

รายงาน

Assignment 2: Experiencing the usage of API & framework

รายวิชา

242-420 WEB ENGINEER & APPLICATIONS

เสนอ

ผู้ช่วยศาสตราจารย์ ดร. สุนทร วิฑูรพจน์

จัดทำโดย

นายเฉลิม แซ่หวาง	5810110058
นายธีรวัช รียา	5810110160
นายธนพนธ์ ท่อประสิทธิ์	5810110451

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์

การพัฒนาแอปพลิเคชัน

ในการพัฒนาแอปพลิเคชันนี้จะใช้ Vue2 และ Firebase โดยมีเหตุผลดังนี้

Vue2 เป็น Framework สำหรับการพัฒนา User Interface ที่ง่ายต่อการใช้, ทรงพลัง และมีสื่อการสอนพร้อม โดยไลบรารีหลักของ Vue2 จะให้ความสนใจไปที่การจัดการ View layer เป็นหลัก ทำให้ง่ายต่อการนำไปใช้หรือใช้ร่วมกับ API อื่น

Firebase เป็น platform สำหรับพัฒนา โมบาย แอปพลิเคชัน และ เว็บแอปพลิเคชัน โดย Firebase ทำให้เราสามารถพัฒนาแอปพลิเคชันได้ โดยไม่ต้องทำ Server-side programming โดยจะสามารถใช้ API ต่าง ๆ ที่ Firebase มี เช่น Real-time Database หรือ การยืนยันตัวตนผู้ใช้

ในการพัฒนาจะเริ่มจากการติดตั้งเครื่องมือต่าง ๆ ที่จำเป็นให้พร้อมก่อน โดยสิ่งแรกที่เราจะใช้คือ Vue-cli ซึ่งเป็นเครื่องมือสำหรับใช้งาน Vue2 โดยฟีเจอร์หลักอย่างหนึ่งที่เราจะใช้คือ การสร้างโปรเจกต์โดยอิงจากเทมเพลต โดยเทมเพลตที่จะใช้มีชื่อว่า webpack

เมื่อทำการสร้างโปรเจกต์เสร็จแล้ว จะเริ่มทำในส่วนของการทำ Authentication ซึ่งจะทำทั้งหน้า Login และหน้า Sign-up โดยในการทำหน้า Login เริ่มจากการสร้าง Component ใน src/components แล้วเขียน input field และ button ง่าย ๆ ไว้ ในการจัดการ pathing ของแอป จะใช้ตัว Vue-router โดยตัว Vue-router จะทำการ map คอมโพเนนต์ และ path ไว้ ซึ่งจะมีตัว Router-View มาทำการ render ให้

ในการทำหน้า Sign-up จะคล้ายกับการทำหน้า Login โดยเริ่มจากการสร้าง คอมโพเนนต์ Sign-up ก่อน ซึ่งจะมี input field และ button ไว้ แล้วนำคอมโพเนนต์ที่สร้างไว้ ใส่ใน Vue-router

เมื่อมีทั้ง 2 หน้าสำหรับ Login และ Sign-up แล้ว จะใช้ router-link ในการเชื่อมโยงทั้ง 2 หน้า ให้ไปหากันได้ โดย router-link เป็นคอมโพเนนต์ที่ใช้จัดการการเรื่องการเคลื่อนย้ายของ user

ในการทำ Authentication จะใช้งาน API ของ Firebase โดยเริ่มจากการสร้าง project Firebase ไว้ แล้วทำการ install และ import โมดูล Firebase ลงในโปรเจกต์ของเรา จากนั้น Enable การ Sign-in ใน Firebase Console แล้วทำการนำข้อมูลที่ user กรอกในหน้า Sign-in มาสร้าง account ใหม่โดยใช้ API ของ Firebase และทำการแก้ไขหน้า Login ให้มีการยืนยันผู้ใช้ผ่านการใช้ API ของ Firebase เช่นเดียวกัน

