

IONIC

Ionic เป็นเฟรมเวิร์กตัวหนึ่งที่สามารถสร้างเป็นแอปพลิเคชันได้หลายแพลตฟอร์ม สามารถทำงานร่วมกับเฟรมเวิร์กตัวอื่นได้เช่น Angular และ React

Ionic มีพื้นการใช้คอมโพเนนต์ UI ในการสร้างแอปพลิเคชัน UI มีมากมายให้เลือกใช้ คล้ายกับ UI ที่มีใน Bootstrap เช่น Button, Card, Model, Navigation

Capacitor เป็นวิวัฒนาการของ Ionic ที่เน้นการทำงานบนเว็บแอปพลิเคชันของอุปกรณ์โทรศัพท์เคลื่อนที่ บนแพลตฟอร์มต่างๆ ไม่ว่าจะเป็น Android และ IOS ใน Ionic รุ่นใหม่จึงผนวก Capacitor ได้ด้วยกัน

ติดตั้ง

การติดตั้งจำเป็นต้องใช้ Nodejs ซึ่งเป็น JavaScript Network Application ทำงานเป็นส่วนเซิร์ฟเวอร์ จึงต้องติดตั้ง Nodejs ที่เว็บไซต์ <https://nodejs.org/en/download/>

```
>npm install -g @ionic/cli
    -g หมายถึงติดตั้งระดับ Global

>npm install @ionic/lab
    ไม่ต้องใส่ -g เพื่อติดตั้ง ionic
    lab หมายถึงติดตั้ง ตัวจำลอง iOS และ Android
    ionic/lab ได้หยุดพัฒนาตั้งแต่ปี 2017 (version 3.1.7)
```

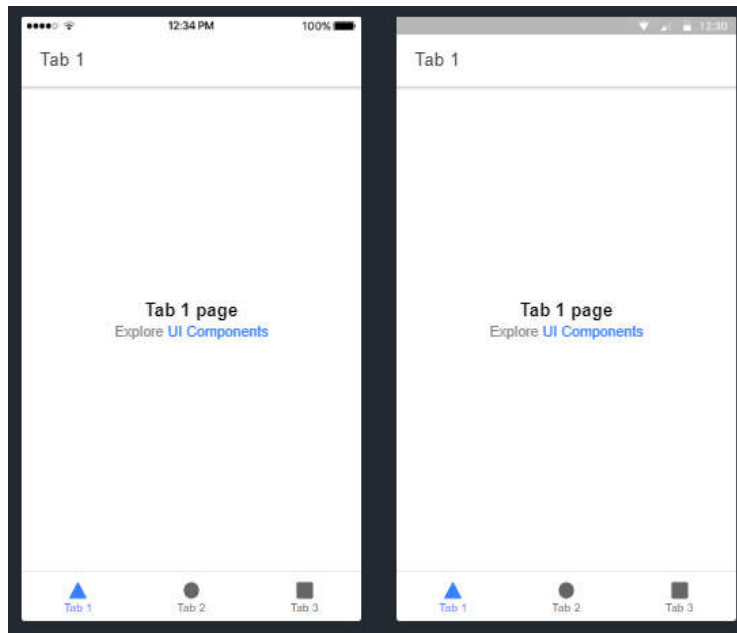
เริ่มต้นกับ App Tab

```
>ionic start myApp tabs --capacitor --type=angular
```

คำสั่งนี้เป็นการสร้าง แอปฯ มีการจัดวางแบบ tabs ผนวก capacitor เพื่อให้ทำงานแบบข้ามแพลตฟอร์มได้ และเลือกชนิดการทำงานกับ Angular ในระหว่างสร้างแอปฯ จะมีคำถามให้ตอบ Yes หรือ No เช่น คำถามสุดท้ายให้สร้างบัญชี ionic ให้ตอบ No เพื่อไม่สร้างบัญชี ionic

