

# Foldable/dual-screen form factors

CS50



**Craig Dunn**

Microsoft Surface Duo  
Developer Experience Team



**Justin Willis**

Microsoft Edge Team

# Surface Duo 2

- Two screens
- Front and back cameras
- Android 11
- NFC
- 5G
- and more...



# Foldable agenda

## **Responsive design**

Media queries

Environment variables

JavaScript

## **Testing**

Browser developer tools (Edge and Chrome)

## **Progressive Web Apps (PWA)**

Starter kit

## **Docs & samples**

Design Kit guidance

Dual-screen web docs

Other developer platforms

# Building apps for foldable/dual-screen devices

## Surface Duo overview

About Surface Duo  
Surface Duo Android emulator  
User experience design  
Design Kit (Figma)

## Kotlin & Java

Jetpack Window Manager  
Jetpack Compose  
Hinge sensor  
Dual-screen controls library  
Samples

## Flutter

Get started  
MediaQuery  
TwoPane Widget  
Samples

## React Native

Get started  
DualScreenInfo  
TwoPaneView  
Samples

## Xamarin

Get started  
DualScreenInfo  
TwoPaneView  
Jetpack Window Manager for  
Xamarin  
Samples

## Unity game development

Get started  
Jetpack Window Manager for  
Unity  
Microsoft dual-screen SDK  
Samples

## Web

Overview  
CSS  
JavaScript  
Emulator and device testing  
Desktop developer tools  
Samples

## Windows

Overview  
TwoPaneView

<https://docs.microsoft.com/dual-screen>

# Responsive design



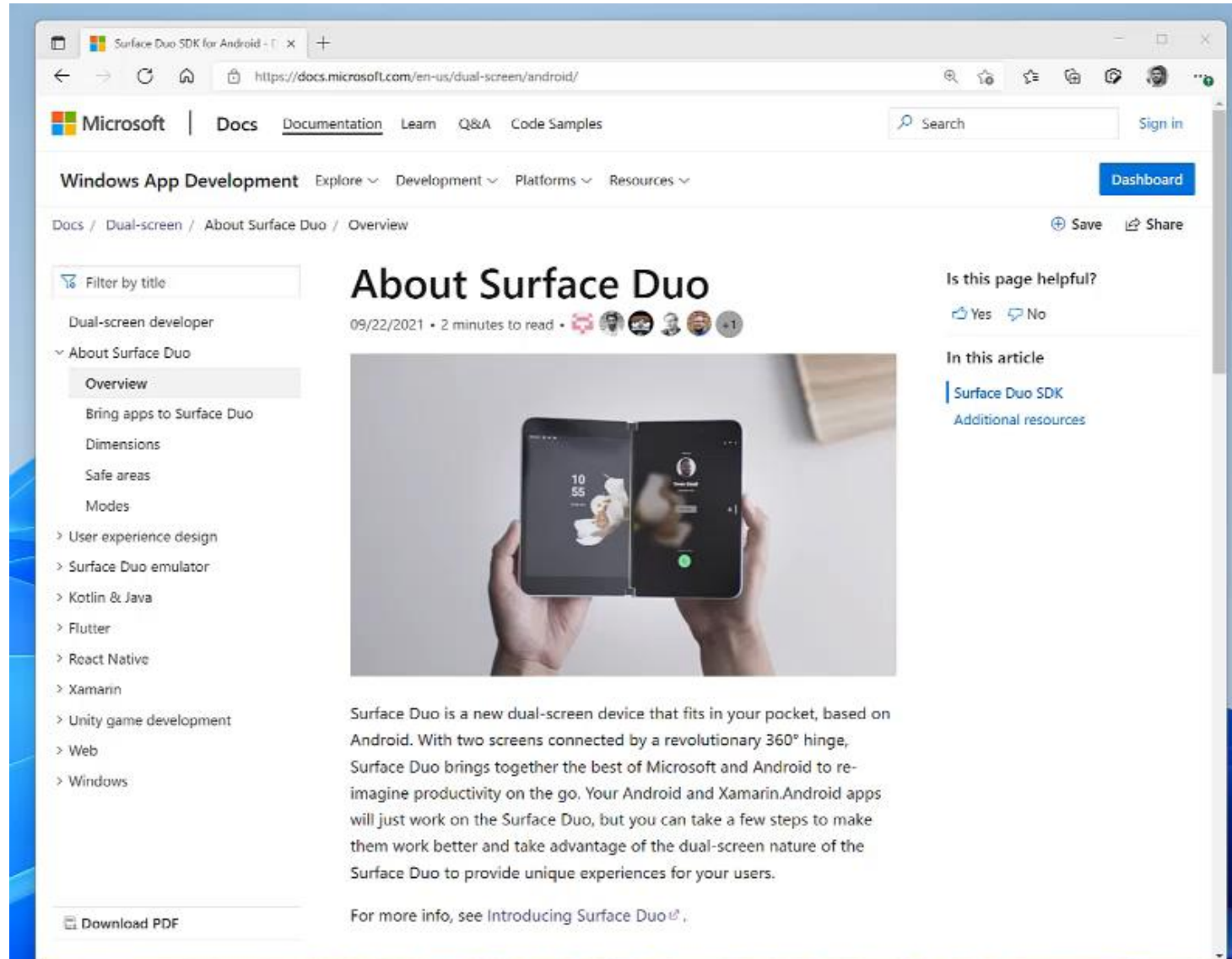
# Responsive design



# Responsive design



# Responsive design



The screenshot shows a web browser displaying the Microsoft Docs page for 'About Surface Duo'. The page is designed to be responsive, with a navigation sidebar on the left and a main content area. The sidebar contains a search bar, a filter by title dropdown, and a list of navigation links including 'Dual-screen developer', 'About Surface Duo', 'Overview', 'Bring apps to Surface Duo', 'Dimensions', 'Safe areas', 'Modes', 'User experience design', 'Surface Duo emulator', 'Kotlin & Java', 'Flutter', 'React Native', 'Xamarin', 'Unity game development', 'Web', and 'Windows'. The main content area features the title 'About Surface Duo', a date '09/22/2021', a reading time '2 minutes to read', and a list of social media sharing icons. Below the title is a large image of a person holding a Surface Duo device. The text below the image describes the device as a new dual-screen device based on Android, designed for productivity on the go. The page also includes a 'Download PDF' button at the bottom left and a 'Is this page helpful?' feedback section on the right.

Surface Duo SDK for Android - [X]

https://docs.microsoft.com/en-us/dual-screen/android/

Microsoft | Docs Documentation Learn Q&A Code Samples

Search Sign in

Windows App Development Explore Development Platforms Resources Dashboard

Docs / Dual-screen / About Surface Duo / Overview Save Share

Filter by title

Dual-screen developer

About Surface Duo

Overview

Bring apps to Surface Duo

Dimensions

Safe areas

Modes

> User experience design

> Surface Duo emulator

> Kotlin & Java

> Flutter

> React Native

> Xamarin

> Unity game development


> Web

> Windows

Download PDF

## About Surface Duo

09/22/2021 • 2 minutes to read



Surface Duo is a new dual-screen device that fits in your pocket, based on Android. With two screens connected by a revolutionary 360° hinge, Surface Duo brings together the best of Microsoft and Android to re-imagine productivity on the go. Your Android and Xamarin.Android apps will just work on the Surface Duo, but you can take a few steps to make them work better and take advantage of the dual-screen nature of the Surface Duo to provide unique experiences for your users.

For more info, see [Introducing Surface Duo](#).

Is this page helpful?