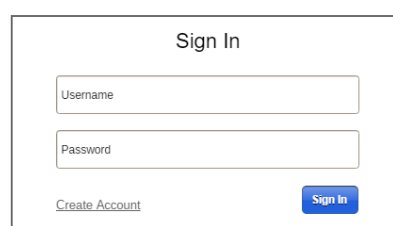
การออกแบบหน้าแอปพลิเคชัน

การออกแบบหน้าแอปพลิเคชันเป็นการออกแบบเบื้องต้นว่าเว็บที่จะสร้างจะมีหน้าตาอย่างไรโดยใช้ทฤษฎีต่าง ๆ เช่น ทฤษฎีสี ทฤษฎีการแบ่งโครงสร้างของหน้าเว็บ ทฤษฎีการใช้ตัวอักษร โดยหลัก ๆ จะใช้การออกแบบของ material design ซึ่งเห็นได้ชัดจากการจัดแสงและเงาจะทำให้เห็นถึงการแบ่งระดับชั้นของ components ได้อย่างชัดเจนรวมถึงการคำนึงถึงผู้ใช้งาน (User Experience) ในการดูเนื้อหาต่าง ๆ ภายในหน้าเว็บโดยมีตัวอย่างการออกแบบหน้าเว็บดังต่อไปนี้

หน้า Login

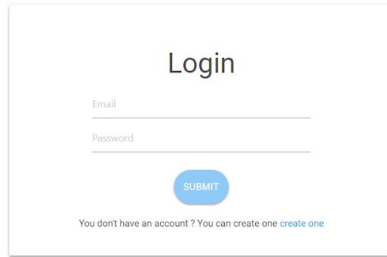
ในส่วนของหน้าล็อกอินจะมี Content หลักคือส่วนกล่องล็อกอินที่อยู่ตรงกลางหน้าเท่านั้น เพื่อให้ง่ายต่อการสังเกตของผู้ใช้ เพราะ Content ส่วนที่อยู่ตรงกลางและเป็นแค่ Content เดียวที่อยู่ใน Page

ในกล่องล็อกอินประกอบไปด้วย Text ที่เขียนว่า Login ขนาดใหญ่ เพื่อให้ผู้ใช้สังเกตเห็นได้ง่าย Text Field ที่รับข้อมูลของผู้ใช้ และ Text ที่เป็น HyperLink สีฟ้าเพื่อแสดงให้ผู้ใช้เห็นว่าสามารถกดได้ ที่เชื่อมไปยังหน้า Register และ Button สำหรับการ Submit โดยใช้สีฟ้า เพื่อสื่อถึงการเข้าถึงและสะดวกแก่ผู้ใช้ ดังรูปที่ 1



The image shows a login form titled "Sign In". It contains two input fields: "Username" and "Password". Below the "Password" field, there is a link "Create Account" and a blue button labeled "Sign In".

รูปที่ 1 หน้า Login Design

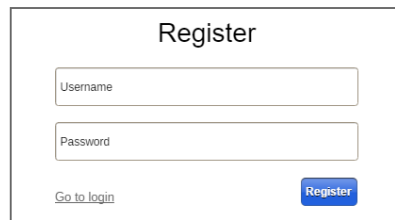


A login form with a white background and a light gray border. At the top, the word "Login" is centered in a bold, black font. Below it, there are two input fields: "Email" and "Password", each with a light gray border and a small gray label above it. A blue "SUBMIT" button is centered below the input fields. At the bottom, there is a line of text: "You don't have an account ? You can create one [create one](#)".

รูปที่ 2 หน้า Login Develop

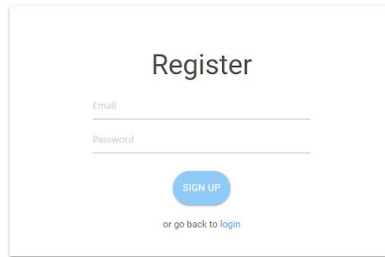
หน้า Register

ในหน้า Register จะใช้หลักการเดียวกับหน้า Login โดยมี ตำแหน่งเหมือนกับหน้า Login ดังรูปที่ 3



A register form with a white background and a light gray border. At the top, the word "Register" is centered in a bold, black font. Below it, there are two input fields: "Username" and "Password", each with a light gray border and a small gray label above it. A blue "Register" button is centered below the input fields. At the bottom left, there is a link: "[Go to login](#)".