รันแอปฯ

```
>cd myApp
>ionic serve --lab
```



รูป 1 ผลการแสดงผลแรกของแอปฯ ผ่าน localhost:8200 ซ้าย IOS, ขวา Android

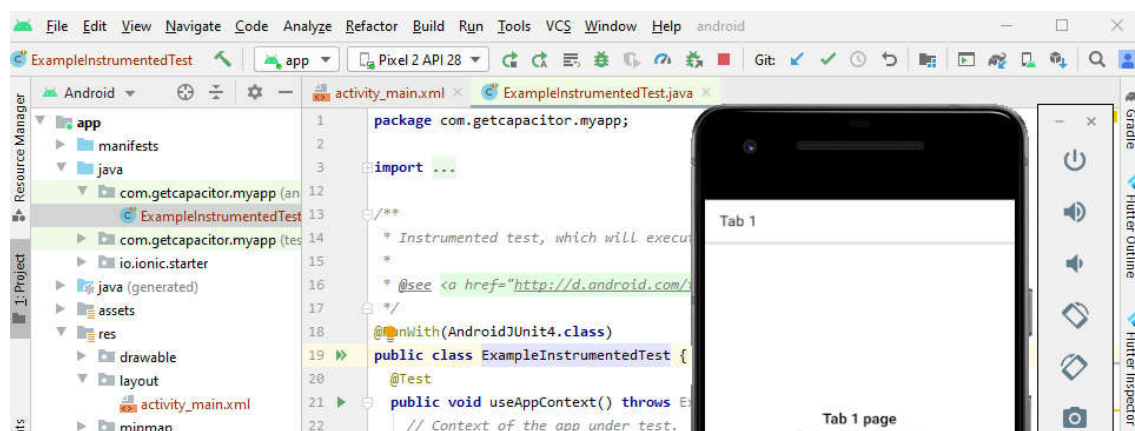
ใช้งานบน Android Studio

- >ionic build
- >npx cap copy
- >npx cap add android
- >npx cap open android

หลังจากคำสั่งสุดท้ายนี้ โปรแกรม **Android Studio** จะเปิดทำงาน แล้วจะใช้เวลาอีกหลายนาทีเพื่อดาวน์โหลดและสร้างส่วนประกอบต่างๆ

กรณีใช้กับ ios

- >npx cap add ios
- >npx cap open ios



รูป 2 ผลการทดสอบบน Android Studio และ Virtual Device

การจัดวางตำแหน่งส่วนประกอบหน้าจอ (Layout)

IONIC มีโครงสร้าง (Structure) หลัก 4 แบบ

- Header/content/footer
- Tab
- Menu
- Split Pane

นอกจากนี้ในส่วนโครงสร้าง โดยเฉพาะส่วนเนื้อหา (content) หรือภายในสามารถสร้างเป็นระบบกริด (Grid) เป็นการจัดวางที่ทำให้การแสดงผลมีความยืดหยุ่น การจัดวางแบบนี้ประกอบด้วย 3 ส่วนคือ

- grid
- row(s)
- column(s)

Header/content/footer

แอปฯโดยทั่วไปจะมีส่วนหัว (head) และส่วนเนื้อหา (content) ในส่วนหัวจะอยู่ด้านบนเสมอ แม้ว่าจะเขียนไม่อยู่ในลำดับบนก็ตาม ส่วน `<ion-app>` ทำหน้าที่แทนขอบเขตทั้งหมดของแอปฯ ดังนั้นการแสดงผลทั้งหมดจะอยู่ในส่วนนี้ ทดลองแทนทั้งหมดของ Code 1 ใน `src/app.component.html`

Code 1.

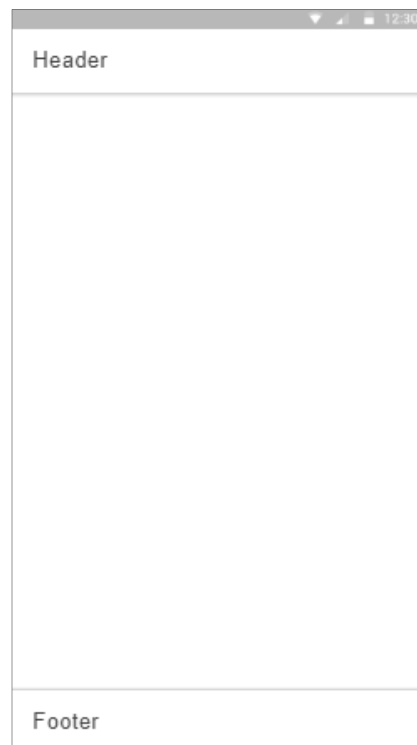
```
<ion-app>
  <ion-header>
    <ion-toolbar>
      <ion-title>Header</ion-title>
    </ion-toolbar>
  </ion-header>

  <ion-content class="ion-padding">
    <h1>Main Content</h1>
  </ion-content>
</ion-app>
```

