

ALL
IN

CISCO *Live!*

DevNet Zone



The bridge to possible

Embedding Webex In Your Business Applications

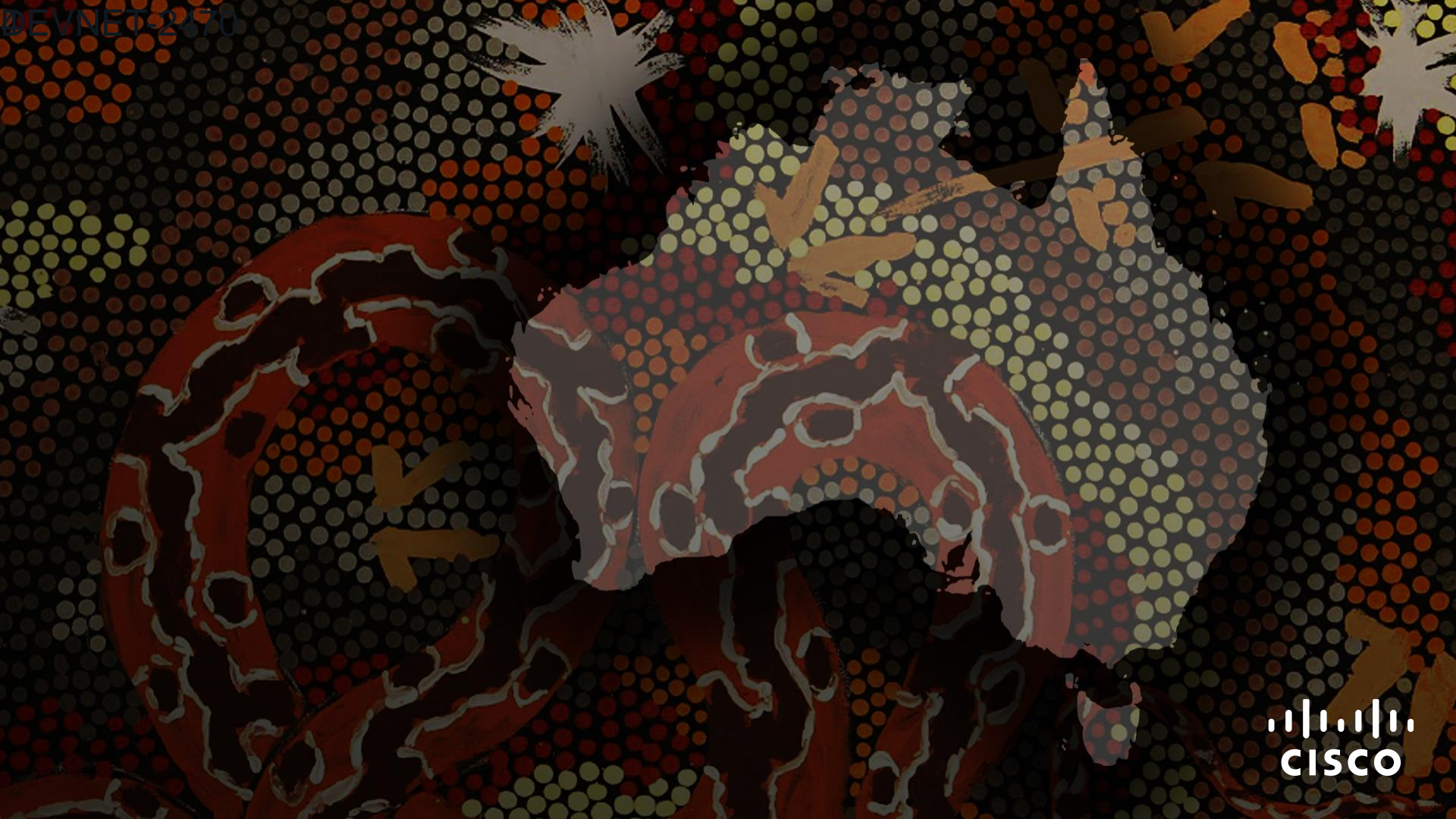
Using Webex Integration, Guest Issuer and SDK

Alvin Lau
Systems Engineer
DEVNET-2470



DevNet Zone

#CiscoLiveAPJC



Cisco Webex App

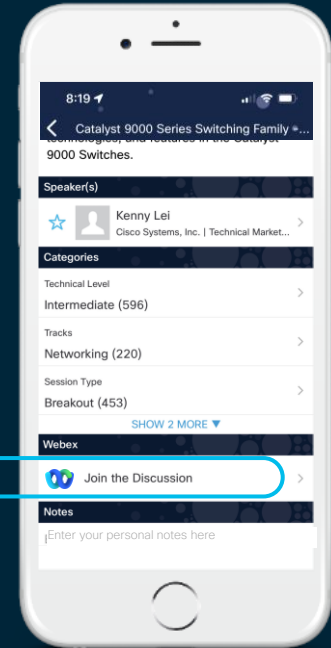
Questions?

Use Cisco Webex App to chat with the speaker after the session

How

- 1 Find this session in the Cisco Live Mobile App
- 2 Click “Join the Discussion”
- 3 Install the Webex App or go directly to the Webex space
- 4 Enter messages/questions in the Webex space

Webex spaces will be moderated by the speaker until Thursday 22 December, 2022.





