้มีสีแดง

เพียงแค่ระบุชื่อ Class ที่เป็นคำเฉพาะ เช่น container ก็จะช่วยทำให้ Tag มี CSS Style แบบ Responsive ปรับไปตามหน้าจอหรือ btn btn-danger ที่ทำให้ Tag มีลักษณะเป็นปุ่ม ที่มีสีแดงในแบบทั้งตอนปกติ ตอน Hover หรือตอน Disabled ซึ่งแต่ละชื่อ Class ก็ส่งผลให้ปรับเปลี่ยนไปตามคีย์เวิร์ดใน Bootstrap Frameworks โดยสามารถดูได้จากในเว็บไซต์

- **การออกแบบ CSS ด้วย BEM** : เป็นลักษณะการเขียน CSS ให้เป็นระเบียบและมีมาตรฐานกลาง ทำให้ผู้ที่มาแก้ไขรายต่อไปเข้าใจ CSS ง่ายขึ้น แบ่งออกเป็น 3 ส่วน ได้แก่ Block , Element และ Modifier ซึ่งก็จะคั่นด้วยสัญลักษณ์ \_\_ ระหว่าง Block กับ Element และ - ระหว่าง Element กับ Modifier เพื่อแบ่งส่วนประกอบย่อยใน Tag นั้นอีกที ทำให้ CSS โดยรวมอ่านง่ายขึ้น เช่น

```
.header{
    Background-color: green;
    Border: 1px solid black;
}
.header__logo{}
.header__menu{}
.header__logo--icon{}
.header__logo--text{}
```

ทำให้เวลาแก้ไข CSS ที่หลังจะทราบได้ทันทีว่า Class ใดเป็น Class ย่อยของ Class ใดอีกที ทำให้ง่ายต่อการอ่านนั่นเอง แต่ชื่อ Class ก็จะมีลักษณะยาวเมื่อเขียนด้วยเช่นกัน

- **#Id** : เป็นเหมือนการกำหนดชื่อเฉพาะให้กับ HTML Tag ตัวหนึ่งๆ โดยแต่ละชื่อจะไม่สามารถซ้ำได้ในหน้าเว็บได้ สามารถใช้ได้ทั้งในการระบุ CSS Style ของ Tag ให้มีลักษณะเฉพาะ หรือจะใช้ระบุให้ Tag ดังกล่าวมีลูกเล่นจาก JavaScript ได้ ตัวอย่างเช่น

```
<p id="clicknum">click : 0</p>
<button onclick="clickFunction()">click me</button>
```

จากตัวอย่างเป็นการระบุให้ Tag หนึ่งๆมี id เฉพาะว่า clicknum โดยสามารถกำหนดให้ปุ่มเรียกฟังก์ชัน clickFunction() ใน Javascript เพื่อเรียกใช้ getElementById สำหรับแก้ไขค่าใน Tag ที่มี id เฉพาะว่า clicknum ให้อัปเดตตามข้อความที่เราตั้งค่าไว้ได้ต่อไป

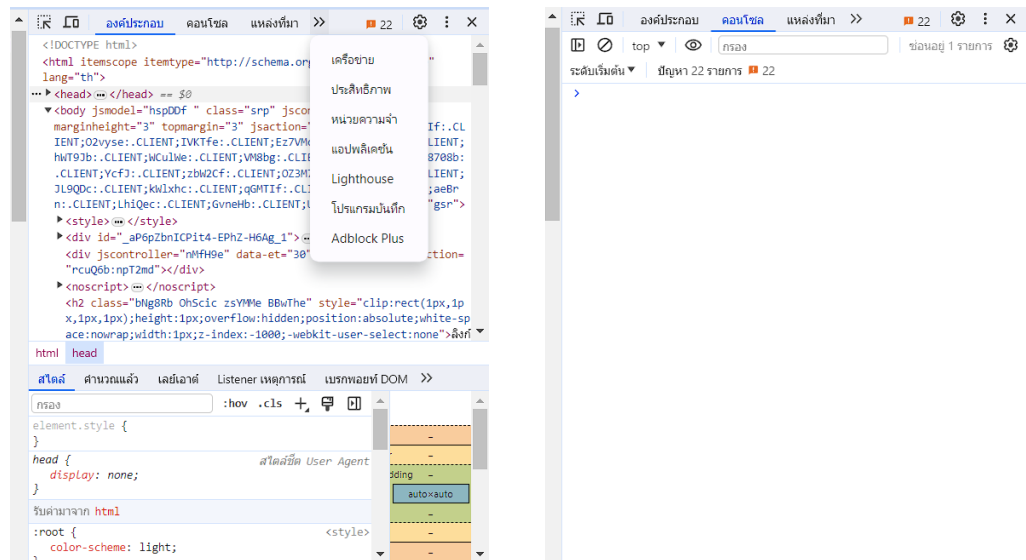
- **.Class** : เป็นการกำหนดชื่อเฉพาะให้กับ HTML Tag โดยสามารถกำหนดได้หลาย Tag ในหน้าเว็บเดียวกัน จึงเหมาะกับการนำมารระบุ CSS Style ของ Tag ให้มีลักษณะตามต้องการ เช่น

```
<div class="background">
  <div class="text-box">
    <h5>ผู้ใช้ที่ 1 :</h5>
    <p>ขอเอกสารระบุตัวตนด้วยครับ</p>
  </div>
  <div class="text-box">
    <h5>ผู้ใช้ที่ 2 :</h5>
    <p>ได้ค่ะ ขอเวลาเตรียมเอกสารสักครู่นะคะ</p>
  </div>
</div>
```