การเพิ่ม ส่วนท้าย `<ion-footer>` ก็ต้องอยู่ในส่วน `<ion-app>` เช่นเดียวกับส่วนหัว `<ion-footer>` ไม่จำเป็นต้องอยู่ในลำดับท้ายของการเขียนโปรแกรม

Code 2.

```
<ion-footer>
  <ion-toolbar>
    <ion-title>Footer</ion-title>
  </ion-toolbar>
</ion-footer>
```



รูป 3 ผลการแสดงผลแบบ header/content/footer

Tab

โครงสร้างแบบแถบ (Tab) ประกอบด้วย แถบแนวนอน ในแต่ละแถบสามารถเก็บข้อมูล หรือ การนำทาง ด้วย `ion-router-outlet` หรือ `ion-nav`

Code 3.

```
<ion-app>
  <ion-tabs>
    <ion-tab tab="home">
      <h1>Home Content</h1>
    </ion-tab>
    <ion-tab tab="settings">
      <h1>Settings Content</h1>
    </ion-tab>

    <ion-tab-bar slot="bottom">
      <ion-tab-button tab="home">
        <ion-label>Home</ion-label>
        <ion-icon name="home"></ion-icon>
      </ion-tab-button>
      <ion-tab-button tab="settings">
        <ion-label>Settings</ion-label>
        <ion-icon name="settings"></ion-icon>
      </ion-tab-button>
    </ion-tab-bar>
  </ion-tabs>
</ion-app>
```

Menu

รูปแบบมาตรฐานของเมนูจะอยู่ด้านซ้ายที่เลื่อนเข้าออกได้ ภายในเมนูประกอบด้วยตัวนำทางต่างๆ และยังประกอบด้วยข้อมูลเนื้อหาได้ด้วย

Code 4.

```
<ion-app>
  <ion-menu content-id="main-content">
    <ion-header>
      <ion-toolbar color="primary">
        <ion-title>Menu</ion-title>
      </ion-toolbar>
    </ion-header>

    <ion-content>
      <ion-list>
        <ion-list-header>
          Navigate
        </ion-list-header>
        <ion-menu-toggle auto-hide="false">
          <ion-item button>
            <ion-icon slot="start" name='home'></ion-icon>
            <ion-label>
              Home
            </ion-label>
          </ion-item>
        </ion-menu-toggle>
      </ion-list>
    </ion-content>
  </ion-menu>
</ion-app>
```

