

ใบงานเก็บคะแนน

1.จาก ionic framework ให้นักศึกษาอธิบายแต่ละส่วนว่าทำหน้าที่อะไรบ้าง

- Ionic framework มันคือเครื่องมือที่ใช้ในการสร้าง HTML ,CSS และ JavaScript เพื่อใช้ในการสร้าง Mobile Application ซึ่งสามารถใช้งานได้ค่อนข้างง่าย และยังมีการใช้ Command-line interface(CLI) เข้ามาช่วยในการจัดการส่วนต่าง ๆ ของแอป เช่นการสร้างหน้าใหม่ หรือการเข้ามาช่วยด้านการติดตั้งให้ง่ายขึ้นได้อีกด้วยซึ่ง ionic framework เป็นเครื่องมือสร้างแอปมือถือที่สามารถสร้างที่เดียวใช้งานได้หลายระบบปฏิบัติการ ทั้งระบบปฏิบัติการ Windows ระบบปฏิบัติการ Android และระบบปฏิบัติการ IOS ซึ่งสามารถใช้งานร่วมกับ Framework อื่น ๆ ด้วย เช่น Angular เป็นต้น

การพัฒนาด้วย Ionic สามารถทำได้ 2 แบบคือใช้ Ionic studio ที่เป็น tools ของทาง ionic เอง พัฒนาหรือจะใช้ Visual Studio Code เขียนก็ได้ร่วมกับ ionic command line tools ซึ่งจะต้องมีการติดตั้ง Android Studio เพื่อใช้ ADV ในการทดสอบและการสร้างแอปออกมาเป็น APK เพื่อติดตั้งหรือทดสอบลงบนมือถือ สำหรับการติดตั้ง ionic command line จะใช้การติดตั้งผ่าน npm ของ node.js ซึ่งจะต้องมีการติดตั้งก่อน หลังจากนั้นจะสามารถใช้คำสั่ง npm install -g ionic เพื่อทำการติดตั้ง ionic command line เพื่อให้สำหรับการสร้างสร้างโปรเจก โดยการสร้างโปรเจกขึ้นมาใหม่สามารถใช้คำสั่ง ionic start myApp tabs ในกรณีที่ต้องการทำแอปพลิเคชันที่ใช้ฟิเจอร์ tabs ซึ่งตัว ionic สามารถสร้างได้ 3 รูปแบบคือ blank, tabs และ sidemenu

- "โครงสร้างโครงการ" หมายถึงการรวมกันของโครงสร้างตัวระบุทั้งหมดที่ใช้ในโครงการ สำหรับออบเจกต์เพจอุปกรณ์และฟังก์ชัน อ็อบเจกต์ทั้งหมดในโปรเจกต์ (เพจอุปกรณ์และฟังก์ชัน) ต้องถูกระบุและวางไว้ในโครงสร้างลำดับชั้นภายในโปรเจกต์ โครงสร้างโครงการที่จัดตามลำดับชั้นช่วยลดการระบุและตำแหน่งของแต่ละออบเจกต์ภายในโปรเจกต์ โปรเจกต์มีโครงสร้างโดยใช้บล็อกตัวระบุที่คุณกำหนดใน EPLAN โดยใช้สกีม
- Navigation การนำทางในไอออนิกทำงานเหมือนสแต็กทั่วไปโดยเราจะpushสร้างหน้าใหม่ขึ้นมาที่ด้านบนสุดของสแต็กซึ่งจะพาเราไปข้างหน้าในแอปและแสดงปุ่มย้อนกลับ หากต้องการย้อนกลับเราจะpopปิดหน้าบนสุด เนื่องจากเราตั้งค่าthis.navCtrlในตัวสร้างเราจึง

สามารถเรียกthis.navCtrl.push()และส่งผ่านไปยังหน้าที่เราต้องการนำทางไปได้ นอกจากนี้เรายังสามารถส่งผ่านวัตถุที่มีข้อมูลที่เราต้องการส่งผ่านไปยังหน้าที่กำลังนำทางไป การใช้pushเพื่อนำทางไปยังหน้าใหม่นั้นทำได้ง่าย แต่ระบบการนำทางของ Ionic นั้นมีความยืดหยุ่นมาก ดูเอกสารการนำทางเพื่อดูตัวอย่างการนำทางขั้นสูงเพิ่มเติม

- Data Integration ในองค์กร

กระบวนการรวบรวมข้อมูลขององค์กร สามารถเกิดขึ้นได้ภายใต้ 4 ขั้นตอนในระบบไอที (Information Technology) คือdata, application, business process และ user interaction เทคโนโลยีและเครื่องมือ หลายชนิดที่มีความเหมาะสมกับงานนี้แต่เพื่อให้การจัดหมวดหมู่และจัดการง่าย ขึ้น เราจะจำแนกเครื่องมือต่างๆ ออกได้เป็น 4 ระดับดัง ต่อไปนี้ Data Integration: ให้ ประโยชน์ในการแสดงภาพรวมข้อมูลของธุรกิจ ที่กระจายอยู่ในส่วนต่างๆ ในองค์กร ภาพรวมวงกว้าง ณ จุดเดี่ยวนี้อาจสร้างได้โดยใช้เครื่องมือ เครื่องมือ เทคนิค และ เทคโนโลยี แตกต่างชนิดกัน มันอาจจะเป็นภาพรวมทางกายภาพของข้อมูลที่ได้จากแหล่งข้อมูลหลายๆที่รวมกัน หรือ มันอาจจะเป็นข้อมูลที่มีการรวบรวมเสมือน(Virtual federate) ให้เสมือนว่าข้อมูลมีการอยู่ร่วมกัน แต่ว่าแท้จริงแล้วทางกายภาพอยู่กันคนละที่ก็เป็นได้