Yes No

In this article

[Surface Duo SDK](#)

[Additional resources](#)

# A new class of devices...



# How can we handle this?



# Media queries in CSS

```
@media print {  
    // styles applied only when printing/previewing  
    // set appropriate layouts and colors  
    // (i.e. no background color)  
}  
  
@media screen and (max-width: 600px) {  
    // narrow screens only (like mobile)  
    // hide or shrink elements  
}
```

```
<link rel="stylesheet" media="print" href="example.css" >
```

# Media queries – print

```
@media print {  
  // styles applied only when printing/previewing  
  // set appropriate layouts and colors  
  
  // e.g. no background color  
  html {background-color: white;}  
  
  // e.g. hide menus or nav  
  .nav {display:none;}  
}
```

# Media queries – print

Surface Duo SDK for Android

https://docs.microsoft.com/en-us/dual-screen/android/

Microsoft | Docs Documentation Learn Q&A Code Samples

Search

Sign in

Windows App Development

Explore Development Platforms Resources

Dashboard

Docs / Dual-screen / About Surface Duo / Overview

Save Share

Filter by title

Dual-screen developer

About Surface Duo

Overview

Bring apps to Surface Duo

Dimensions

Safe areas

Modes

User experience design

Surface Duo emulator

Kotlin & Java

Flutter

React Native

Xamarin







Unity game development


Web

Windows

Download PDF

## About Surface Duo



09/22/2021 • 2 minutes to read •       +1



Surface Duo is a new dual-screen device that fits in your pocket, based on Android. With two screens connected by a revolutionary 360° hinge, Surface Duo brings together the best of Microsoft and Android to re-imagine productivity on the go. Your Android and Xamarin.Android apps will just work on the Surface Duo, but you can take a few steps to make them work better and take advantage of the dual-screen nature of the Surface Duo to provide unique experiences for your users.

For more info, see [Introducing Surface Duo](#).

Is this page helpful?

 Yes  No

In this article

[Surface Duo SDK](#)

[Additional resources](#)

# Media queries - print

Surface Duo SDK for Android - [x]

https://docs.microsoft.com/en-us/dual-screen/android/

Microsoft | Docs Documentation Learn Q&A Code Samples Sign in

Windows App Development Explore Development Platforms Resources Dashboard

Docs / Dual-screen / About Surface Duo / Overview Save Share


Filter by title

- Dual-screen developer
- About Surface Duo
  - Overview
  - Bring apps to Surface Duo
  - Dimensions
  - Safe areas
  - Modes
  - User experience design
  - Surface Duo emulator
  - Kotlin & Java
  - Flutter
  - React Native
  - Xamarin
  - Unity game development
  - Web
  - Windows

Download PDF

## About Surface Duo

09/22/2021 • 2 minutes to read • [User Avatars]



Is this page helpful?  
Yes No

In this article

- Surface Duo SDK
- Additional resources

Surface Duo is a new dual-screen device that fits in your pocket, based on Android. With two screens connected by a revolutionary 360° hinge, Surface Duo brings together the best of Microsoft and Android to re-imagine productivity on the go. Your Android and Xamarin.Android apps will just work on the Surface Duo, but you can take a few steps to make them work better and take advantage of the dual-screen nature of the Surface Duo to provide unique experiences for your users.

For more info, see [Introducing Surface Duo](#).

Surface Duo SDK for Android - [x]

https://docs.microsoft.com/en-us/dual-screen/android/

Microsoft | Docs Documentation Learn Q&A Code Samples Sign in

Windows App Development Explore Development Platforms Resources Dashboard

Docs / Dual-screen / About Surface Duo / Overview Save Share

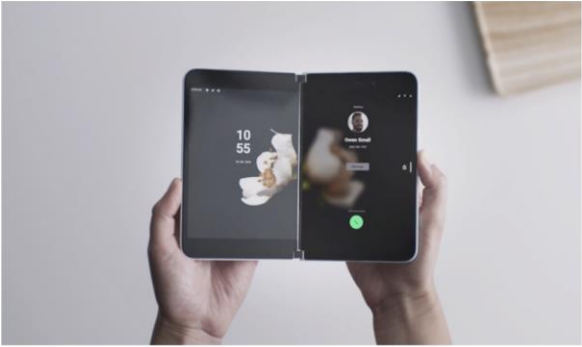
Filter by title

- Dual-screen developer
- About Surface Duo
  - Overview
  - Bring apps to Surface Duo
  - Dimensions
  - Safe areas
  - Modes
  - User experience design
  - Surface Duo emulator
  - Kotlin & Java
  - Flutter
  - React Native
  - Xamarin
  - Unity game development
  - Web
  - Windows

Download PDF

## About Surface Duo

09/22/2021 • 2 minutes to read • [User Avatars]



Is this page helpful?  
Yes No

In this article

- Surface Duo SDK
- Additional resources

Surface Duo is a new dual-screen device that fits in your pocket, based on Android. With two screens connected by a revolutionary 360° hinge, Surface Duo brings together the best of Microsoft and Android to re-imagine productivity on the go. Your Android and Xamarin.Android apps will just work on the Surface Duo, but you can take a few steps to make them work better and take advantage of the dual-screen nature of the Surface Duo to provide unique experiences for your users.

For more info, see [Introducing Surface Duo](#).

Print

Total: 3 pages

Printer: Save as PDF

Layout: Portrait Landscape

Pages: All e.g. 1-5, 8, 11-13

More settings Troubleshoot printer issues

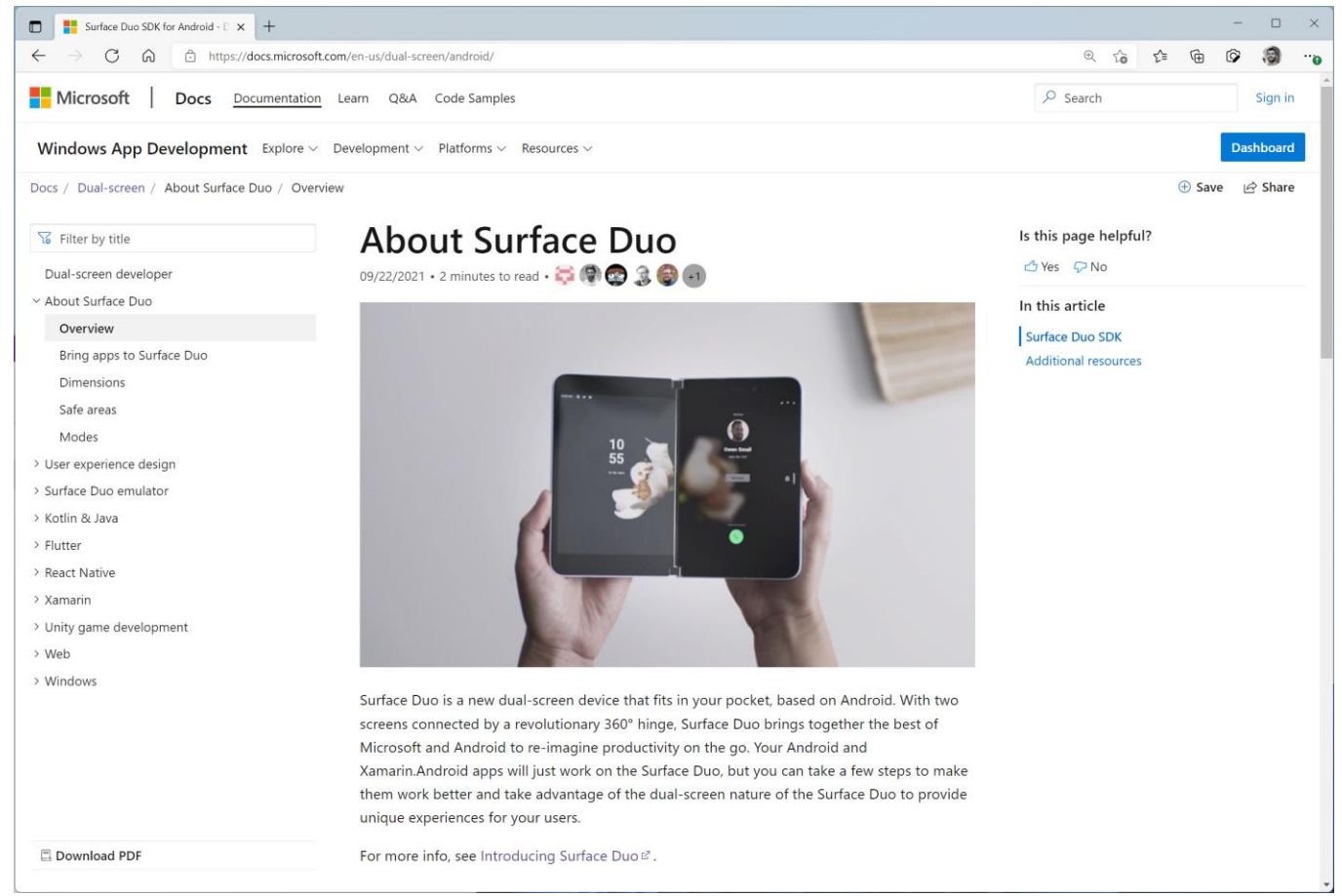
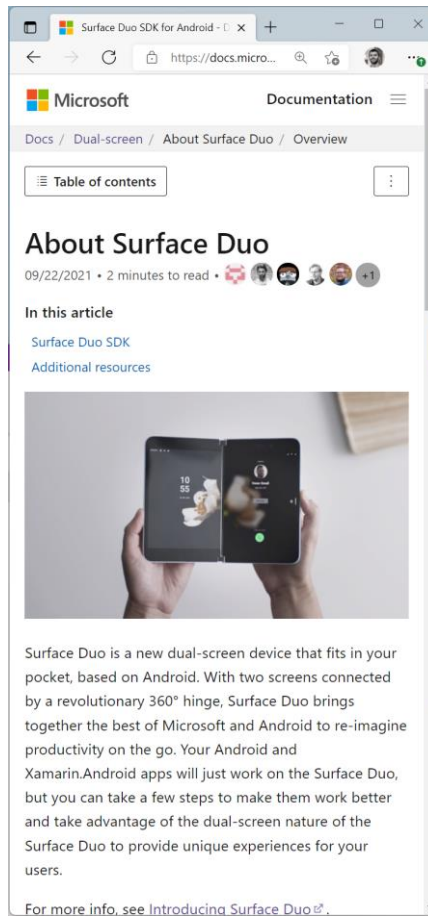
Save Cancel