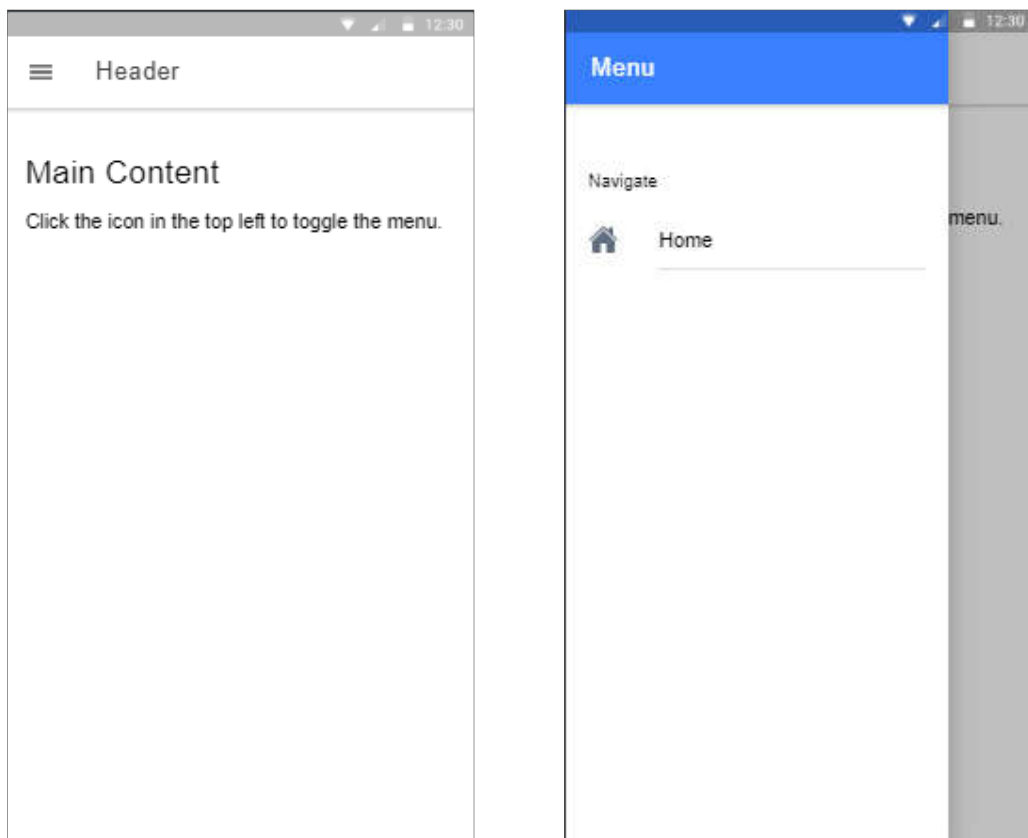
```

    </ion-content>
  </ion-menu>

  <ion-page class="ion-page" id="main-content">
    <ion-header>
      <ion-toolbar>
        <ion-buttons slot="start">
          <ion-menu-toggle>
            <ion-button>
              <ion-icon slot="icon-only" name="menu"></ion-icon>
            </ion-button>
          </ion-menu-toggle>
        </ion-buttons>
        <ion-title>Header</ion-title>
      </ion-toolbar>
    </ion-header>
    <ion-content class="ion-padding">
      <h1>Main Content</h1>
      <p>Click the icon in the top left to toggle the menu.</p>
    </ion-content>
  </ion-page>
</ion-app>

<ion-menu-controller></ion-menu-controller>

```



รูป 4 ผลการแสดงผลแบบ เมนู หน้าเริ่มต้น (ซ้าย) เมนูที่เปิดจากทางซ้าย (ขวา)

Split Pane

Split Pane มีโครงสร้างคล้ายกับ Menu แต่มีความซับซ้อนมากกว่า เพราะสามารถแสดงผลได้หลายขนาดหน้าจอขึ้นอยู่กับว่า หน้าจะใหญ่หรือเล็ก กรณีขนาดหน้าจอเล็กจะซ่อนส่วนที่เป็นเมนูได้ไว้ ทำให้มีโครงสร้างเหมือนกับ Menu แต่ถ้าเป็นจอขนาดใหญ่ จะแสดงทั้งสองส่วน คือ ส่วนเนื้อหา และเมนู

โดยหน้าจอขนาดใหญ่ จะมีขนาดกว้างกว่า 768px ซึ่งสามารถปรับแต่งได้ โดยกำหนดคุณสมบัติ ใน `<ion-split-pane>` ด้วย `when` โดยคำบรรยาย `when` กำหนดเท่ากับ `md` ตัวอย่างต่อไปนี้กำหนดให้ `when` มีค่าเท่ากับ `sm` ซึ่งเป็นขนาด 576px ซึ่งเป็นจอขนาดเล็ก ทำให้มีการซ่อนส่วนของเมนู

Code 5.

```
<ion-app>
  <ion-split-pane when="sm" content-id="main-content">
    <ion-menu content-id="main-content">
      <ion-header>
        <ion-toolbar color="primary">
          <ion-title>Menu</ion-title>
        </ion-toolbar>
      </ion-header>

      <ion-content>
        <ion-list>
          <ion-list-header>
            Navigate
          </ion-list-header>
          <ion-menu-toggle auto-hide="false">
            <ion-item button>
              <ion-icon slot="start" name='home'></ion-icon>
              <ion-label>
                Home
              </ion-label>
            </ion-item>
          </ion-menu-toggle>
        </ion-list>
      </ion-content>
    </ion-menu>

    <div class="ion-page" id="main-content">
      <ion-header>
        <ion-toolbar>
          <ion-buttons slot="start">
            <ion-menu-toggle>
              <ion-button>
                <ion-icon slot="icon-only" name="menu"></ion-icon>
              </ion-button>
            </ion-menu-toggle>
          </ion-buttons>
          <ion-title>Header</ion-title>
        </ion-toolbar>
      </ion-header>
```

```

        <ion-content class="ion-padding">
          <h1>Main Content</h1>
        </ion-content>
      </div>

    </ion-split-pane>
  </ion-app>