รูปที่ 3 หน้า Register Design



Register

Email

Password

SIGN UP

or go back to login

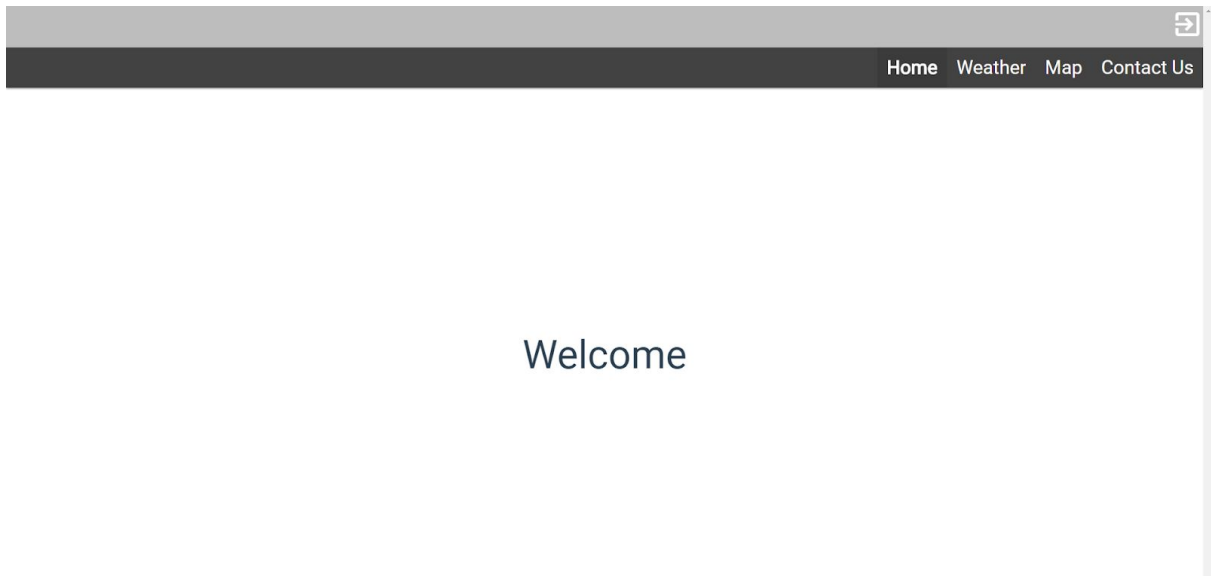
รูปที่ 4 หน้า Register Develop

หน้า Home

หน้า Home จะมี 2 Menu Bar โดย Bar เล็กจะเป็นส่วนของ Icon Logout และ Bar ใหญ่จะเป็นส่วนของ Menu ที่ประกอบด้วย Home, Map, Weather และ Contact Us โดยสีของ Font ที่ใช้จะเป็นสีขาว โดยจะตัดกับสีของ Bar ที่เป็นสีดำเพื่อสร้างความโดดเด่นในแง่ ข้อความ Menu รวมไปถึงข้อความที่หนากว่าเมนูอื่นเพื่อแสดงให้เห็นว่าผู้ใช้อยู่ในหน้านั้นๆ และเหตุผลที่ส่วนของเมนูทั้งหมดอยู่ด้านขวา เพราะต้องการให้ Content ตรงกลางเด่นขึ้นมาเมื่อผู้ใช้กวาดสายตาจากซ้ายไปขวา ดังรูปที่ 5