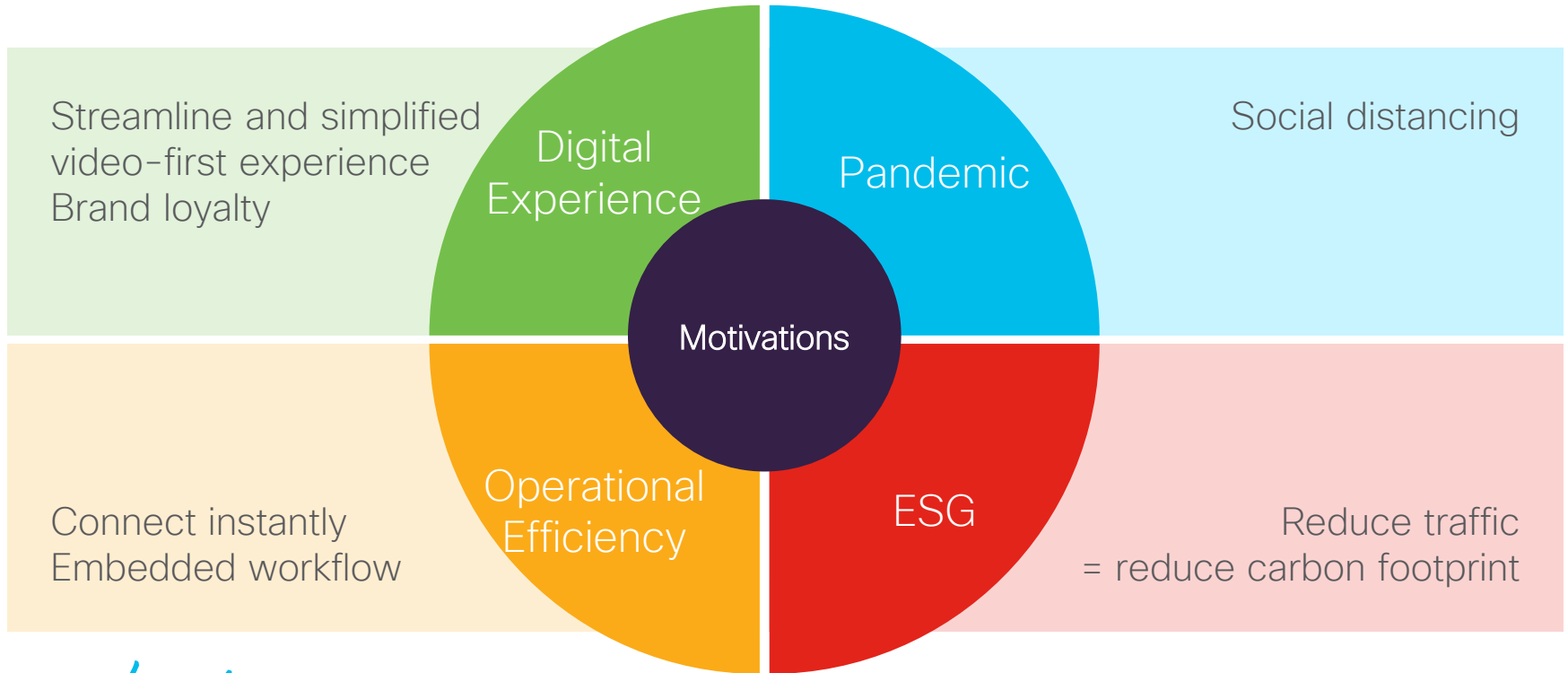
Agenda

- Motivations and User Journey
- Webex for Developers
- Use Case Demo

Motivations and User Journey



Motivations – Video Embedded Applications



Video Embedded Applications

Finance & Insurance

- *Remote loan officer*
- *Financial advisor*
- *Insurance broker*



Healthcare

- *Telehealth consultation*
- *Guest visit*



Government

- *Remote court hearings*
- *Firefighting collaboration*

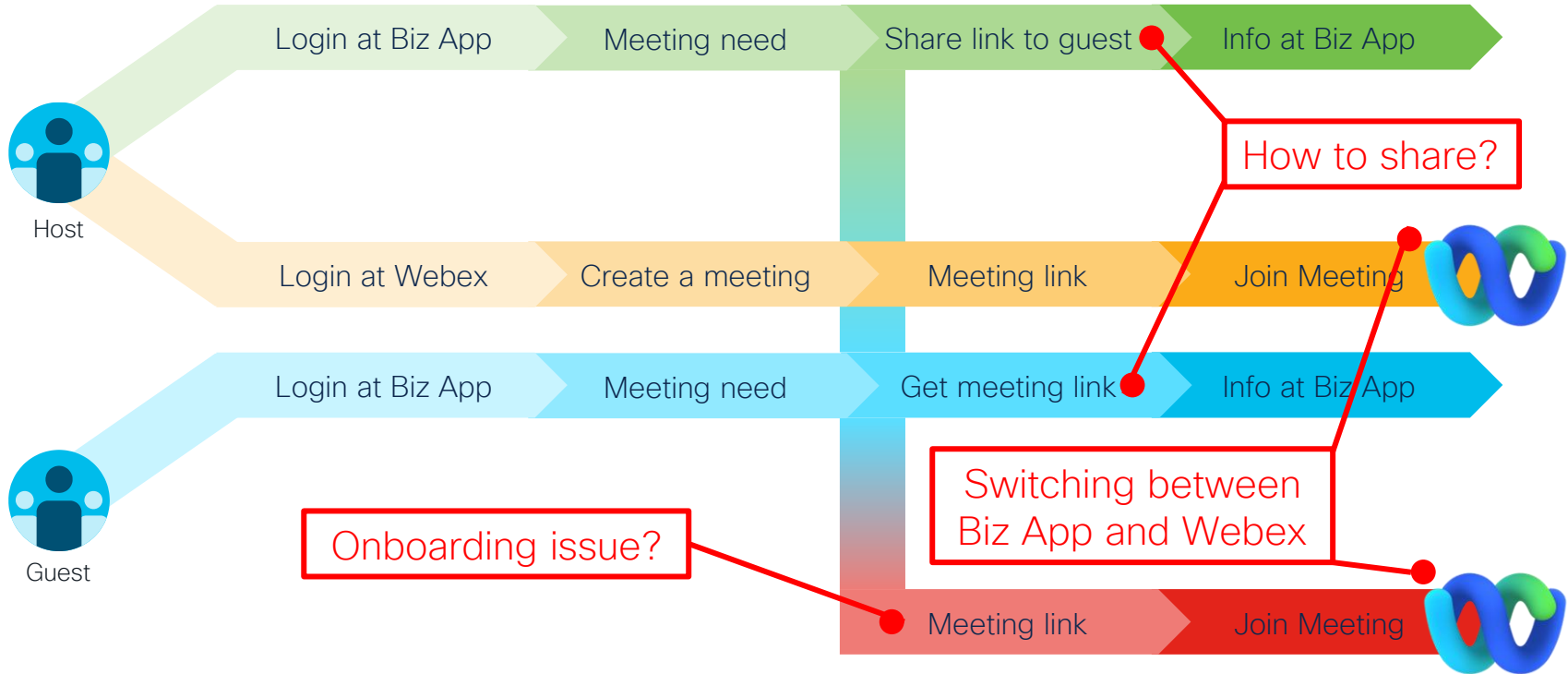


Education

- *Remote classes*



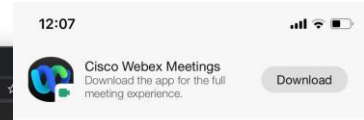
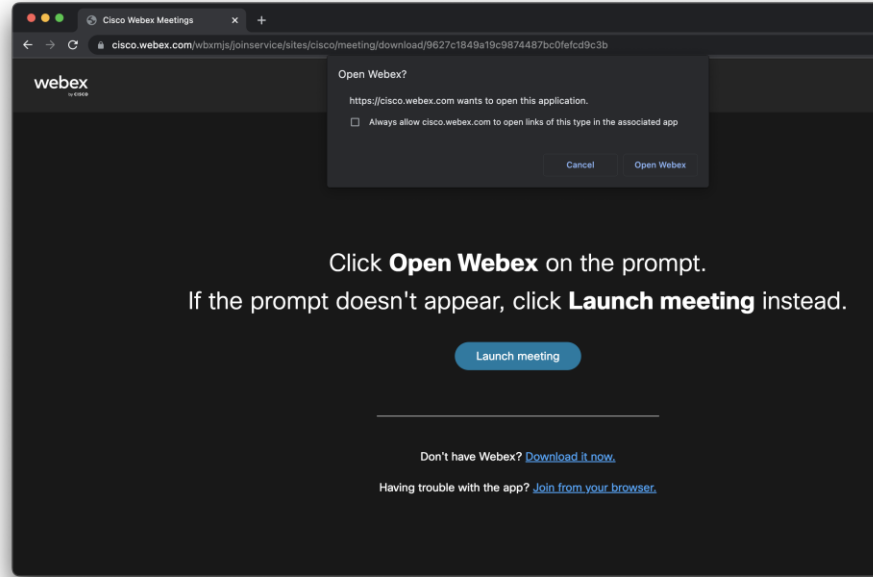
Webex in Business Workflow Today



Webex in Business Workflow Today



Biz App



Alvin Lau's Personal Room

If you don't have Webex Meetings:

[Download Webex Meetings](#)

If you already have Webex Meetings:

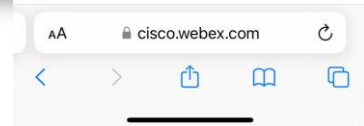
[Join meeting](#)

or

Call-in with audio only

[CALL: 1-](#)

Phone charges may apply.