Download PDF

# Media queries – screen width

```
@media screen and (max-width: 600px) {  
    // narrow screens only (like mobile)  
}  
  
@media screen and (min-width: 768px) {  
    // wide screens (like maximized desktop)  
}  
  
@media (min-width: 768px) {  
    // 'screen' is optional  
    // wide screens (like maximized desktop)  
}
```

# Media queries - width



# We need a new @media standard!

**World Wide Web Consortium (W3C)**

<https://www.w3.org/>

**Proposal process**

<https://github.com/w3c/csswg-drafts/issues/4736>

**Adoption...**



Microsoft Edge



Google Chrome

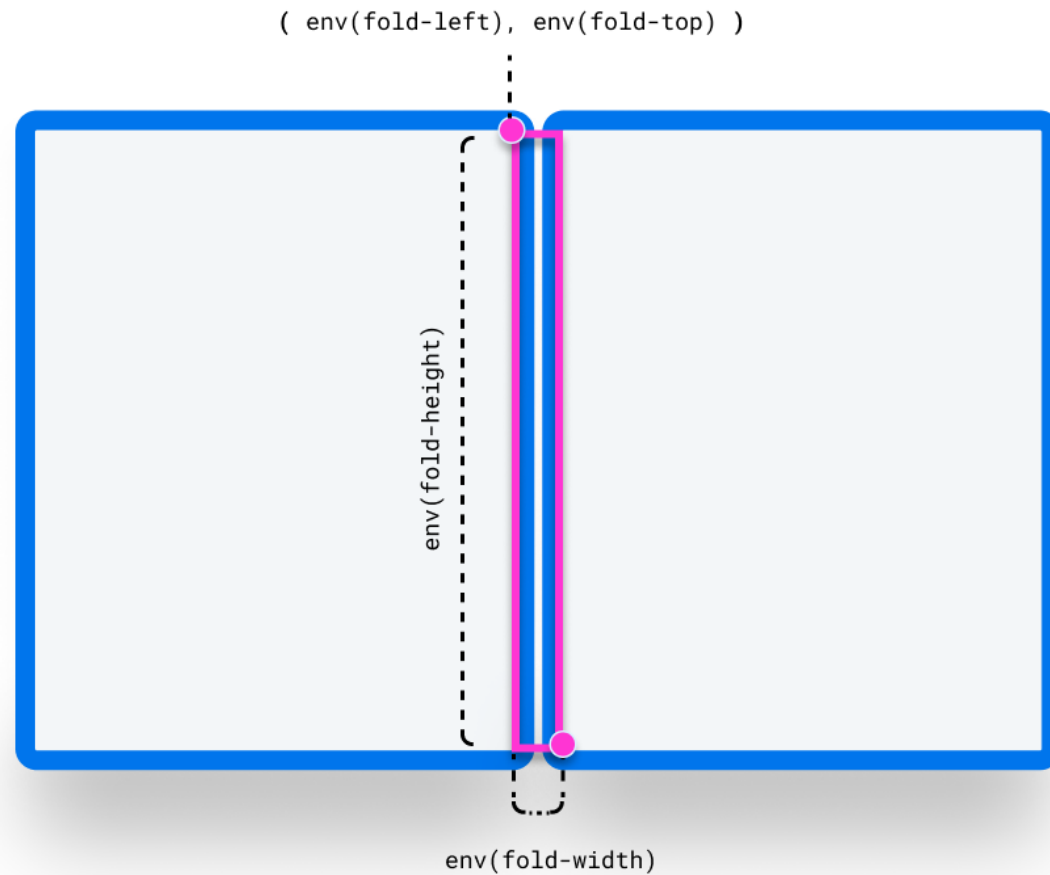
# New media query

```
@media (screen-spanning: single-fold-vertical) {  
  // styles applied in double-portrait (wide) mode  
}  
  
@media (screen-spanning: single-fold-horizontal) {  
  // styles applied in double-landscape (tall) mode  
}
```



# New environment variables

```
env(fold-left)  
env(fold-top)  
env(fold-width)  
env(fold-height)
```

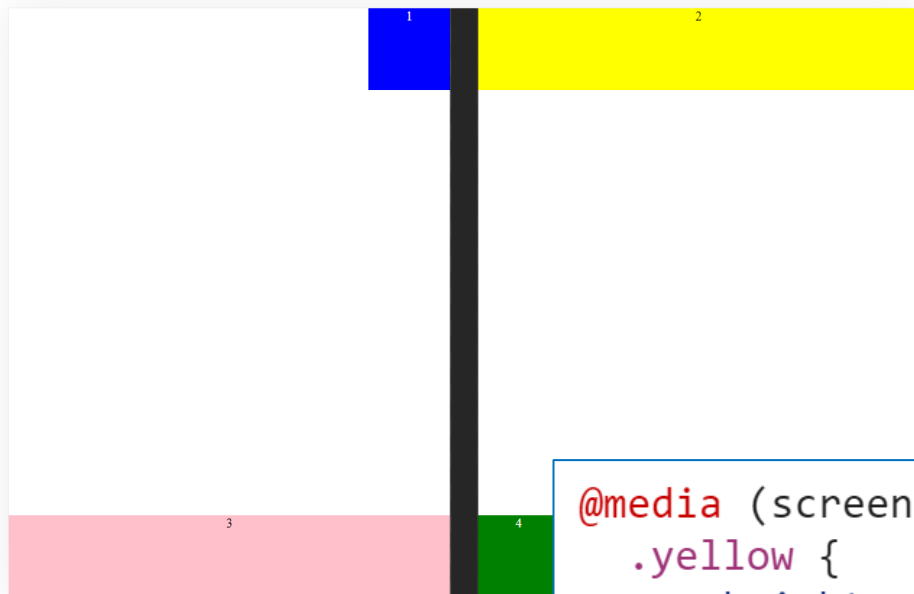


# New JavaScript

```
const screenSegments = window.getWindowSegments();

if(screenSegments.length > 1 ) {
    // now we know the device is a foldable
    // and we can update CSS classes in our layout
    document.body.classList.add('is-foldable');
    // other changes as required for layout
}
```