จากตัวอย่าง เราสามารถกำหนด CSS Style ให้กับ Tag ที่มี Class “text-box” หรือ Tag ที่อยู่ใน Class ดังกล่าวให้มีลักษณะตามที่ต้องการ นอกจากนี้ยังสามารถใช้ JavaScript เพิ่มลูกเล่นให้ Tag ด้วยการ GetElementsByName ได้อีกด้วย

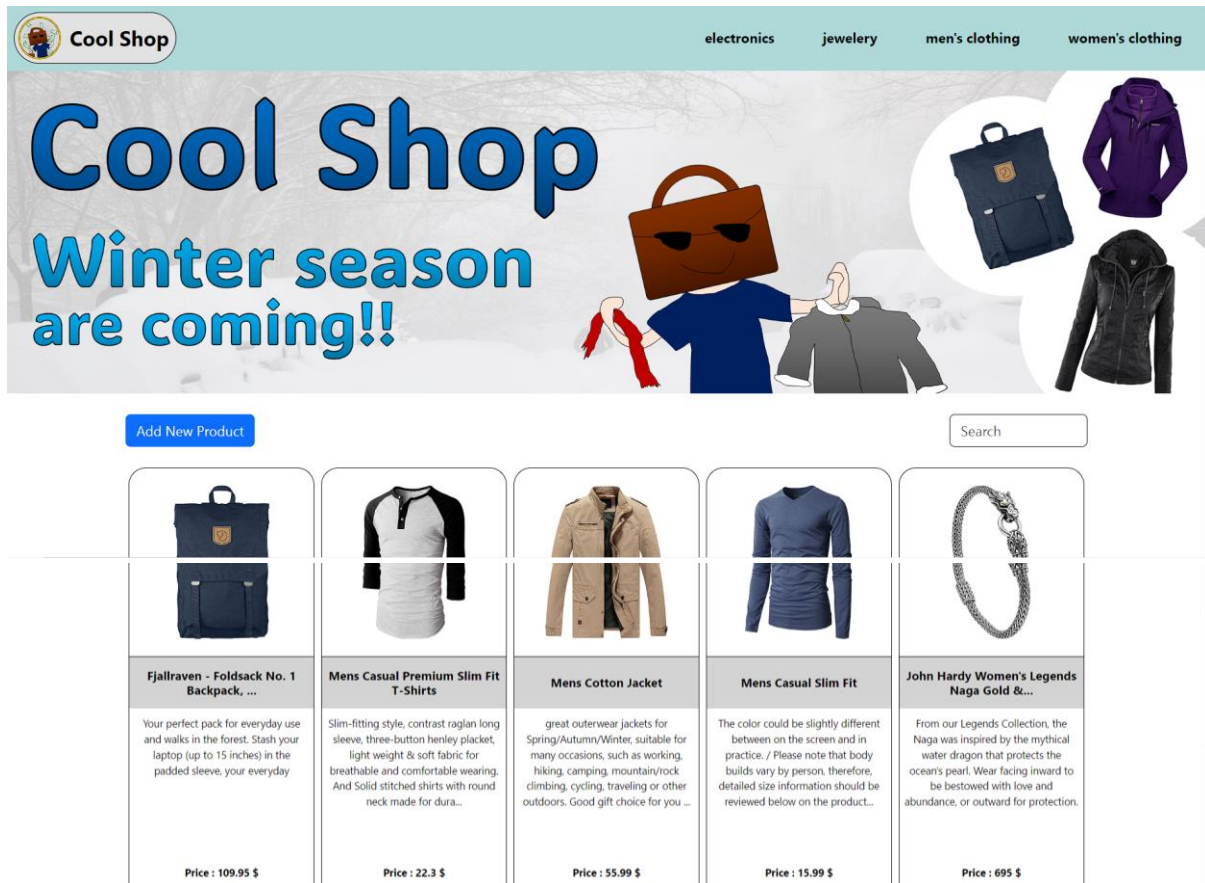
- **Ajax** : เป็นรูปแบบการเขียน Code ให้รองรับการรับส่งข้อมูลจาก Server ให้เกิดประสิทธิภาพมากขึ้น ด้วยการส่งคำขอและทำงานส่วนอื่นต่อแบบ Asynchronous และทำงานโค้ดส่วนที่กำหนดเมื่อมีข้อมูลส่งมาในภายหลัง นอกจากนี้หากต้องการประมวลผล Code เมื่อข้อมูลฝั่ง Server มีการอัปเดต จะมีการประมวลผลเฉพาะข้อมูลที่มีการอัปเดตเพิ่มเติมเท่านั้น ส่งผลให้เว็บไซต์โดยรวมโหลดไวขึ้น และใช้ทรัพยากรน้อยลง