CISCO *Live!*

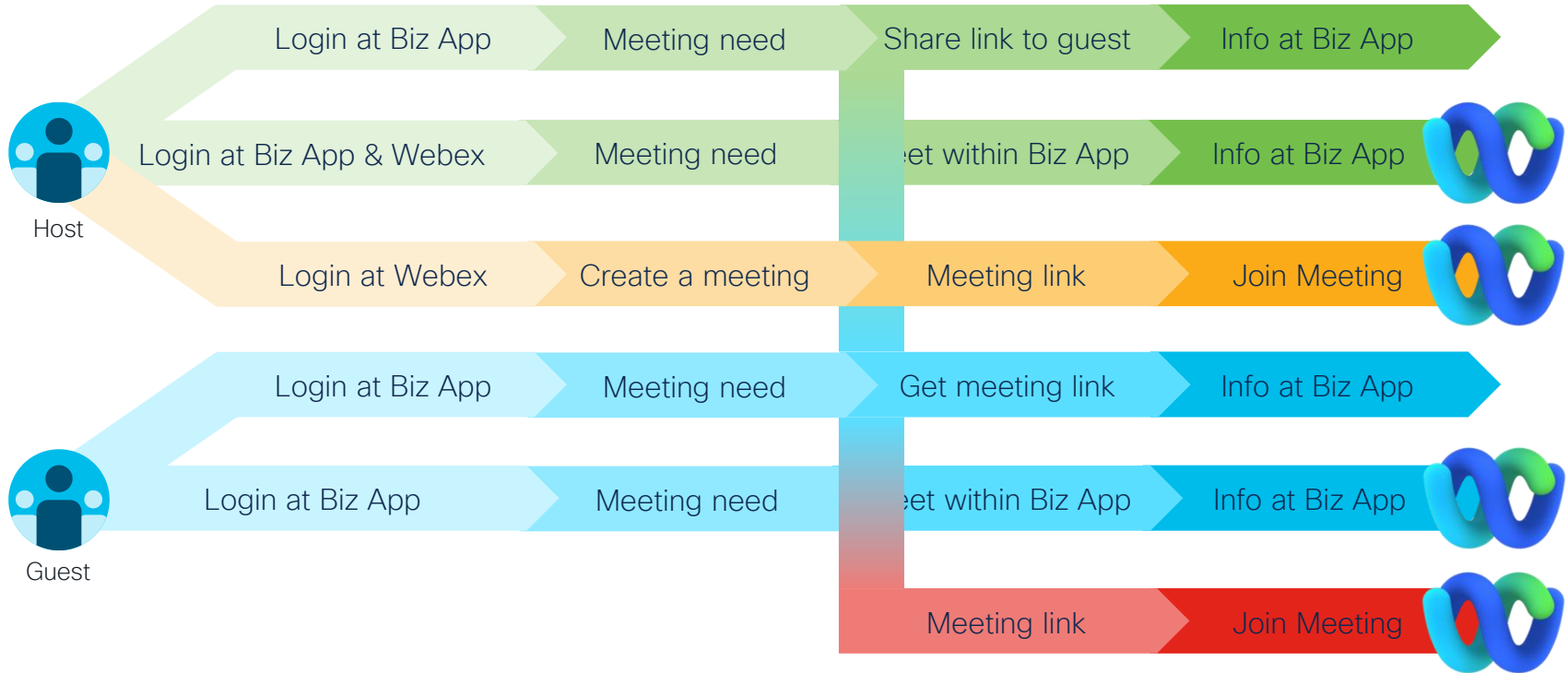
DevNet Zone

#CiscoLiveAPJC DEVNET-2470

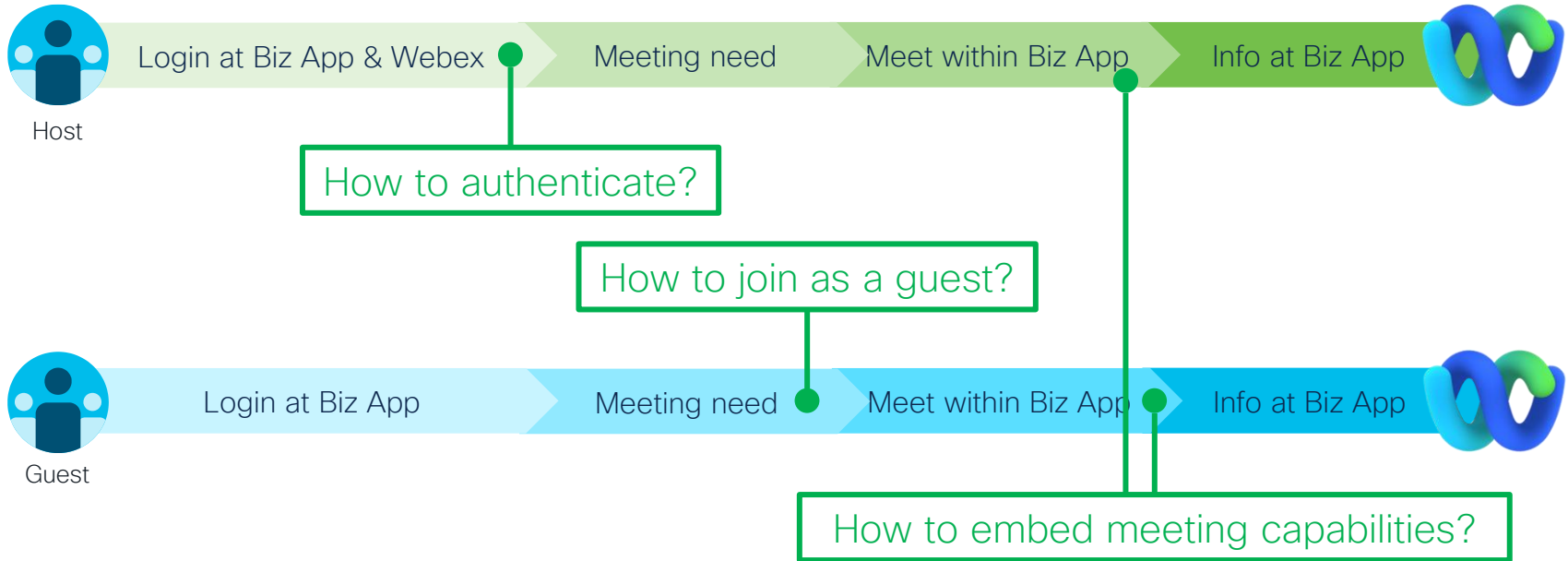
© 2022 Cisco and/or its affiliates. All rights reserved. Cisco Public