```

Grid

ในการจัดการเนื้อหาเพื่อจัดวาง โดยใช้ระบบกริด จะช่วยให้การจัดวางมีความยืดหยุ่นมากขึ้นด้วยการปรับแต่งของแถว และคอลัมน์ในรูปแบบระบบกริดแทนที่จะเป็นตาราง โดยคอลัมน์สามารถขยายเต็มขนาดแถวได้ หนึ่งแถวแบ่งได้เป็น 12 ช่วงกว้างย่อย ขนาดของคอลัมน์จึงเลือกความกว้างตามระดับ 12 ช่วงกว้างนี้

ตัวอย่างต่อไปนี้จะใช้การวางกริดในโครงสร้างแบบ head/content/footer โดยมีกริดแบบ หนึ่งแถว สามคอลัมน์

Code 6.

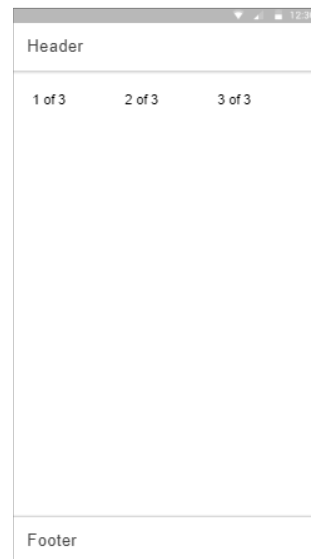
```

<ion-app>
  <ion-header>
    <ion-toolbar>
      <ion-title>Header</ion-title>
    </ion-toolbar>
  </ion-header>

  <ion-content class="ion-padding">
    <ion-grid>
      <ion-row>
        <ion-col> <div> 1 of 3 </div> </ion-col>
        <ion-col> <div> 2 of 3 </div> </ion-col>
        <ion-col> <div> 3 of 3 </div> </ion-col>
      </ion-row>
    </ion-grid>
  </ion-content>

  <ion-footer>
    <ion-toolbar>
      <ion-title>Footer</ion-title>
    </ion-toolbar>
  </ion-footer>
</ion-app>

```



รูป 5 ผลการแสดงผลแบบกริด

ขนาดกริดตามค่าปริยายจะมีความกว้างเต็มพื้นที่ (100%) การปรับแต่งจุดแบ่งความกว้าง (breakpoint) กำหนดในรูปแบบ: --ion-grid-width-{breakpoint} ตามรูปแบบ CSS

ตาราง 1 ใช้ชื่อ(xs, sm, md, lg, xl) แทนความกว้างในรูปแบบเดียวกับ CSS เพื่อแทนขนาดของกริด เมื่อเขียนตามรูปแบบ จะเขียนได้คือ --ion-grid-width-lg

ตาราง 1 ขนาดกริด

ชื่อความกว้าง	ขนาด	คำอธิบาย
xs (extra small)	100%	ไม่ต้องกำหนดเมื่อใช้กับหน้าจอขนาด xs
sm (small)	540px	กริดกว้างถึง 540px (min-width: 576px)
md (middle)	720px	กริดกว้างถึง 720px (min-width: 768px)
lg (large)	960px	กริดกว้างถึง 960px (min-width: 992px)
xl (extra large)	1140px	กริดกว้างถึง 1140px (min-width: 1200px)

สำรวจแอปพลิเคชันต้นแบบ

จากที่ได้สร้าง myApp และเลือกการจัดวางแบบ tab ซึ่ง Ionic ได้สร้างแอปพลิเคชันต้นแบบมาให้

UI Component

Ionic มี UI มากมาย ตัวแรกที่น่าจะนำการใช้คือ Action Sheet

Native functionality

Theming

ข้อมูลเพิ่มเติม

ionic: <https://ionicframework.com/>

capacitor: <https://capacitorjs.com/>