# Example



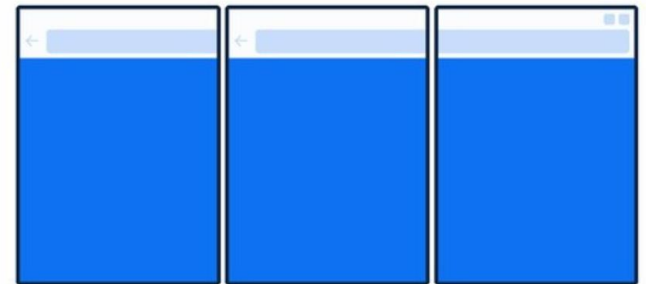
```
@media (screen-spanning: single-fold-vertical) {  
  .yellow {  
    height: 100px;  
    width: calc(100vw - env(fold-left) - env(fold-width)); /*fold-right*/  
    position: absolute;  
    left: calc(env(fold-left) + env(fold-width)); /*fold-right*/  
    top: 0;  
    background-color: yellow;  
    text-align: center;  
  }  
}
```

...but what about this?



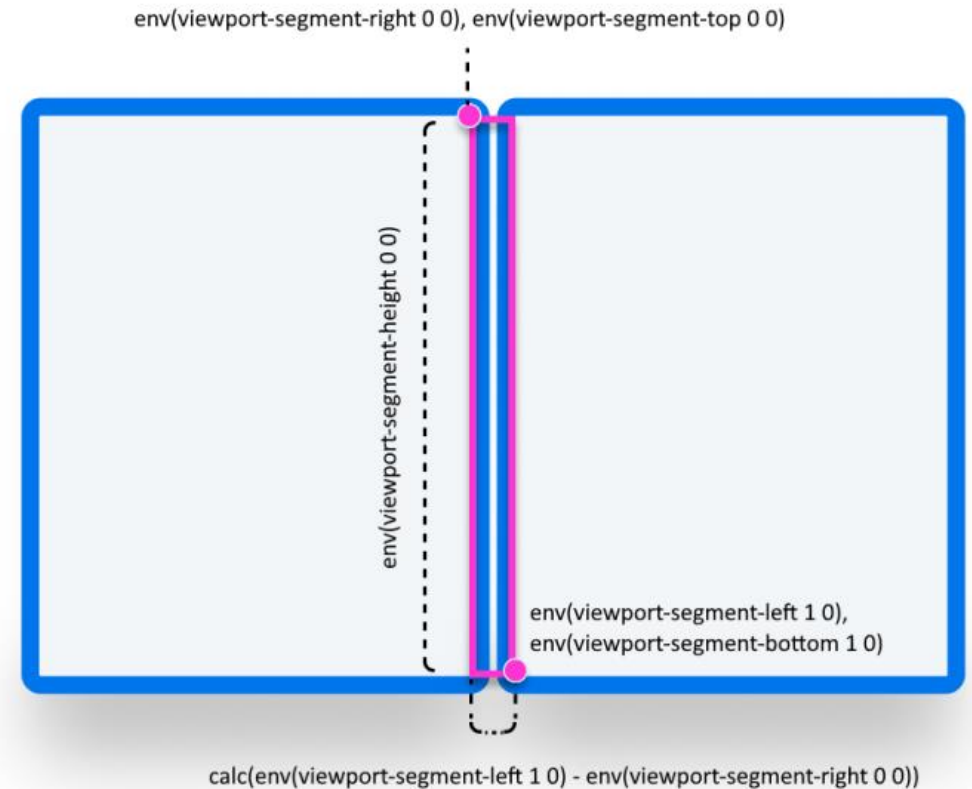
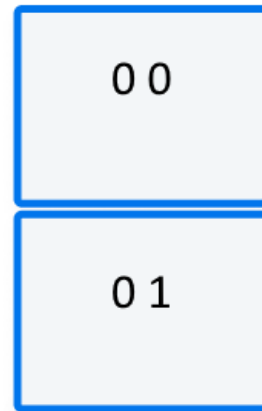
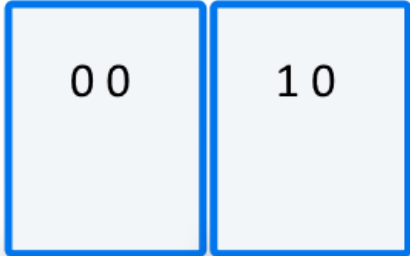
# Updated media query

```
@media (horizontal-viewport-segments:2) {  
  // styles applied in double-portrait (wide) mode  
}  
@media (vertical-viewport-segments:2) {  
  // styles applied in double-landscape (tall) mode  
}  
@media (horizontal-viewport-segments:3) {  
  // styles applied in triple-portrait (really wide) mode  
}
```



# Updated environment variables

```
env(viewport-segment-width 0 0)  
env(viewport-segment-height 0 0)  
env(viewport-segment-right 0 0)  
env(viewport-segment-left 1 0)
```



# Updated JavaScript

```
const screenSegments = window.visualViewport.segments;  
  
if(screenSegments) {  
  if(screenSegments.length > 1 ) {  
    // now we know there are multiple segments  
    // and we can update CSS classes in our layout  
    document.body.classList.add('is-foldable');  
    // other changes as required for layout  
  }  
}
```

# Testing

Enable via flags

*edge://flags*

*chrome://flags*

The screenshot shows a web browser interface in emulation mode. The top toolbar includes a dropdown menu set to 'Surface Duo', resolution '1114 x 720', zoom '50%', and a 'No' button. The browser window displays a web page with four colored rectangular divs: a blue div (labeled '1') on the top left, a yellow div (labeled '2') on the top right, a pink div (labeled '3') on the bottom left, and a green div (labeled '4') on the bottom right. A thick black vertical line separates the left and right halves of the screen. To the right of the browser window, the 'Elements' panel shows the HTML structure, including a 'content' div containing the four colored divs. Below the browser window, a code editor displays the following CSS code:

```
@media (screen-spanning: single-fold-vertical) {  
  .yellow {  
    height: 100px;  
    width: calc(100vw - env(fold-left) - env(fold-width)); /*fold-right*/  
    position: absolute;  
    left: calc(env(fold-left) + env(fold-width)); /*fold-right*/  
    top: 0;  
    background-color: yellow;  
    text-align: center;  
  }  
}
```