11

Embed Webex in Business Workflow

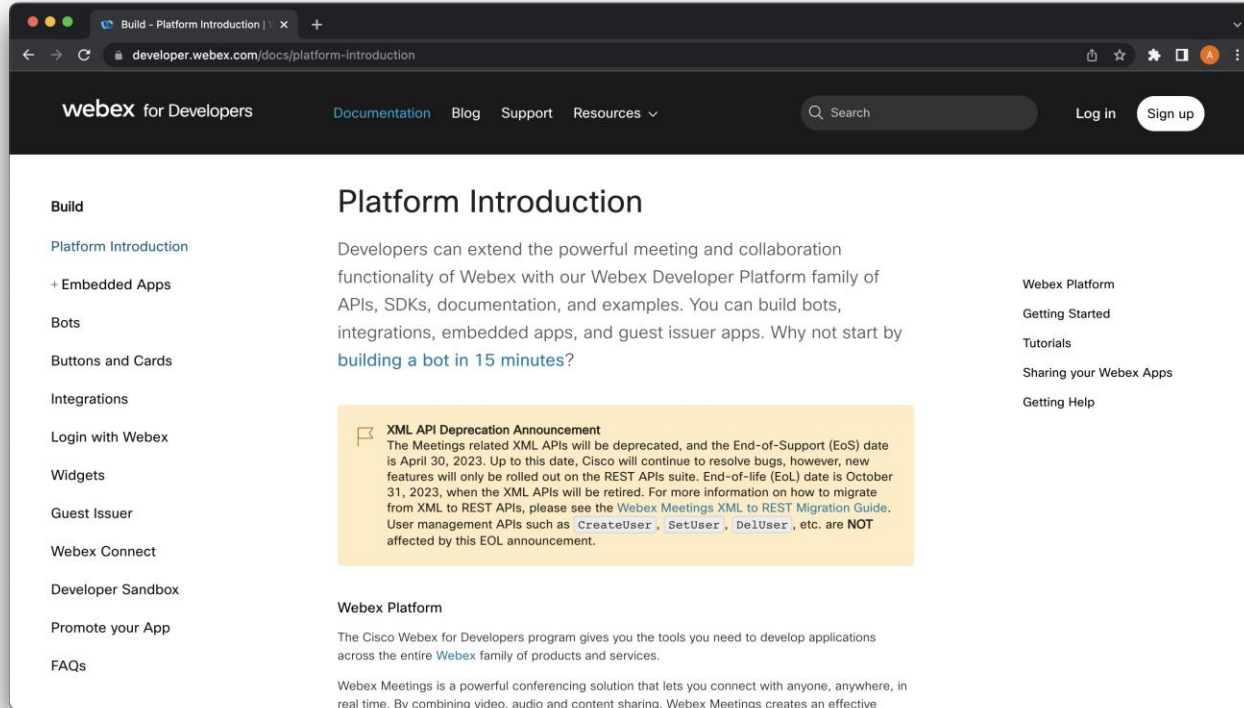


Embed Webex in Business Workflow



Webex for Developers

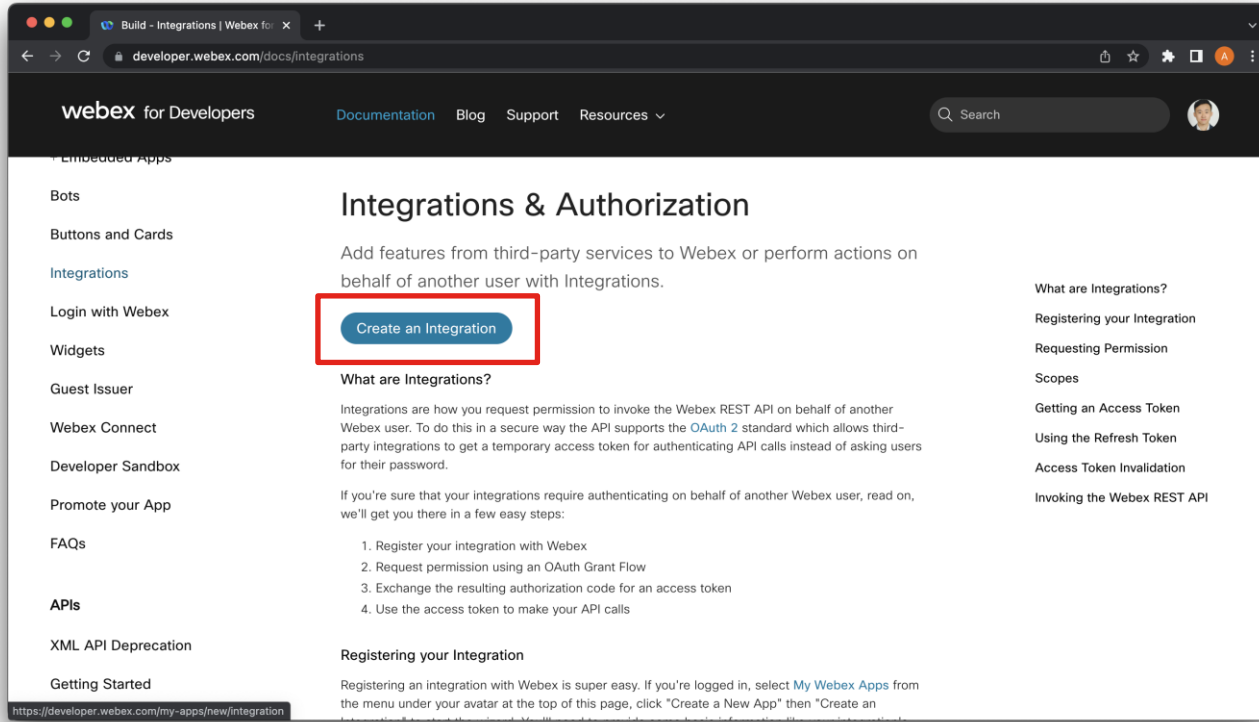
Rich Features for Developers



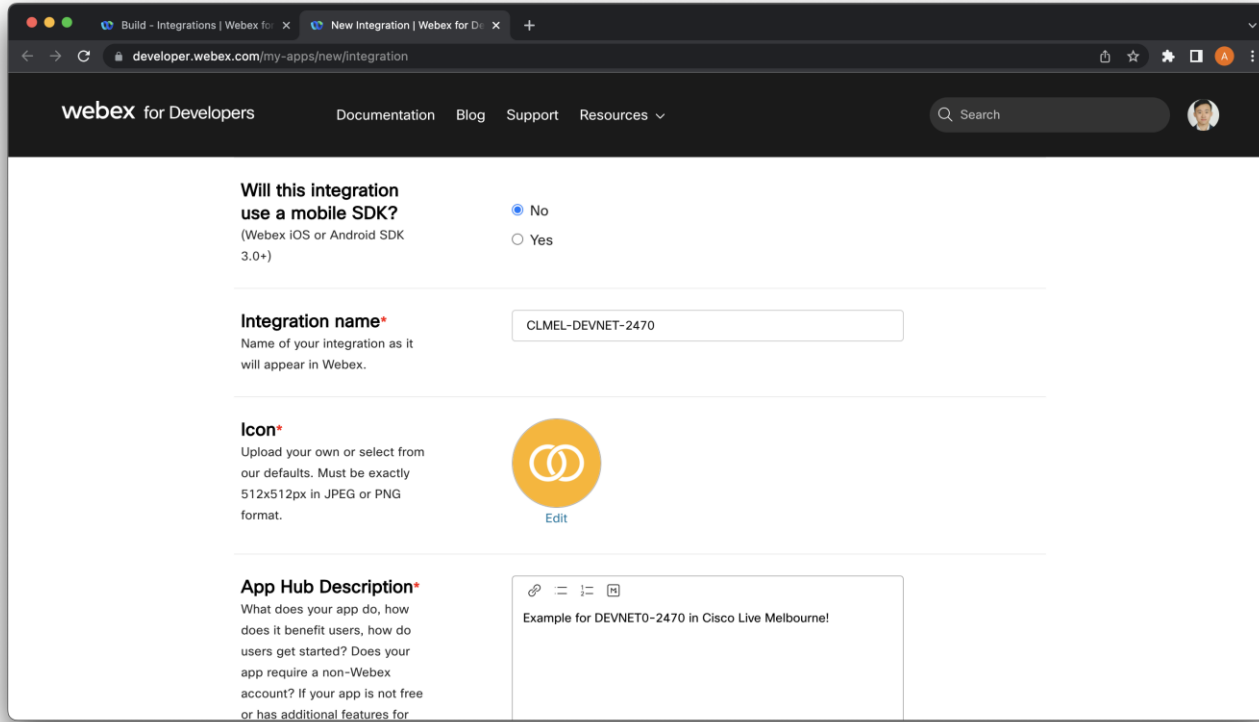
Webex Integrations

- Request permission to perform actions on behalf of a Webex user
1. Register your integration with Webex
 2. Request permission using an OAuth Grant Flow
 3. Exchange the resulting authorization code for an access token
 4. Use the access token to perform actions

Register Your Integration with Webex



Register Your Integration with Webex

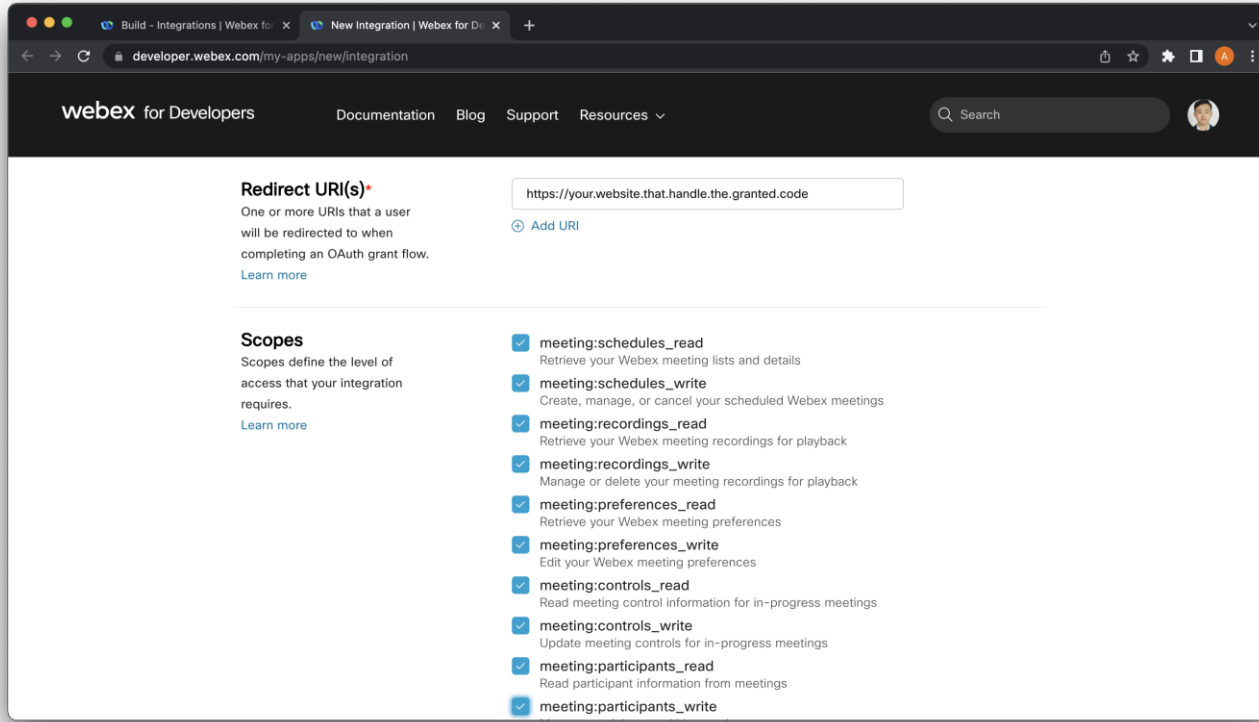


The screenshot shows a web browser window with the URL `developer.webex.com/my-apps/new/integration`. The page has a dark header with the 'webex for Developers' logo and navigation links: 'Documentation', 'Blog', 'Support', and 'Resources'. A search bar and a user profile icon are also present in the header.

The main content area contains the following sections:

- Will this integration use a mobile SDK?**
(Webex iOS or Android SDK 3.0+)
Radio buttons for 'No' (selected) and 'Yes'.
- Integration name***
Name of your integration as it will appear in Webex.
Text input field containing 'CLMEL-DEVNET-2470'.
- Icon***
Upload your own or select from our defaults. Must be exactly 512x512px in JPEG or PNG format.
A circular icon with two interlocking rings is shown. Below it is an 'Edit' link.
- App Hub Description***
What does your app do, how does it benefit users, how do users get started? Does your app require a non-Webex account? If your app is not free or has additional features for...
A rich text editor is shown with the text: 'Example for DEVNET0-2470 in Cisco Live Melbourne!'.

Register Your Integration with Webex



webex for Developers Documentation Blog Support Resources

Search

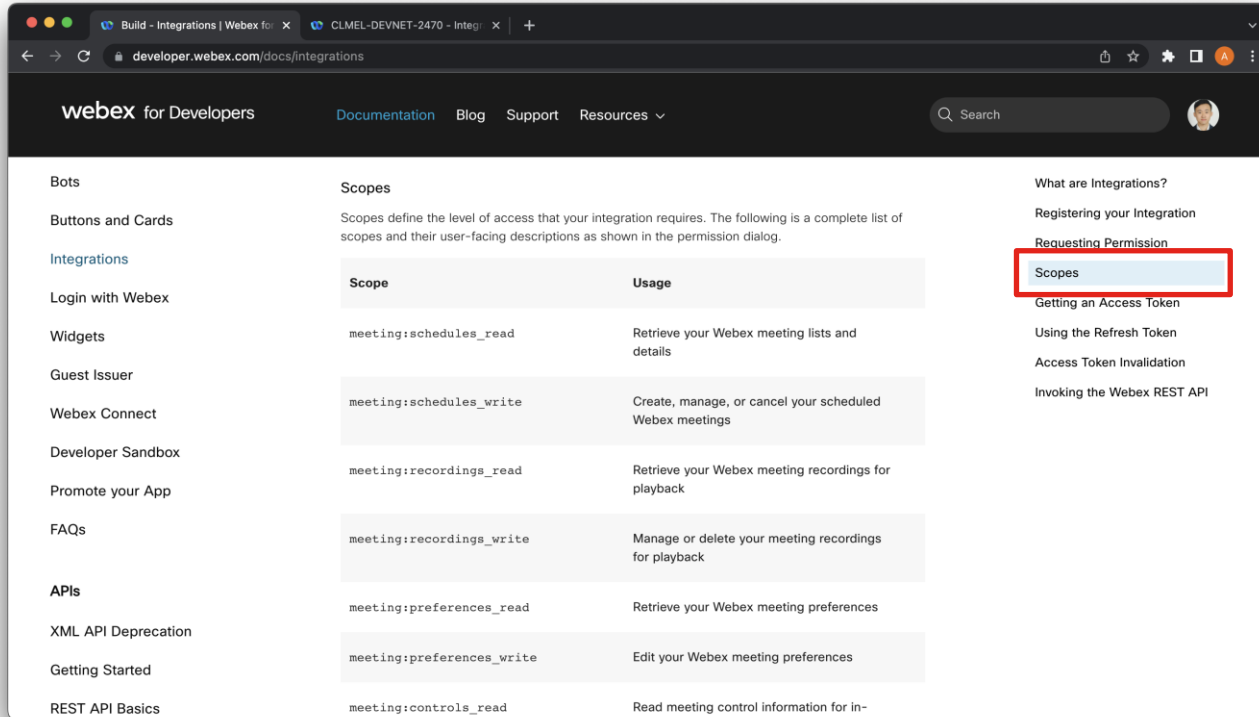
Redirect URI(s)*
One or more URIs that a user will be redirected to when completing an OAuth grant flow.
[Learn more](#)

