

Q: Folder Structure ที่จะใช้ในการจัดวาง

A:

/components ← เก็บ component ที่จะมีการเรียกใช้กันบ่อย ๆ ในหลาย ๆ หน้า

  /Card

    index.js

    style.module.css

/styles ← เก็บ global styles ที่ทุก ๆ หน้าจำเป็นต้องใช้

/views ← เก็บหน้าหลัก ๆ รวมถึง component ที่ใช้แค่เฉพาะ ๆ หน้านั้นไว้ด้วย

  /User

    /components

      /idcard ← เก็บ component ที่ใช้สำหรับกรอกฟอร์มเลขประชาชน มีหน้าข้อมูลส่วนตัวเท่านั้นที่ใช้

        index.js

        style.module.css

      /imageupload ← เก็บ component ที่ใช้สำหรับอัปโหลดรูปถ่ายหน้าตรงแบบ Square Resolution

        index.js

        style.module.css

      ...

      ...

      ...

    index.js

    style.module.css

  /Running

    ...

    ...

  /Emergency

    ...

    ...

  /Medical

    ...

    ...

Q: Library ที่คาดว่าจะใช้สำหรับ Project นี้ พร้อมทั้งระบุด้วยว่าทำไมถึงเลือกใช้ เช่น ใช้ lodash สำหรับการจัดการข้อมูล

A: ติดตั้งโปรเจก react แบบเริ่มต้นด้วย create-react-app จากนั้นเพิ่ม lib ที่จะใช้ดังนี้

- Reactstrap : เป็น component library สำหรับ react ที่สวยดี ใช้สำหรับทำให้ component ในโปรเจกมี theme คล้าย ๆ กัน
- TailwindCSS : ถัดเพิ่ม css ลงไปใน className เลย และเป็น lib ที่ใช้ง่าย เรียนรู้ได้เร็ว
- React router : จะแบ่งแต่ละ step เป็นหน้า ๆ เพราะฉะนั้นต้องมีการใช้ routing
- Redux : หน้าสุดท้ายต้องมีการส่งข้อมูลไปที่ server และจำเป็นต้องใช้ข้อมูลจากหน้าก่อน ๆ นี้ จึงใช้ Redux เพื่อเก็บข้อมูลของหน้าก่อน ๆ เพื่อความสะดวกมากขึ้น

Q: จะต้องจัดการ State ของข้อมูลอย่างไร เพื่อรองรับกับโจทย์นี้ได้

A:

- ในแต่ละหน้าที่เก็บ state ของฟอร์มในหน้าตัวเองไว้
- เมื่อกด next ไปหน้าถัดไป หรือว่า กด save draft เพื่อเก็บข้อมูลปัจจุบัน ก็ให้ไปเก็บที่ redux แต่สำหรับกรณี save draft นอกจากจะเก็บใน redux แล้ว ก็ต้องยังไปเก็บที่ server ด้วย
- ทำแบบนี้สำหรับทุก ๆ หน้าก่อนถึงหน้า confirm
- เมื่อถึงหน้า confirm ซึ่งเป็นหน้าสุดท้าย ให้ทำการยิงข้อมูลที่เก็บไว้จาก redux ออกมาแสดง เพื่อให้ผู้ใช้ตรวจสอบความถูกต้อง ก่อนจะยิงข้อมูลทั้งหมดไปที่ server

Q: จะใช้วิธีใดในการ Validate Form สำหรับการ Save Draft และสำหรับการ Submit

A:

- จะเขียน validate logic ของฟอร์มนั้น ๆ ขึ้นมาเองในหน้านั้น ๆ เลย
- เมื่อผู้ใช้กรอกข้อมูลครบถ้วนตามแบบฟอร์ม ก่อนจะไปหน้าถัดไป หรือ save draft จะต้องผ่านการ validate ก่อน
- ถ้าเกิด validate ไม่ผ่าน ก็ยังอยู่ในหน้าเดิมพร้อมกับแสดง error ที่ไม่ผ่านไว้ได้แต่ละ input ด้วย
- ถ้าเกิด validate ผ่าน ก็จะสามารถไปหน้าถัดไป หรือกด save draft ได้