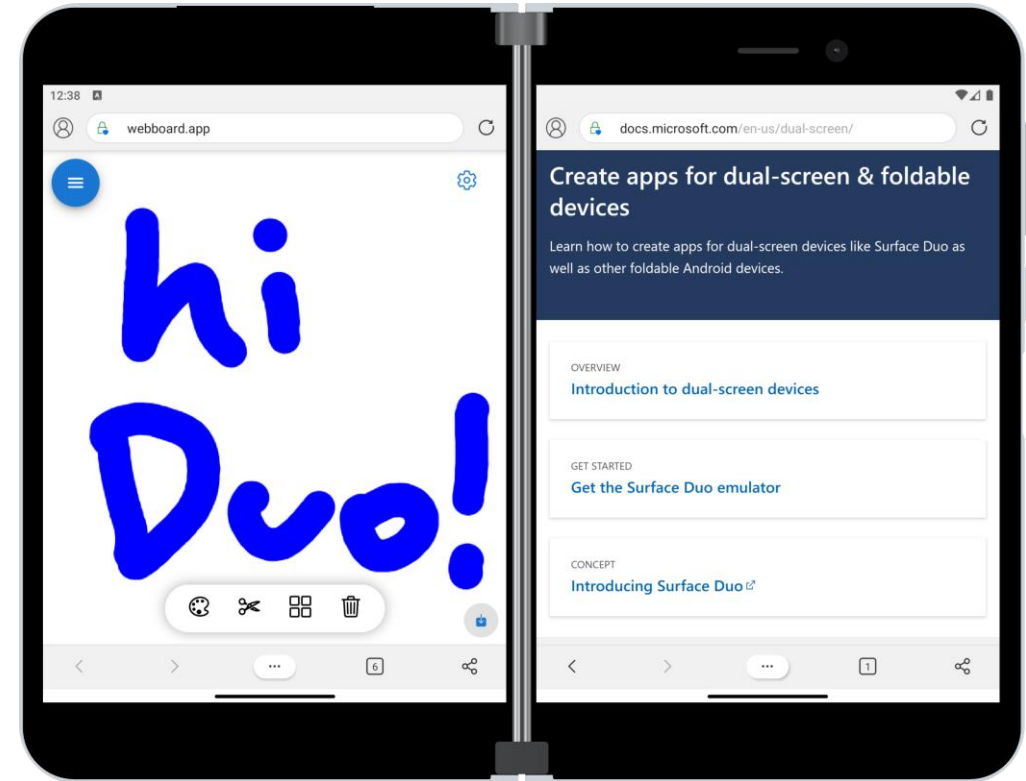
<https://docs.microsoft.com/dual-screen/web/desktop-developer-tools>

# WEB DEMO

<https://aka.ms/cs50-demo>

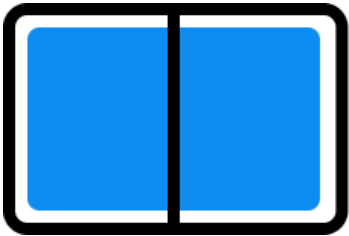
# Samples

- Webboard
- SimpleEdit photos
- Photo gallery
- Battleship game
- FoldNews reader
- StarFold game