+ Add URI

Scopes
Scopes define the level of access that your integration requires.
[Learn more](#)

- ☒ **meeting:schedules_read**
Retrieve your Webex meeting lists and details
- ☒ **meeting:schedules_write**
Create, manage, or cancel your scheduled Webex meetings
- ☒ **meeting:recordings_read**
Retrieve your Webex meeting recordings for playback
- ☒ **meeting:recordings_write**
Manage or delete your meeting recordings for playback
- ☒ **meeting:preferences_read**
Retrieve your Webex meeting preferences
- ☒ **meeting:preferences_write**
Edit your Webex meeting preferences
- ☒ **meeting:controls_read**
Read meeting control information for in-progress meetings
- ☒ **meeting:controls_write**
Update meeting controls for in-progress meetings
- ☒ **meeting:participants_read**
Read participant information from meetings
- ☒ **meeting:participants_write**

Register Your Integration with Webex



The screenshot shows the Webex for Developers documentation page. The left-hand navigation menu includes links for Bots, Buttons and Cards, Integrations, Login with Webex, Widgets, Guest Issuer, Webex Connect, Developer Sandbox, Promote your App, FAQs, APIs, XML API Deprecation, Getting Started, and REST API Basics. The main content area is titled 'Scopes' and explains that scopes define the level of access required for an integration. It includes a table with the following data:

Scope	Usage
meeting:schedules_read	Retrieve your Webex meeting lists and details
meeting:schedules_write	Create, manage, or cancel your scheduled Webex meetings
meeting:recordings_read	Retrieve your Webex meeting recordings for playback
meeting:recordings_write	Manage or delete your meeting recordings for playback
meeting:preferences_read	Retrieve your Webex meeting preferences
meeting:preferences_write	Edit your Webex meeting preferences
meeting:controls_read	Read meeting control information for in-

The right-hand navigation menu, titled 'What are Integrations?', includes links for 'Registering your Integration', 'Requesting Permission', 'Scopes' (highlighted with a red box), 'Getting an Access Token', 'Using the Refresh Token', 'Access Token Invalidation', and 'Invoking the Webex REST API'.

Register Your Integration with Webex

webex for Developers Documentation Blog Support Resources

Congratulations! 🎉

Your integration has been created. Use the OAuth credentials below to finish building your integration. When you're ready, return to submit your app to Webex App Hub.

CLMEL-DEVNET-2470

OAuth settings
Learn more about authentication in the [Apps & OAuth Guide](#).

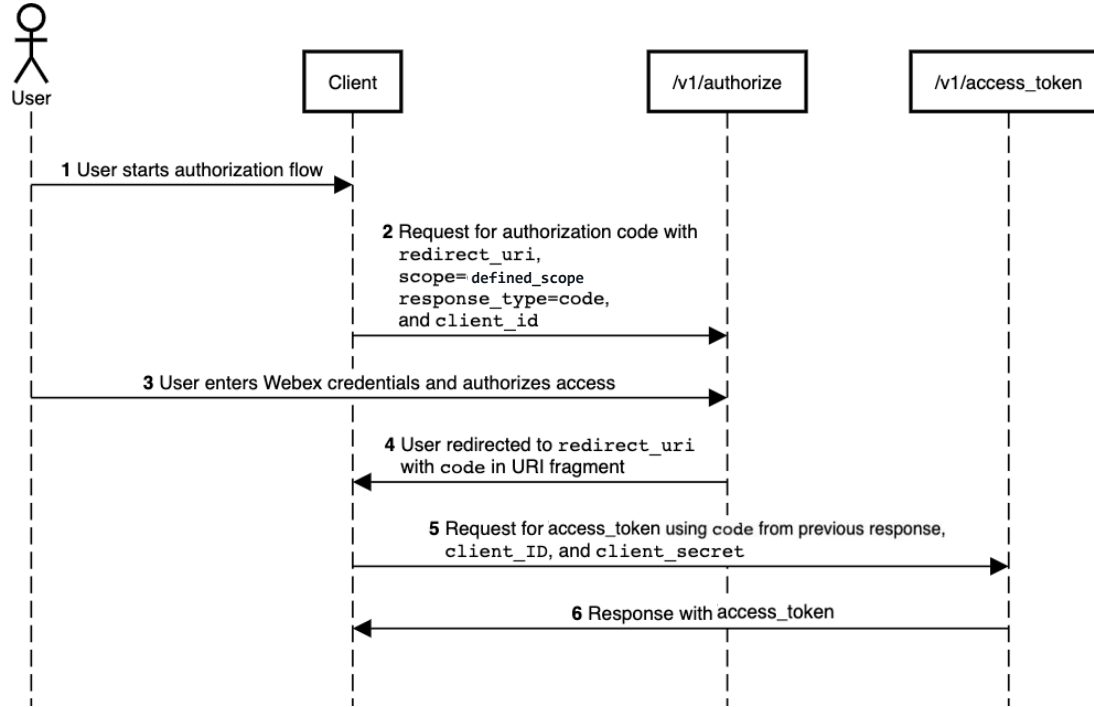
Client ID
Ccdd34bf67341b2e [Copy](#)

Client Secret
3599efc0e1d6a2dc [Copy](#)

OAuth Authorization URL
You can use the URL below to initiate an OAuth permission request for this app. It is configured with your redirect URI and app scopes. Be sure to update the state parameter.

`https://webexapis.com/v1/authorize?
client_id=Ccdd34bf67341b2e&response_type=code&redirect_uri=htt
ps%3A%2F%2Fyour.website.that.handle.the.granted.code&scop
e=meeting%3Arecordings_read%20spark%3Aall%20spark%3Aak
ms%20meeting%3Acontrols_write%20meeting%3Aschedules_r
ead%20meeting%3Aparticipants_read%20meeting%3Acontrols`


OAuth and Access Token Grant Flow



Request Permission using OAuth



United States - English

 CISCO

Log in to your account

Email

Password

[Forgot password?](#)

[Log in](#)

[Create a new account](#)

[Terms & Conditions](#) | [Privacy](#) | [Cookies](#) | [Trademarks](#)



CLMEL-DEVNET-2470
is requesting the following:

- Retrieve your Webex meeting recordings for playback
- Full access to your Webex Teams account
- Allow decryption and encryption
- Update meeting controls for in-progress meetings
- Retrieve your Webex meeting lists and details
- Read participant information from meetings
- Read meeting control information for in-progress meetings
- Edit your Webex meeting preferences
- Retrieve your Webex meeting preferences
- Manage or delete your Webex meeting recordings
- Manage participants within meetings
- Create, manage, or cancel your scheduled Webex meetings

[Accept](#)

☒ Only ask when requesting new permissions.

[Decline](#)

Request Permission using OAuth

HTML frontend

```
<form action="/oauth" method="GET">
  <button>Login with Webex</button>
</form>
```

Node.js Express backend

```
app.get('/oauth', (req, res) => {
  res.redirect(`https://webexapis.com/v1/authorize?client_id=Ccdd34bf67341b2ab843c4[REDACTED]
[REDACTED]&response_type=code&redirect_uri=https%3A%2F%2Fyour.website.that.handle.the.granted.code&scope=meeting%3Arecordings_read%20spark%3Aall%20spark%3Akms%20meeting%3Acontrols_write%20meeting%3Aschedules_read%20meeting%3Aparticipants_read%20meeting%3Acontrols_read%20meeting%3Apreferences_write%20meeting%3Arecordings_write%20meeting%3Apreferences_read%20meeting%3Aparticipants_write%20meeting%3Aschedules_write&state`)
});
```

Exchange Auth Code for Access Token

- After successful authorization, the Auth Code is passed as a parameter with the redirect_url

```
https://your_redirect_url?code=YjAzYzgyNDYtZTE3YS00OWZkLTg2YTgtNDc3Zjg4YzFiZDIkNTRIN2FhMjMtYzUz
```