- UX ย่อมาจาก User Experience แปลเป็นภาษาไทยได้ว่า “ประสบการณ์ของผู้ใช้งาน” เป็นส่วนที่ออกแบบกระบวนการใช้งานของสินค้าและบริการให้ผู้ใช้พึงพอใจ เช่น ใช้งานง่าย มีลำดับขั้นตอนชัดเจน โดยสามารถทราบ Feedback ได้จากกลุ่มตัวอย่างที่ทดลองให้ใช้สินค้าหรือบริการ เพื่อนำมาพัฒนาสินค้าหรือบริการได้ต่อไป

- UI ย่อมาจาก User Interface แปลเป็นภาษาไทยได้ว่า “ส่วนติดต่อระหว่างผู้ใช้กับระบบ” เป็นส่วนที่เติมเต็ม UX ให้มีความสวยงาม เช่น เรื่องการจัดวางองค์ประกอบ ในด้านการใช้งานเว็บไซต์ก็หมายถึงการแสดงผลทุกอย่าง หรือ หมายถึงหน้าจอสำหรับผู้ใช้งาน เช่น เทคนิคการวางตัวอักษรให้ดูโดดเด่น ช่องว่างระหว่างส่วนประกอบต่างๆ เช่น ปุ่ม Call to Action (CTA) ขนาดฟอนท์ของตัวอักษร เป็นต้น

กล่าวโดยสรุปคือ UX จะออกแบบในส่วนของการใช้งานให้มีประโยชน์ ใช้งานง่าย สะดวก ส่วน UI จะออกแบบในด้านความสวยงาม

2.สร้างแบบฟอร์ม login โดยใช้ Ionic angular

Code หน้า login และหน้า Register

```
src > app > home > <> home.page.html > <> ion-header > <> ion-toolbar
1  <ion-header [translucent]="true">
2    <ion-toolbar color="primary">
3      <ion-title>
4        <center>LOGIN</center>
5      </ion-title>
6    </ion-toolbar>
7  </ion-header>
8
9  <ion-content [fullscreen]="true">
10    <ion-header collapse="condense">
11      <ion-toolbar color="primary">
12        <ion-title size="large"><center>LOGIN</center></ion-title>
13      </ion-toolbar>
14    </ion-header>
15
16    <div>
17      <ion-item>
18        <ion-label>Username</ion-label>
19        <ion-input></ion-input>
20      </ion-item>
21    </div>
22
23    <div>
24      <ion-item>
25        <ion-label>Password</ion-label>
26        <ion-input></ion-input>
27      </ion-item>
28    </div>
```

```
29 <center>  
30   <ion-button size="small" color="dark"class="mr-1">Login</ion-button>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~<br>31   <ion-button size="small" color="dark"class="mr-1" (click) = "register()">Register</ion-button><br>32 </center><br>33 ~<br>34 ~<br>35 </ion-content><br>36
```

```

src > app > register > register.page.html > ion-header > ion-toolbar > ion-title
1  <ion-header>
2    <ion-toolbar>
3      <input type=button onClick='window.history.back()' value='back'>
4      <ion-title>register</ion-title>
5    </ion-toolbar>
6  </ion-header>
7
8  <ion-content>
9    <div>
10     <ion-item>
11       <ion-label>Username</ion-label>
12       <ion-input></ion-input>
13     </ion-item>
14   </div>
15
16   <div>
17     <ion-item>
18       <ion-label>Password</ion-label>
19       <ion-input></ion-input>
20     </ion-item>
21   </div>
22
23   <div>
24     <ion-item>
25       <ion-label>Confirm Password</ion-label>
26       <ion-input></ion-input>
27     </ion-item>
28   </div>

```

[illegible]

```
src > app > home > TS home.page.ts >  HomePage >  register
```

```
1  import { Component } from '@angular/core';
2  import { Router } from '@angular/router';
3
4  @Component({
5    selector: 'app-home',
6    templateUrl: 'home.page.html',
7    styleUrls: ['home.page.scss'],
8  })
9  export class HomePage {
10     constructor(private Route: Router) {}
11     register() {
12       this.Route.navigate(['/register']);
13     }
14   }
15 }
```

```
home.page.html register.page.html TS home.page.ts home.page.scss
src > app > home > home.page.scss > .mr-5
1  #container {
2
3      text-align: center;
4
5      position: absolute;
6      left: 0;
7      right: 0;
8      top: 50%;
9      transform: translateY(-50%);
10 }
11 #container strong {
12     font-size: 20px;
13     line-height: 26px;
14 }
15 #container p {
16     font-size: 16px;
17     line-height: 22px;
18
19     color: #8c8c8c;
20     margin: 0;
21 }
22 #container a {
23     text-decoration: none;
24 }
25 .mr-5 {
26     margin-right: 30px;
27     margin-top: 25px;
28 }
```