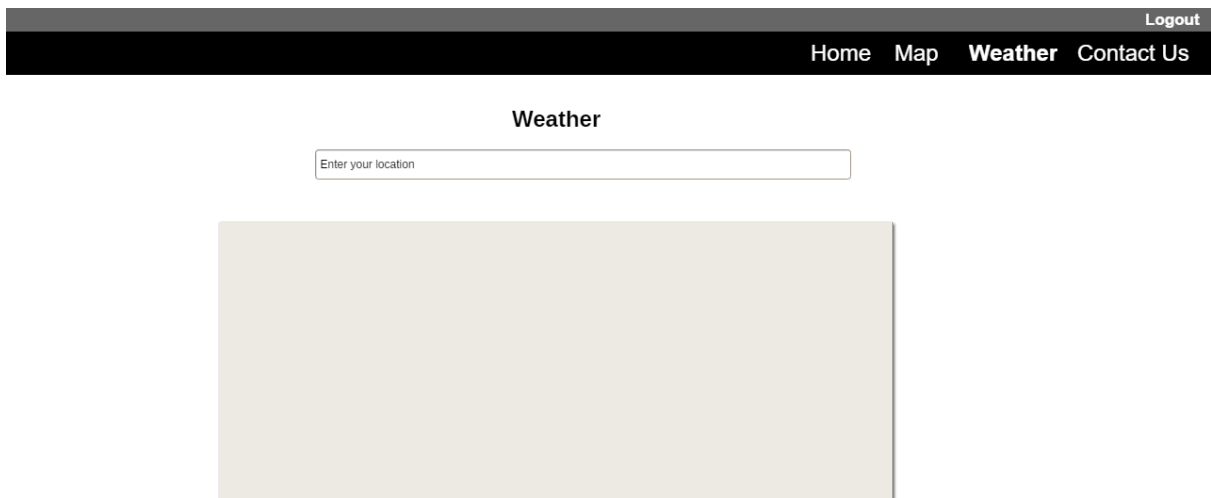
รูปที่ 5 หน้า Home Design



รูปที่ 6 หน้า Home Develop

หน้า Map และ Weather

วาง Content ของ Api ไว้ตรงกลางเพื่อที่ผู้ใช้จะได้เห็นก่อน Menu Bar โดย Content จะแบ่งเป็น 2 ส่วน ในส่วนแรกจะเป็นส่วนของ Text และ Text Field รับข้อมูลของผู้ใช้ ส่วนที่สองจะเป็นส่วนของการแสดงข้อมูลของ Api โดยทั้งหมดจะอยู่ตรงกลางเพื่อให้ง่ายต่อการอ่านข้อมูลของผู้ใช้ ดังรูปที่ 7