- Make a POST request to get an access token

Exchange Auth Code for Access Token

Node.js Express backend

```
async function getAccessToken(code) {  
  const res = await axios  
    .post(`https://webexapis.com/v1/access_token`,  
      {  
        'grant_type': 'authorization_code',  
        'client_id': process.env.WEBEX_CLIENT_ID,  
        'client_secret': process.env.WEBEX_CLIENT_SECRET,  
        'code': code,  
        'redirect_uri': `${process.env.APP_URL}/oauth/redirect`  
      },  
      {  
        'Content-Type': 'application/x-www-form-urlencoded'  
      }  
    )  
    .catch(error => console.log(error.data));  
  return res.data.access_token;  
}
```

payload

header

Exchange Auth Code for Access Token

- Use refresh token to refresh access token

Result

```
{  
  "access_token": "ZDI3MGEyYzQtNmFlNS00NDNhLWF1NzAtZGVjNjE0MGU1OGZmZWNmZDEwN2ItYTU3",  
  "expires_in": 1209600, //seconds  
  "refresh_token": "MDEyMzQ1Njc4OTAxMjM0NTY3ODkwMTIzNDU2Nzg5MDEyMzQ1Njc4OTIyMzQ1Njc4",  
  "refresh_token_expires_in": 7776000 //seconds  
}
```

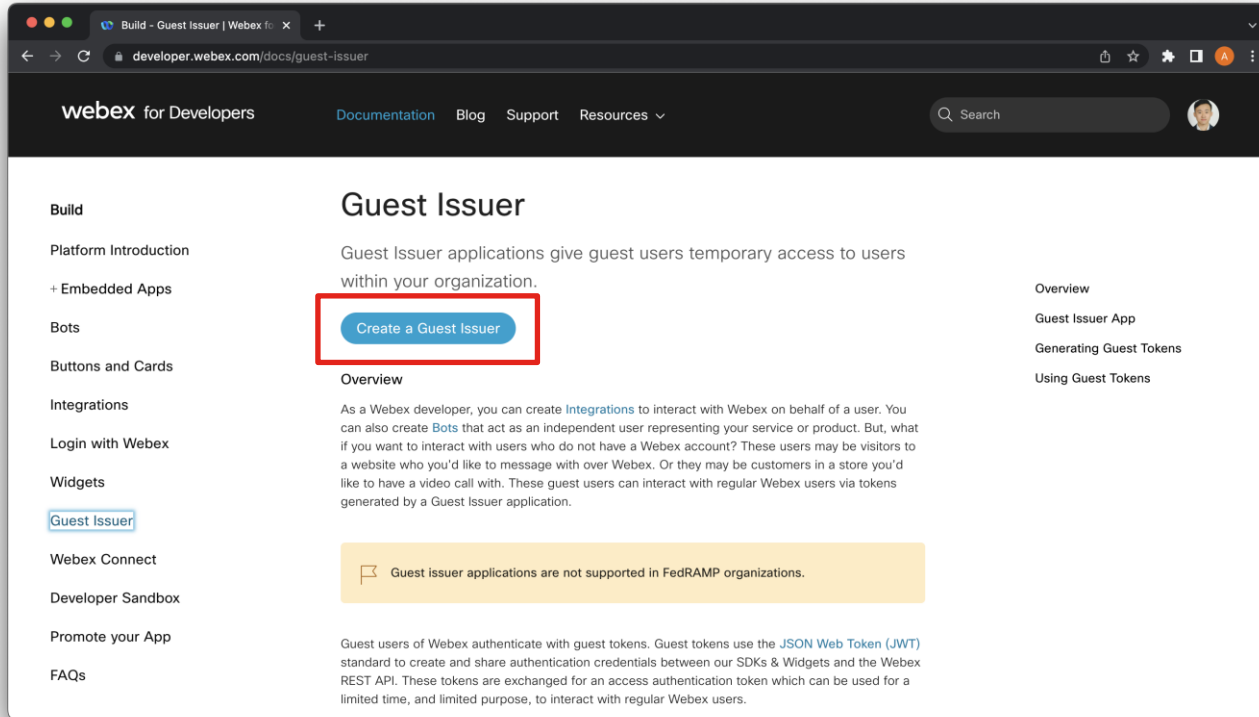
Webex Guest Issuer

- Create temporary access to Webex for guest users
 - Activities within Webex such as message and call history will persist
- Allow guest users to interact with regular Webex users
- Guest-to-guest interaction is not allowed

Webex Guest Issuer

1. Register your Guest Issuer with Webex
2. Generate guest tokens
3. Exchange a guest token for an access token

Register Your Guest Issuer with Webex



The screenshot shows a web browser window with the URL `developer.webex.com/docs/guest-issuer`. The page title is "Guest Issuer" under the "webex for Developers" header. The left sidebar lists navigation items: Build, Platform Introduction, + Embedded Apps, Bots, Buttons and Cards, Integrations, Login with Webex, Widgets (highlighted), Guest Issuer (highlighted), Webex Connect, Developer Sandbox, Promote your App, and FAQs. The main content area has the heading "Guest Issuer" and a subheading "Guest Issuer applications give guest users temporary access to users within your organization." Below this is a blue button labeled "Create a Guest Issuer" which is highlighted with a red rectangle. To the right of the main content is a sidebar with links: Overview, Guest Issuer App, Generating Guest Tokens, and Using Guest Tokens. Below the "Create a Guest Issuer" button is an "Overview" section with text explaining that Guest Issuer applications are not supported in FedRAMP organizations. At the bottom, there is a section about guest tokens using JSON Web Tokens (JWT).

webex for Developers

Documentation Blog Support Resources

Search

Build

Platform Introduction

+ Embedded Apps

Bots

Buttons and Cards

Integrations

Login with Webex

Widgets

Guest Issuer

Webex Connect

Developer Sandbox

Promote your App

FAQs

Guest Issuer

Guest Issuer applications give guest users temporary access to users within your organization.

Create a Guest Issuer

Overview

As a Webex developer, you can create [Integrations](#) to interact with Webex on behalf of a user. You can also create [Bots](#) that act as an independent user representing your service or product. But, what if you want to interact with users who do not have a Webex account? These users may be visitors to a website who you'd like to message with over Webex. Or they may be customers in a store you'd like to have a video call with. These guest users can interact with regular Webex users via tokens generated by a Guest Issuer application.

Guest issuer applications are not supported in FedRAMP organizations.

Guest users of Webex authenticate with guest tokens. Guest tokens use the [JSON Web Token \(JWT\)](#) standard to create and share authentication credentials between our SDKs & Widgets and the Webex REST API. These tokens are exchanged for an access authentication token which can be used for a limited time, and limited purpose, to interact with regular Webex users.

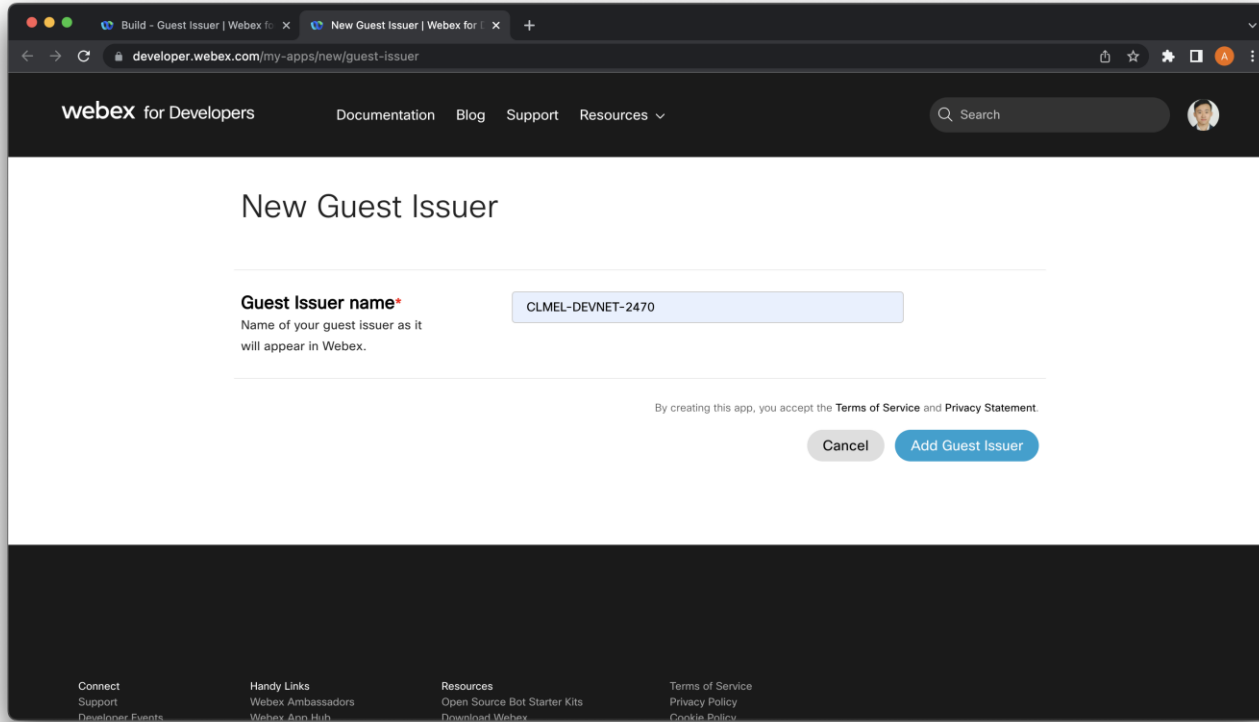
Overview

Guest Issuer App

Generating Guest Tokens

Using Guest Tokens

Register Your Guest Issuer with Webex



The screenshot shows a web browser window with two tabs: 'Build - Guest Issuer | Webex for Developers' and 'New Guest Issuer | Webex for Developers'. The address bar shows 'developer.webex.com/my-apps/new/guest-issuer'. The page header includes the 'webex for Developers' logo, navigation links for 'Documentation', 'Blog', 'Support', and 'Resources', a search bar, and a user profile icon. The main heading is 'New Guest Issuer'. Below it is a form with a label 'Guest Issuer name*' and a description 'Name of your guest issuer as it will appear in Webex.' The input field contains the text 'CLMEL-DEVNET-2470'. Below the form, a message states 'By creating this app, you accept the Terms of Service and Privacy Statement.' At the bottom of the form are two buttons: 'Cancel' and 'Add Guest Issuer'. The footer contains four columns of links: 'Connect' (Support, Developer Events), 'Handy Links' (Webex Ambassadors, Webex App Hub), 'Resources' (Open Source Bot Starter Kits, Download Webex), and 'Terms of Service' (Privacy Policy, Cookie Policy).

webex for Developers Documentation Blog Support Resources

Search

New Guest Issuer

Guest Issuer name*
Name of your guest issuer as it will appear in Webex.