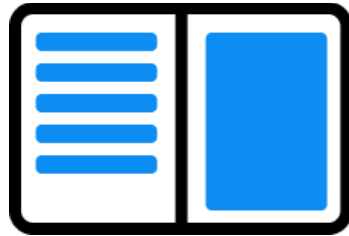
<https://aka.ms/cs50-demo>

# Design patterns



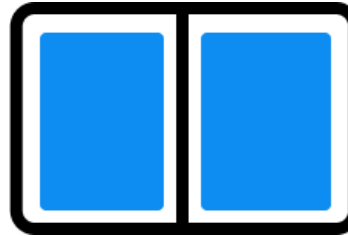
## Extended Canvas

Expand content across the seam



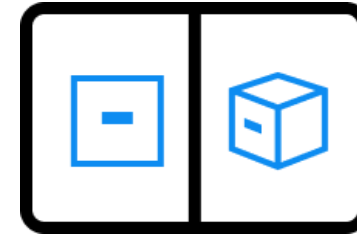
## List-Detail

Drill down one layer deeper into content



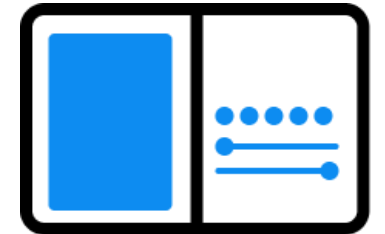
## Two Page

Document-oriented, made for reading



## Dual View

Alternate or transient view of the same info



## Companion Pane

Supporting content or smart suggestions

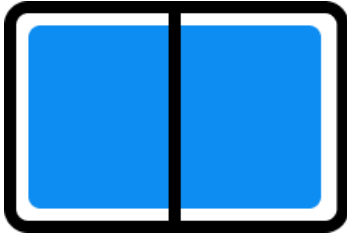


v 2.1

<https://aka.ms/SurfaceDuoDesign>

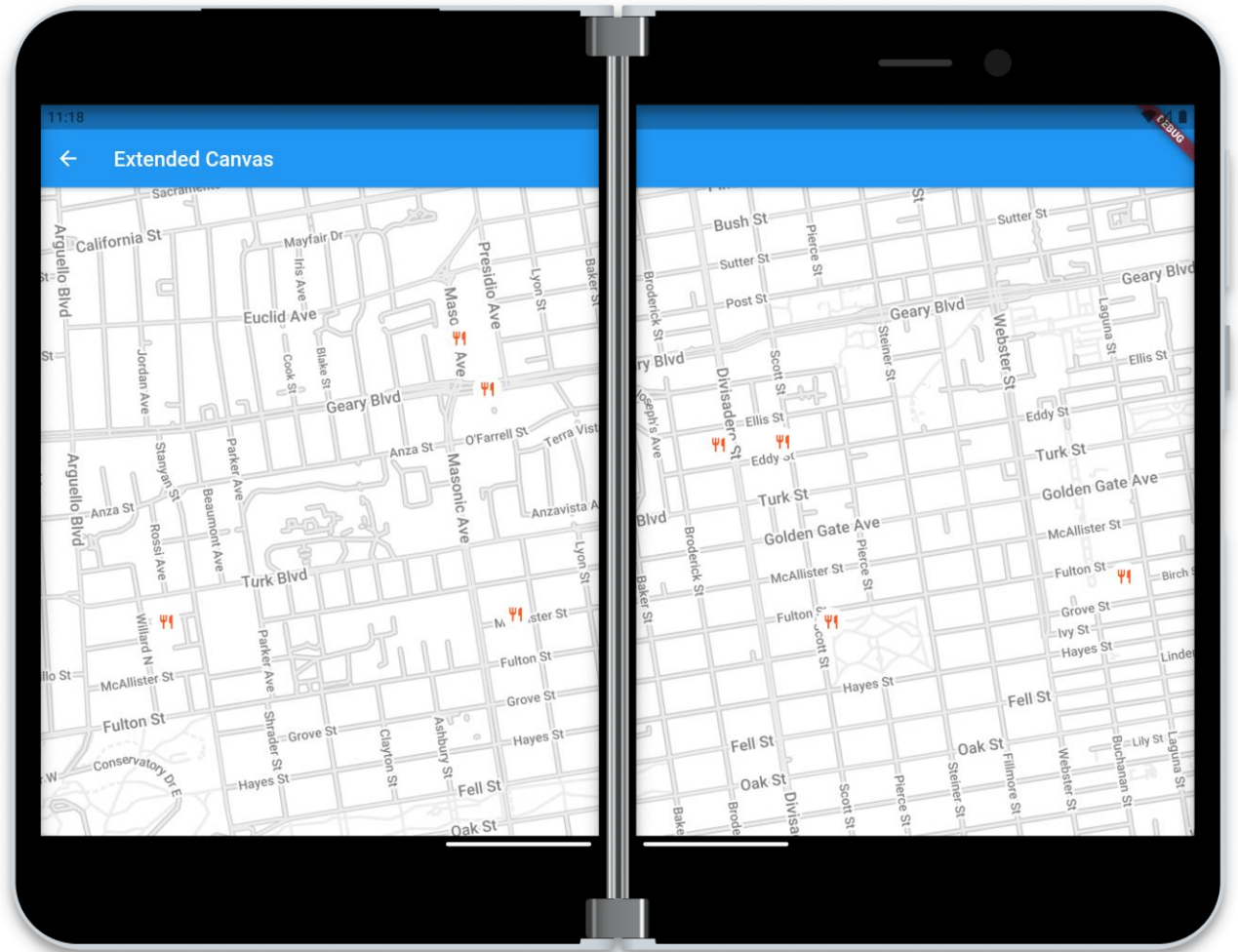
# Surface Duo Design Kit

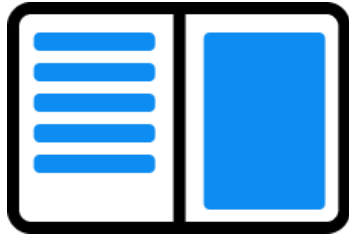




## Extended Canvas

Expand content  
across the seam

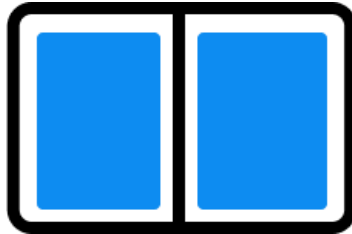




## List-Detail

Drill down one layer deeper into content

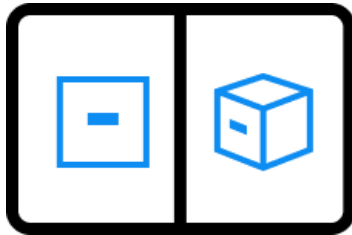




## Two Page

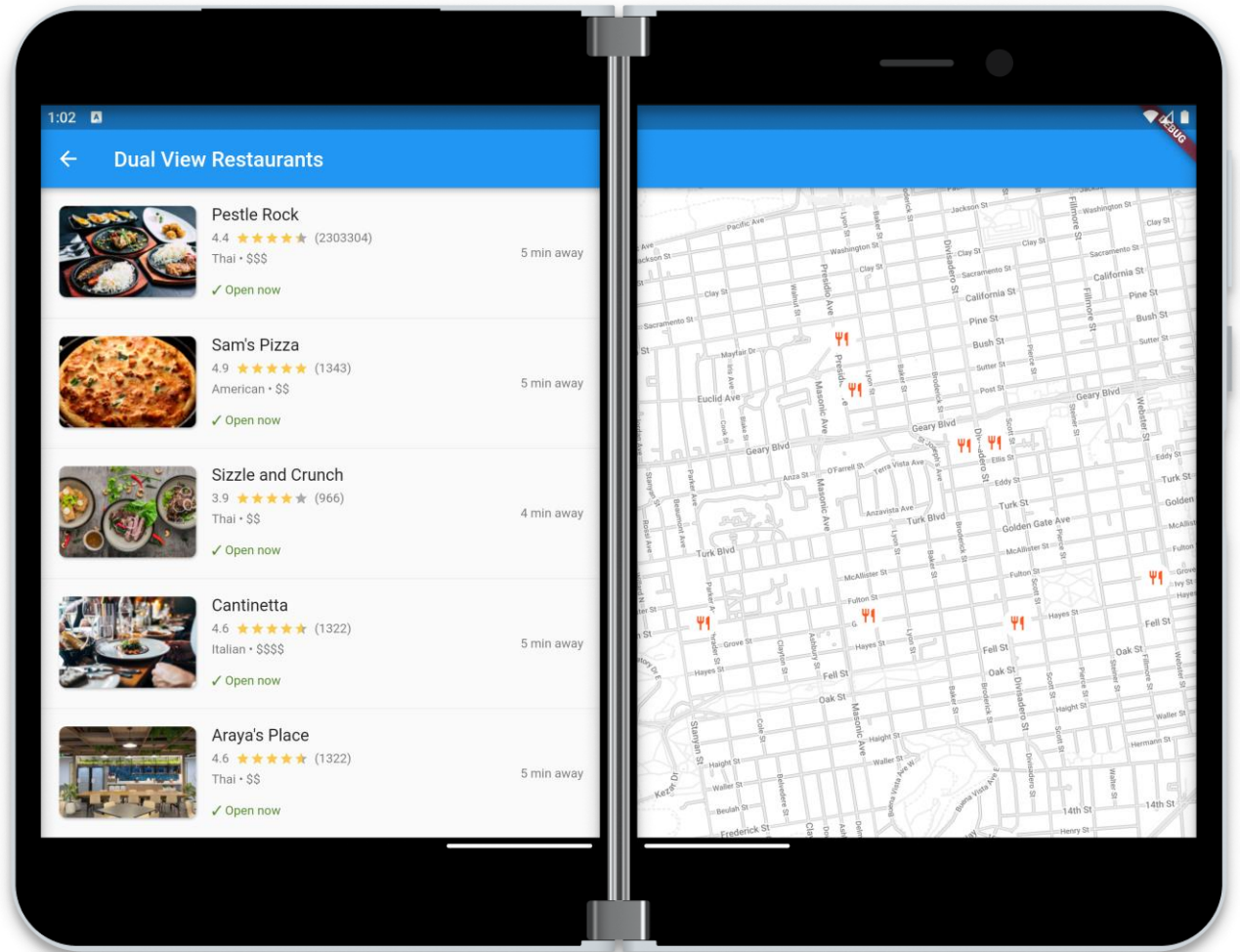