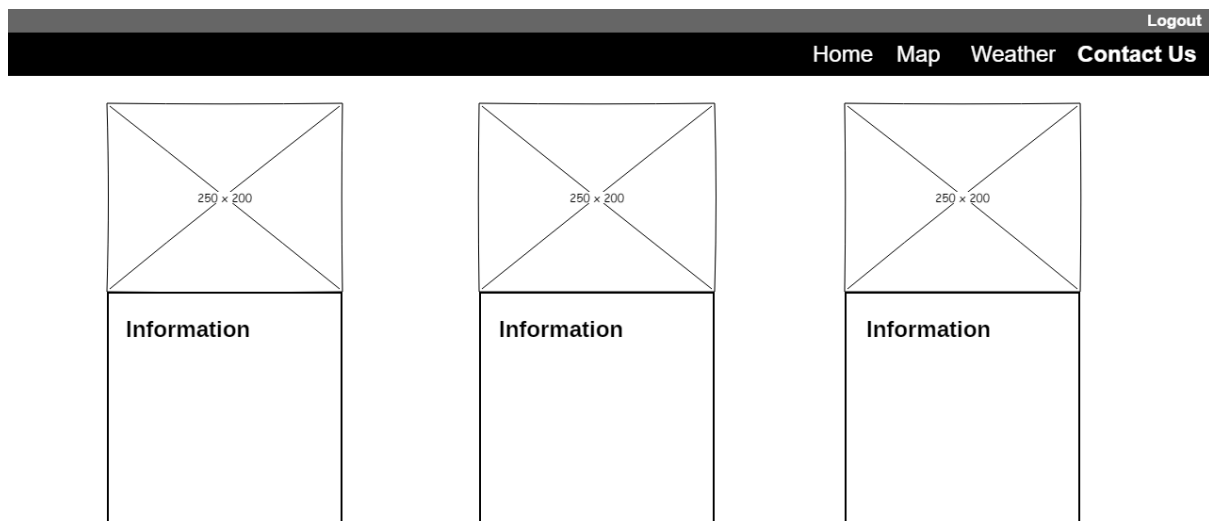
รูปที่ 7 หน้า Weather Design



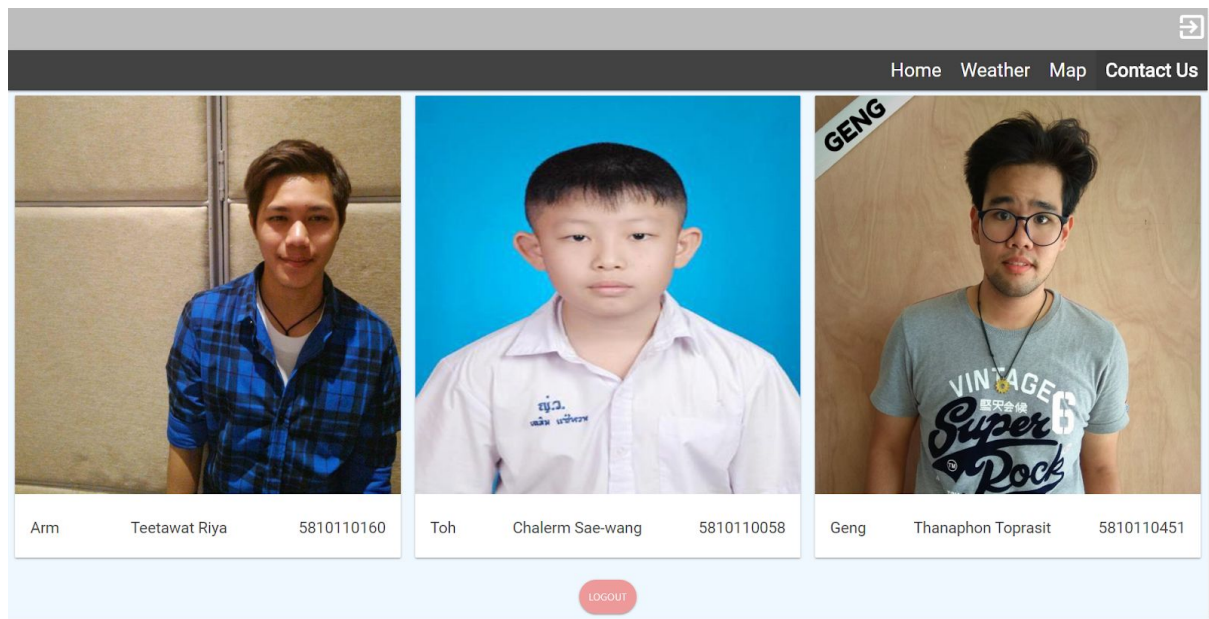
รูปที่ 8 หน้า Weather Develop

หน้า Contact Us

หน้านี้จะแบ่งส่วนของ Content เป็น 3 ส่วนโดยแต่ละส่วนจะเป็นเหมือน CardView ซึ่งทำให้ง่ายต่อการดูของผู้ใช้ ไม่ว่าจะเป็นรูปภาพหรือ ข้อมูลด้านล่าง ในส่วนของข้อมูลด้านล่างก็ยังแบ่งเป็นสามส่วนที่ชัดเจน ผู้ใช้สามารถรับรู้ได้ว่าแต่ละส่วนเป็นข้อมูลที่แตกต่างกัน ดังรูปที่ 9



รูปที่ 9 หน้า Contact Us Design



รูปที่ 10 หน้า Contact Us Develop