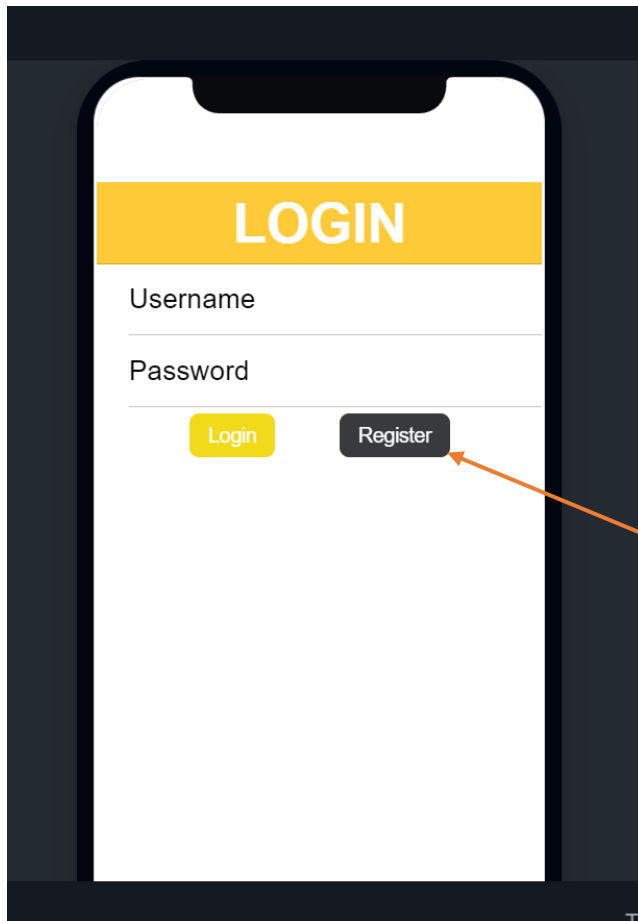
หน้า login

LOGIN

Username

Password

Login Register

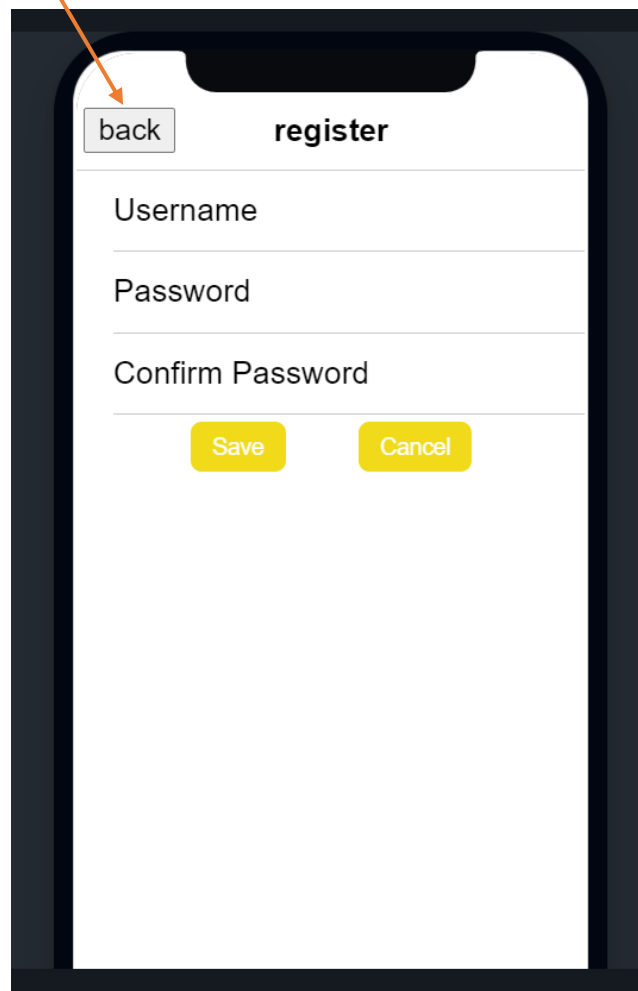


A mobile app login screen with a dark blue header and a white body. The header has a yellow bar with the word "LOGIN" in white. Below the header, there are two input fields labeled "Username" and "Password". At the bottom, there are two buttons: a yellow "Login" button and a dark grey "Register" button.

กดปุ่มนี้เพื่อไปหน้า Register

กด ปุ่มนี้เพื่อกลับไปหน้า LOGIN

หน้า ReGister



A mobile app register screen with a dark blue header and a white body. The header has a grey "back" button and the word "register" in white. Below the header, there are three input fields labeled "Username", "Password", and "Confirm Password". At the bottom, there are two yellow buttons: "Save" and "Cancel".