- **Console** : เป็นหนึ่งใน Tool สำหรับการ Debugging ในขณะที่พัฒนาเว็บไซต์ ใช้งานได้ด้วยการเลือกเมนูตรวจสอบหน้าเว็บหรือ F12 สำหรับ Google Chrome ซึ่งมีประโยชน์มากในการตรวจสอบสถานะปัจจุบันของหน้าเว็บ ตัวอย่างของเมนู Console บางส่วน เช่น Element สำหรับตรวจสอบ Tag และ CSS Style ที่แสดงผล, Console สำหรับตรวจสอบ Error Log จาก JavaScript, Application สำหรับตรวจสอบ Storage และคุกกี้ในหน้าเว็บ



- **Debugger** : เป็นการหาสาเหตุที่ทำให้เกิด Error หรือ Bug ที่เกิดขึ้น หรืออีกความหมายก็คือ Tool ของ VSCode ที่ทำหน้าที่ประมวลผล Code พร้อมแสดงค่าตัวแปรที่ใช้ในปัจจุบัน ซึ่งเป็นประโยชน์ต่อการหาสาเหตุของ Bug หรือ Error ที่เกิดขึ้นได้นั่นเอง

3. สร้าง Project Solution โดยใช้ JavaScript Frameworks ใดก็ได้ โดยเกณฑ์พิจารณามี ดังนี้
  - a. มีการแสดง Theme/Template/Layout ที่เหมาะสม
  - b. มีการทำ Action ต่าง ๆ เช่น Create, Update, Delete, Loading, Swipe เป็นต้น
  - c. มีการเชื่อมต่อ Api โดยใช้ public api จากที่ใดก็ได้ เช่น <https://any-api.com/>
  - d. นำโค้ดขึ้น github และอัปเดตหน้าจอสําหรับ Presentation ความยาวไม่เกิน 10 นาที



สำหรับ Project ที่ผมเลือกใช้ คือ Project ร้านค้าขายสินค้าต่างๆ แสดงออกมาเป็นหน้าเมนูที่สามารถค้นหา เพิ่มสินค้าใหม่ กดเข้าไปดูรายละเอียด แก้ไขและลบสินค้าได้

Github Repository : <https://github.com/KonlawatPach/cnx-devsoft-test>

Clip Presentation :

<https://drive.google.com/drive/folders/1vz8Y3uLMRNkrmZpz0pHeOOUe8SoFpjwy?usp=sharing>

JavaScript Framework : React.js + Vite.js (เดิมถนัด Angular เป็นหลัก แต่เล็งเห็นว่าทางบริษัทใช้ React กับ Vue เป็นหลัก จึงเป็นการฝึกใช้ไปในตัวครับ), Bootstrap 5, SCSS และ SweetAlert2

Action ในหน้าเว็บ : Select, Insert, Update และ Delete

Api ที่ใช้ : <https://fakestoreapi.com/docs> (Api ร้านค้าให้ลองใช้ แต่การ Insert Update Delete จะไม่มีผลกับสินค้าในร้านค้าจริงๆ แต่จะมี Response ส่งกลับให้)