CLMEL-DEVNET-2470

By creating this app, you accept the [Terms of Service](#) and [Privacy Statement](#).

Cancel Add Guest Issuer

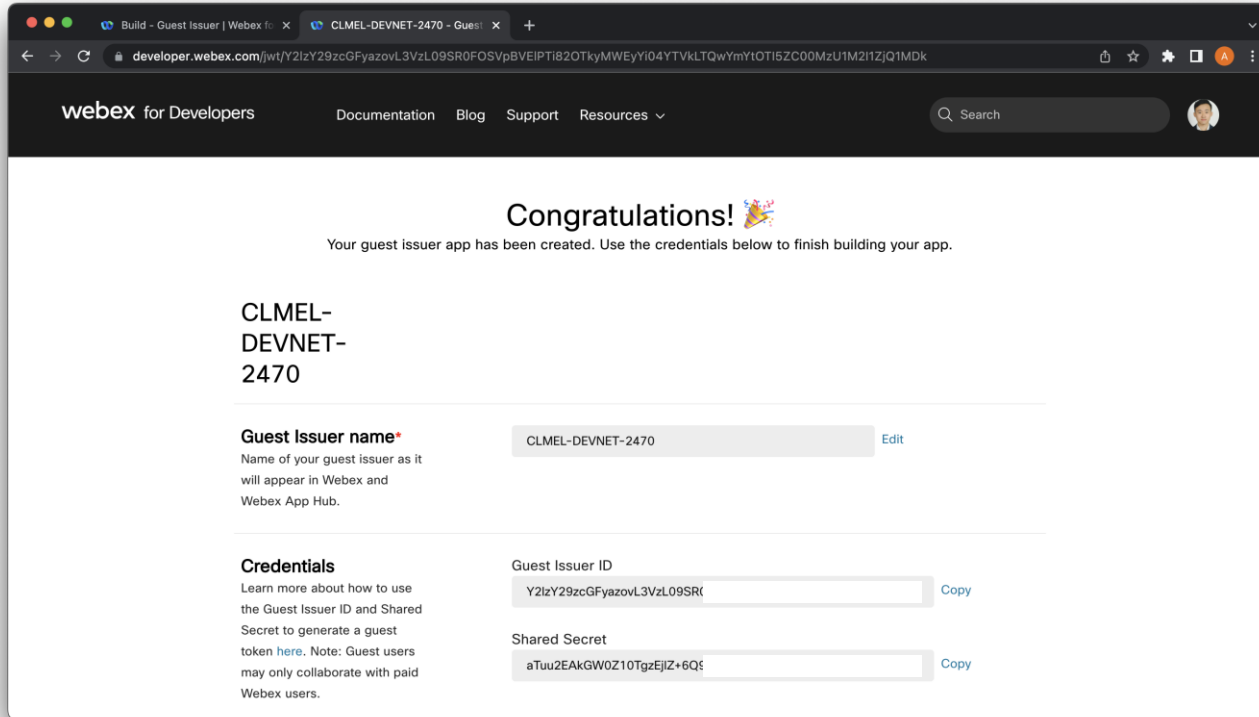
Connect
Support
Developer Events

Handy Links
Webex Ambassadors
Webex App Hub

Resources
Open Source Bot Starter Kits
Download Webex

Terms of Service
Privacy Policy
Cookie Policy

Register Your Guest Issuer with Webex



The screenshot shows a web browser window with the URL `developer.webex.com/jwt/Y2lzY29zcGFyazovL3VzL09SR0FOSVpBEVPTi82OTkyMWEyYi04YTUvLTQwYmY1OTI1ZjQ1MDk`. The page has a dark header with the 'webex for Developers' logo and navigation links: 'Documentation', 'Blog', 'Support', and 'Resources'. A search bar and a user profile icon are also present.

The main content area features a 'Congratulations!' message with a confetti icon, stating: 'Your guest issuer app has been created. Use the credentials below to finish building your app.'

Below the message, the guest issuer name 'CLMEL-DEVNET-2470' is displayed. Under the heading 'Guest Issuer name*', there is a text input field containing 'CLMEL-DEVNET-2470' and an 'Edit' link. A note explains: 'Name of your guest issuer as it will appear in Webex and Webex App Hub.'

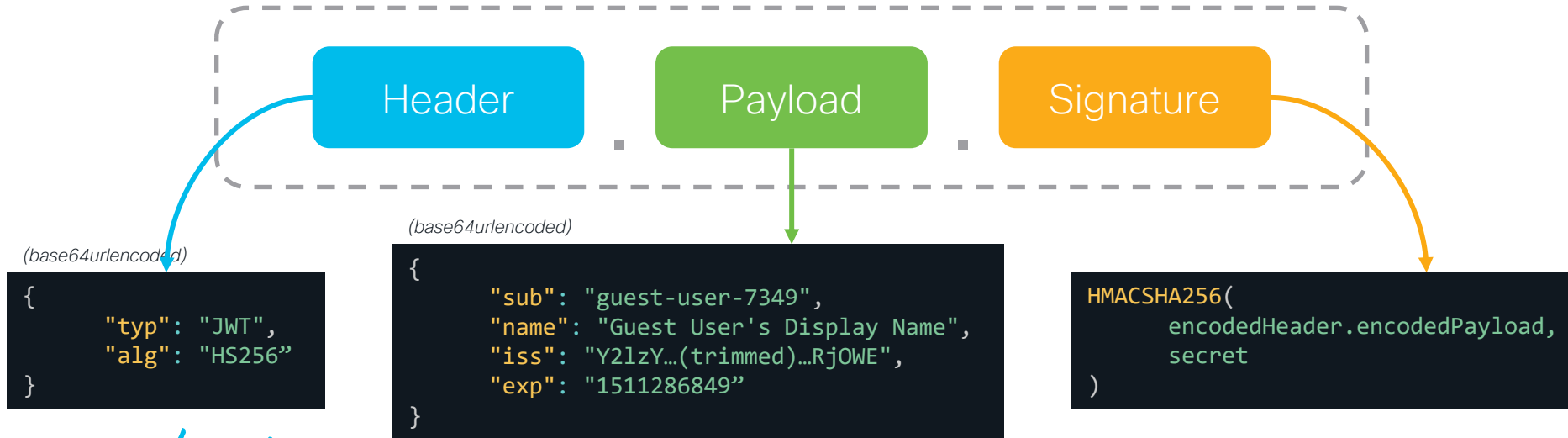
The 'Credentials' section provides instructions: 'Learn more about how to use the Guest Issuer ID and Shared Secret to generate a guest token here. Note: Guest users may only collaborate with paid Webex users.'

Two credential fields are shown:

- Guest Issuer ID:** A text input field containing 'Y2lzY29zcGFyazovL3VzL09SR0FOSVpBEVPTi82OTkyMWEyYi04YTUvLTQwYmY1OTI1ZjQ1MDk' with a 'Copy' button.
- Shared Secret:** A text input field containing 'aTuuz2EakGW0Z10TgzEjIZ+6Q5' with a 'Copy' button.

Generate Guest Tokens

- Use JSON Web Token (JWT) standard
- Composed of three parts, separated by a dot (.)