Document-oriented,  
made for reading

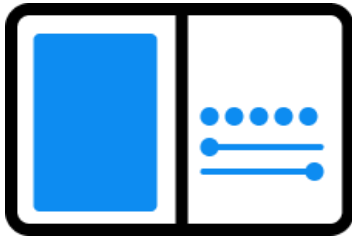




## Dual View

Alternate or transient view of the same info



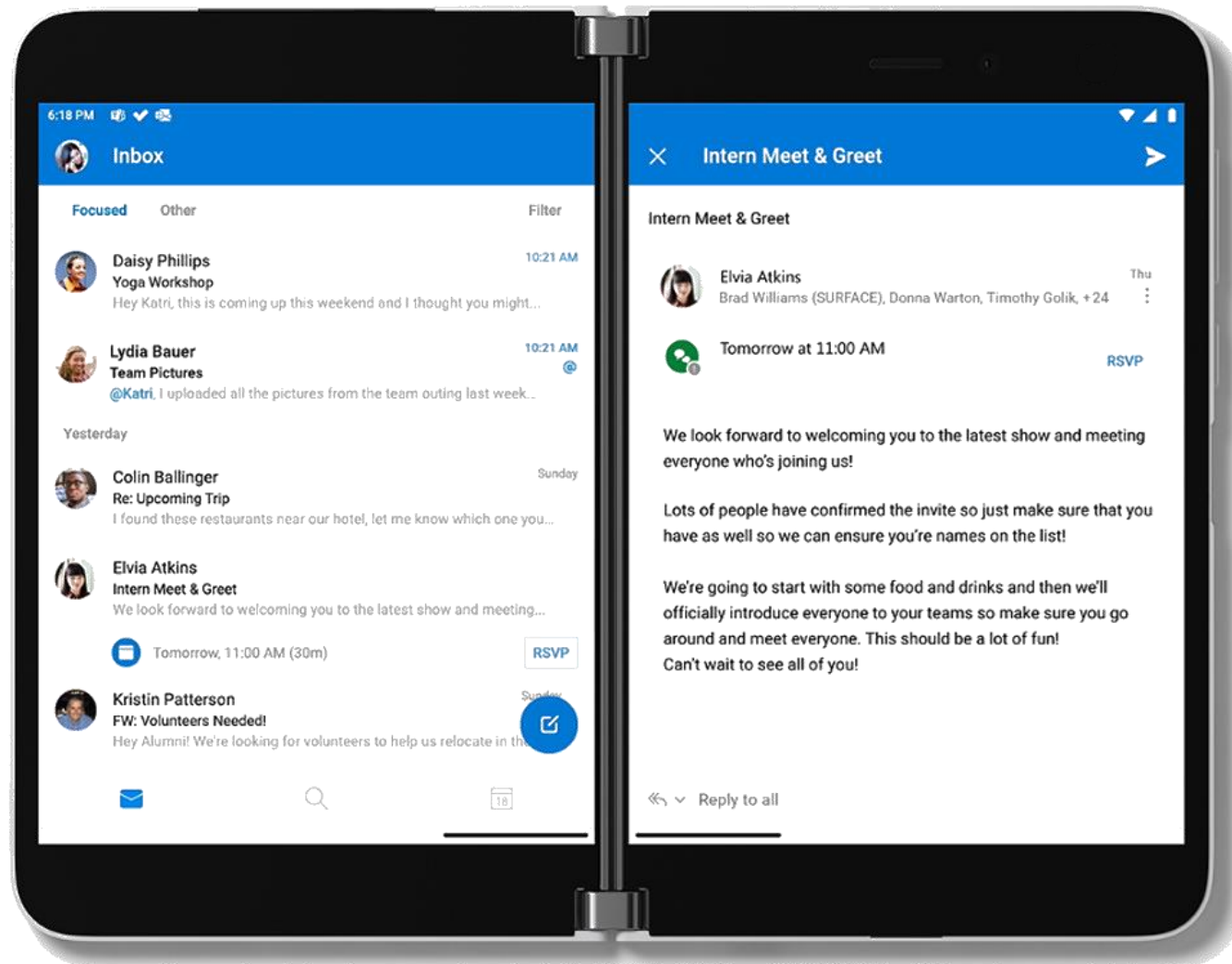


## Companion Pane

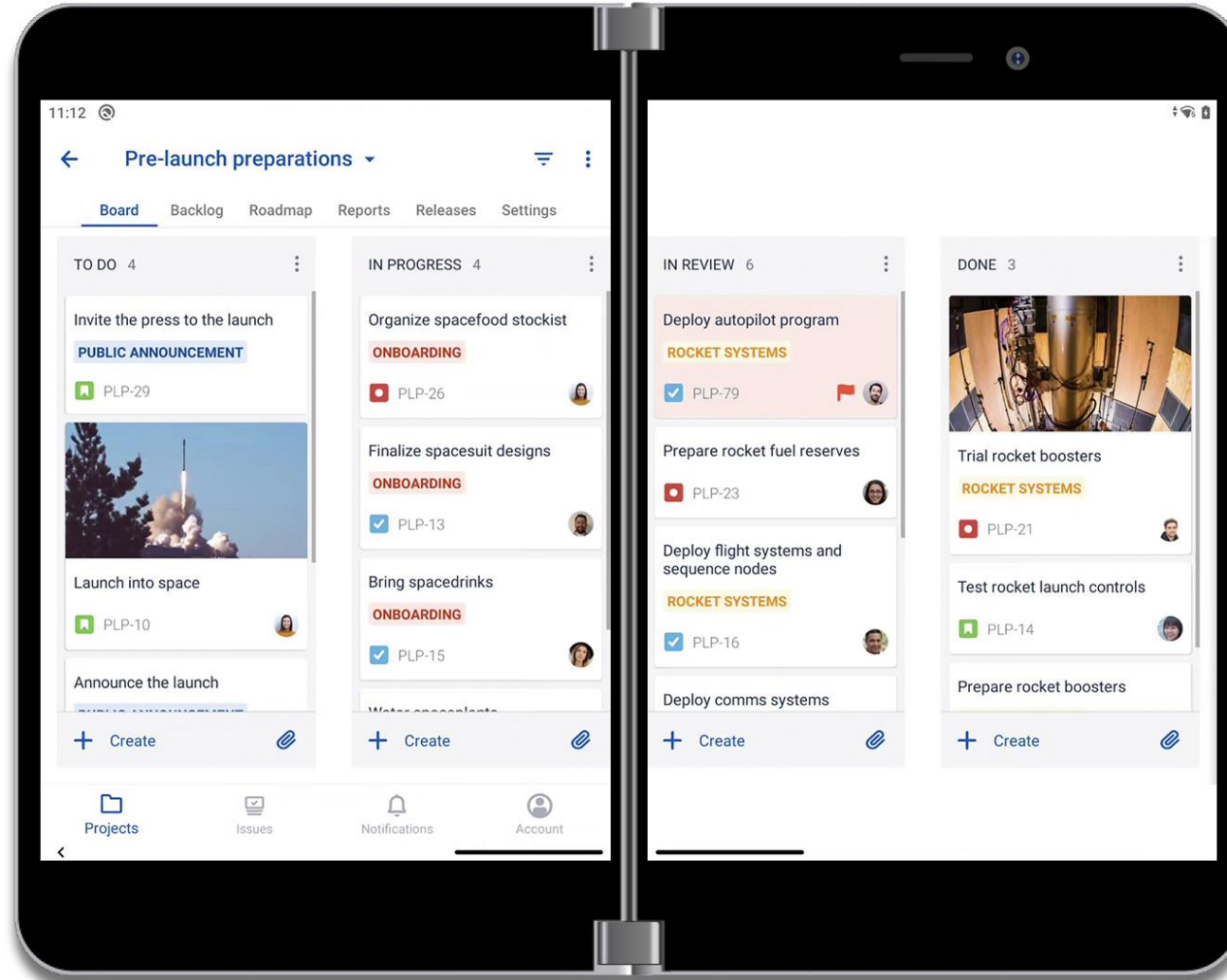
Supporting content or  
smart suggestions



# Outlook



# Atlassian Jira Cloud



# Other developer platforms

## Surface Duo overview

About Surface Duo  
Surface Duo Android emulator  
User experience design  
Design Kit (Figma)

## Kotlin & Java

Jetpack Window Manager  
Jetpack Compose  
Hinge sensor  
Dual-screen controls library  
Samples

## Flutter

Get started  
MediaQuery  
TwoPane Widget  
Samples

## React Native

Get started  
DualScreenInfo  
TwoPaneView  
Samples

## Xamarin

Get started  
DualScreenInfo  
TwoPaneView  
Jetpack Window Manager for  
Xamarin  
Samples

## Unity game development

Get started  
Jetpack Window Manager for  
Unity  
Microsoft dual-screen SDK  
Samples

## Web

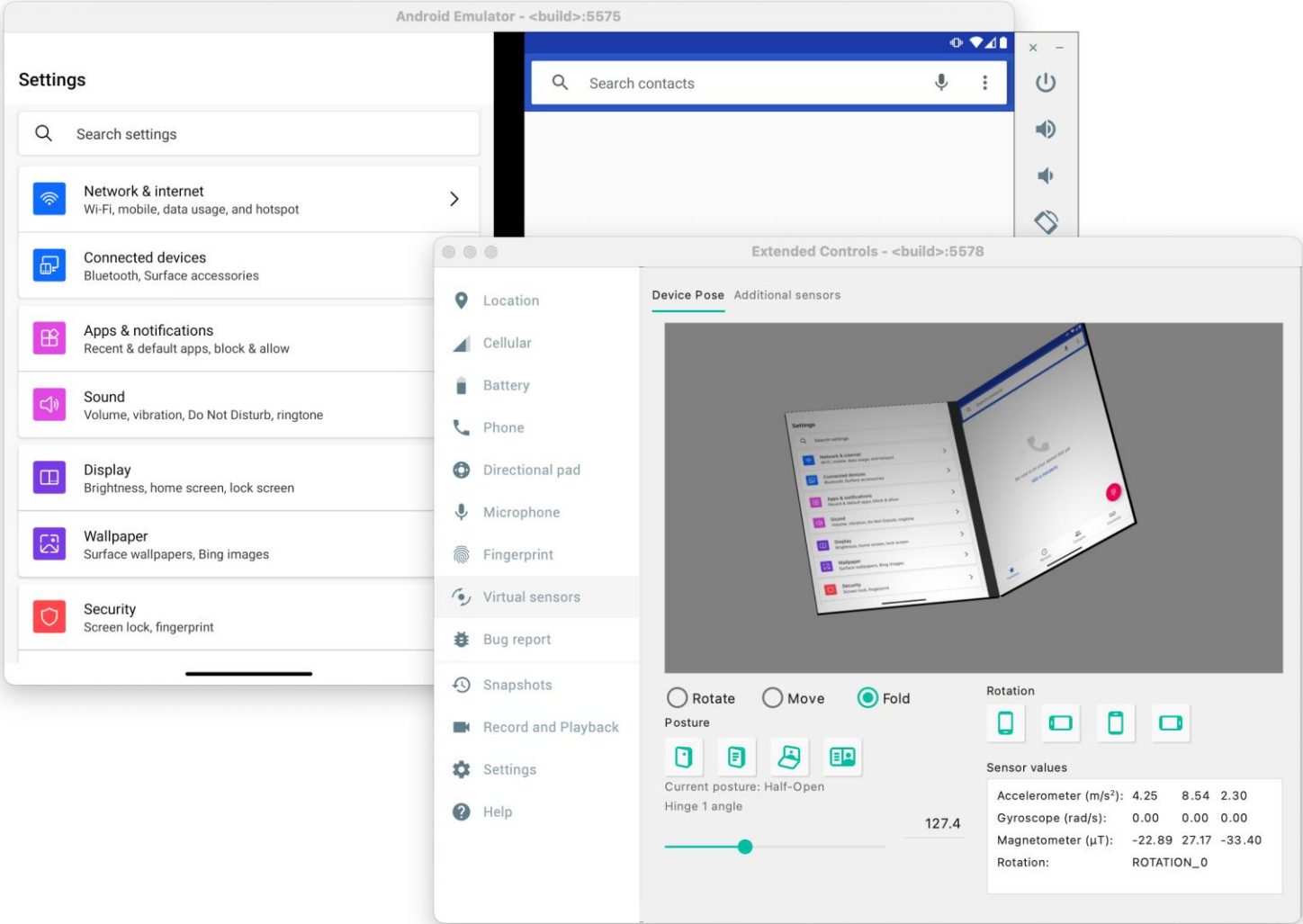
Overview  
CSS  
JavaScript  
Emulator and device testing  
Desktop developer tools  
Samples

## Windows

Overview  
TwoPaneView

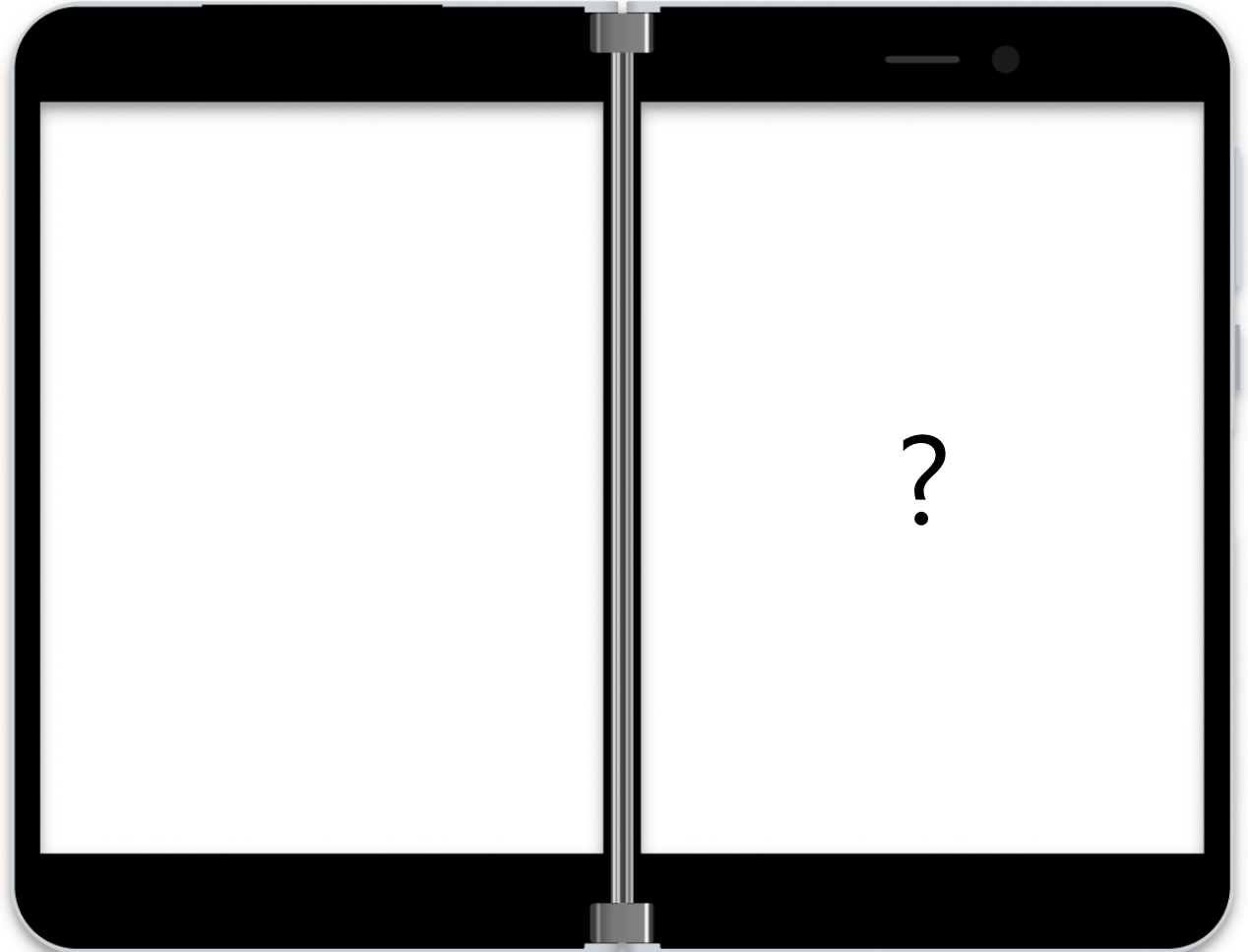
<https://docs.microsoft.com/dual-screen>

# Surface Duo Emulator



# Enhance your final project

What will you do with that extra screen?



# Progressive Web Apps (PWA)

- Build for web (HTML, CSS, JavaScript)
- Deploy to web
- Bundle as downloadable/installable apps
  - Desktop
  - Mobile

[aka.ms/pwa-starter](https://aka.ms/pwa-starter)  
[aka.ms/starter-docs](https://aka.ms/starter-docs)



## **Responsive design**

Media queries

Environment variables

JavaScript

## **Testing**

Browser developer tools (Edge and Chrome)

## **Progressive Web Apps (PWA)**

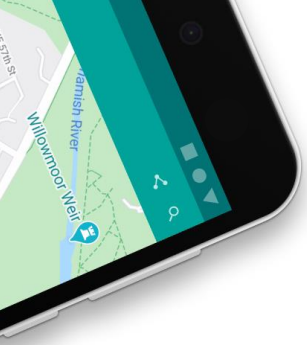
Starter kit

## **Docs & samples**

Design Kit guidance

Dual-screen web docs

Other developer platforms



 [aka.ms/cs50-demo](https://aka.ms/cs50-demo)

 [@surfaceduodev](https://twitter.com/surfaceduodev)

# Thank you.