Generate Guest Tokens

Node.js Express backend

```
const jwt = require('jsonwebtoken');
function createJwtToken() {
  const payload = {
    name: 'Cisco HK Guest'
  };
  const options = {
    expiresIn: 604800,
    issuer: process.env.GUEST_ISSUER_ID,
    subject: 'cisco-hk-guest'
  }
  const token = jwt.sign(
    payload,
    Buffer.from(process.env.GUEST_ISSUER_SECRET, 'base64'),
    options
  );
  return token;
}
```

Exchange Guest Token for Access Token

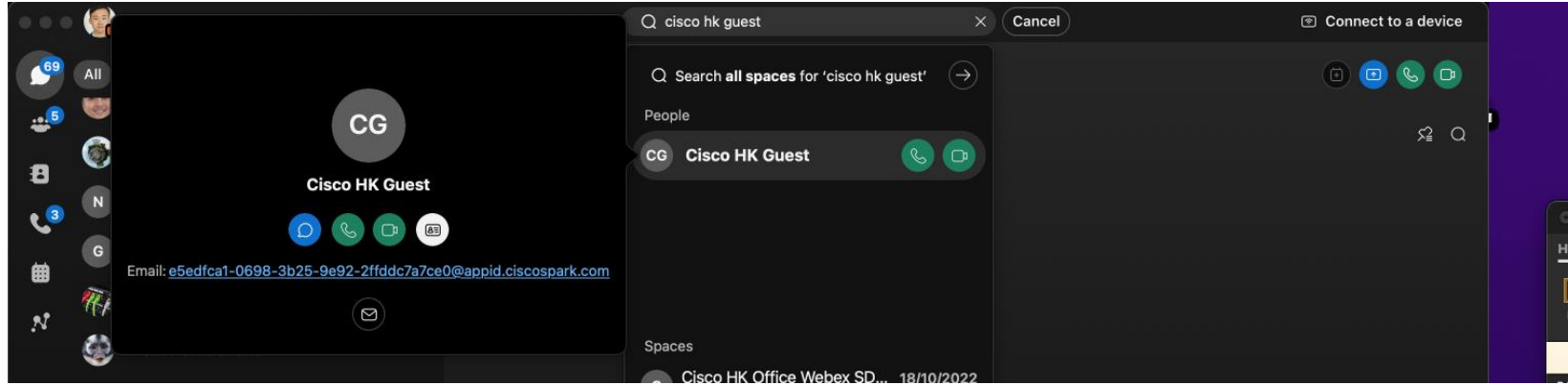
- Webex accounts for guest users are created only after authenticating with a guest token
- Guest users will not have a valid email address associated with their accounts
- Use their individual “personId” for identification

Exchange Guest Token for Access Token

Node.js Express backend

```
async function createGuest() {  
  const jwtToken = createJwtToken();  
  const config = {  
    method: 'post',  
    url: 'https://webexapis.com/v1/jwt/login',  
    headers: {  
      'Authorization': `Bearer ${jwtToken}`  
    }  
  };  
  const res = await axios(config)  
    .then(response => {  
      return response;  
    })  
    .catch(error => console.log(error.data));  
  return res.data.token;  
}
```

Guest Account Creation



Webex Connect

- Previously known as imimobile
- Enterprise CPaaS platform that enables orchestration and automation for customer-employee interactions
- Fully programmable with choice of API, SDK and low code tool

Webex Connect

1. Setup Webex Connect account
2. Copy code snippets from dashboard
3. Send SMS to inform meeting links

Code Snippets and SMS

The screenshot shows the Cisco Connect dashboard at `sandbox.imconnect.io/cpaas-dashboards/sms`. The 'SMS' tab is selected. On the left, the 'Send SMS' form includes fields for 'Service' (My First Service), 'Service Key', 'From Number' (CONNECT), 'To Number' (+85291...), and 'Message Type' (Text). A 'Send SMS' button is at the bottom. On the right, the 'API / SDK Code Snippets' section is highlighted with a red box, showing a Node.js code snippet for sending an SMS. Below it, the 'SMS API Response' section is visible.

API / SDK Code Snippets

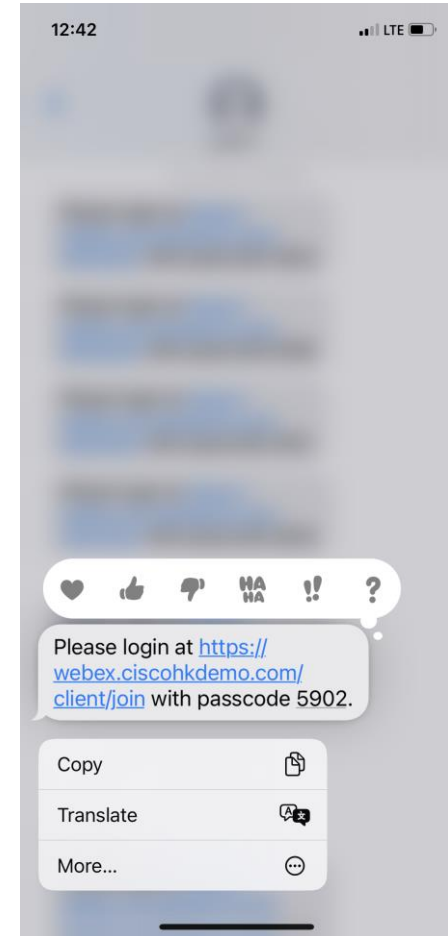
```
const { SmsClient, SmsMessage, ClientConfiguration } = require('connect-sdk-node');
const clientConfiguration = new ClientConfiguration("495!
063d0d6dfdb5", "https://api-sandbox.imconnect.io");
const smsClient = new SmsClient(clientConfiguration);

const smsMessage = new SmsMessage().of_text("CONNECT", "+85291 6", "");

const request = smsClient.sendMessage(smsMessage);

request
```

SMS API Response



Webex SDK



- Swift 5 or later
- Xcode 12 or later
- iOS 13 or later
- CocoaPods



- Android Studio 4.0 or later
- Android SDK Tools 29 or later
- Android API Level 24 or later
- Java JDK 8
- Kotlin 1.3.+
- Gradle



- JavaScript/WebRTC
- Desktop:
Chrome/Firefox/Edge/Safari
- Mobile: Chrome for Android;
Safari for iOS

Webex Browser SDK

- Add Webex products and features to your web apps

Features

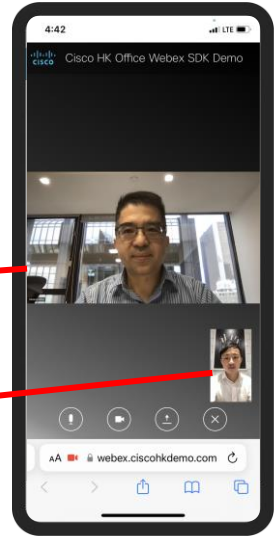
- Create and join audio and/or video meetings
- Create by email address, SIP address, or PSTN number. (Note: Only SIP URI is supported for cloud-registered Webex devices, and isn't supported for third-party SIP endpoints or Webex Calling.)
- Meeting and event controls
- Audio and video meeting control
- Maximum bandwidth controls
- View shared content
- Share screen/application from desktop browsers: Chrome, Firefox, Edge (80+), and Safari (12.1+)
- Device Controls (get device list, send PIN, pair)
- Device Search
- Personal Meeting Room support
- Move Media
- Moderator Controls
- Reception of Real-Time Transcripts. (Note: Real-Time Transcription is ONLY supported by meetings where Webex Assistant can be enabled, which requires a paid license.)
- FedRAMP environment support. See [usage README](#)
- [Background Noise Reduction](#)



Webex Browser SDK – HTML

HTML video and audio elements

```
<div class="video-panel">
  <video id="remote" autoplay></video>
  <audio id="remote-audio" autoplay></audio>
</div>
<div class="video-panel">
  <video id="local" muted autoplay></video>
</div>
```



Link to Webex SDK CDN and frontend javascript

```
<script crossorigin src="https://unpkg.com/webex@^1/umd/webex.min.js"></script>
<script src="./index.js"></script>
```

Webex Browser SDK – JavaScript

Authenticate with Webex

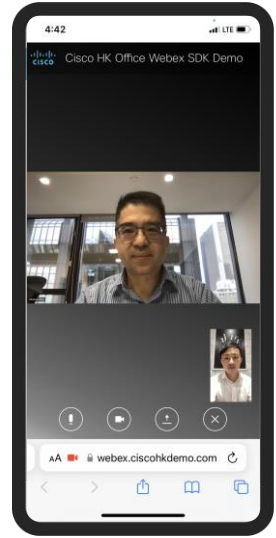
```
const webex = window.Webex.init({  
  credentials: { access_token: "YOUR_ACCESS_TOKEN" }  
});
```

Register with Webex Cloud

```
webex.meetings.register();
```

Set up event listeners for meeting events

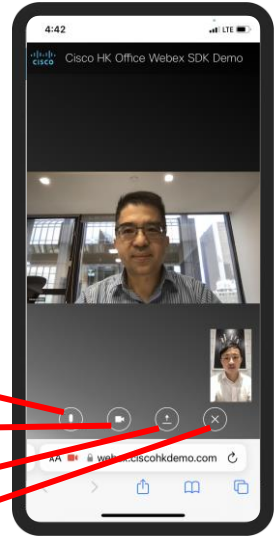
```
meeting.on('media:ready', (media) => {  
  if (media.type === 'local') {  
    document.getElementById('local').srcObject = media.stream;  
  }  
  if (media.type === 'remoteVideo') {  
    document.getElementById('remote').srcObject = media.stream;  
  }  
  if (media.type === 'remoteAudio') {  
    document.getElementById('remote-audio').srcObject = media.stream;  
  }  
});
```



Webex Browser SDK – JavaScript

Set up event listeners for meeting controls

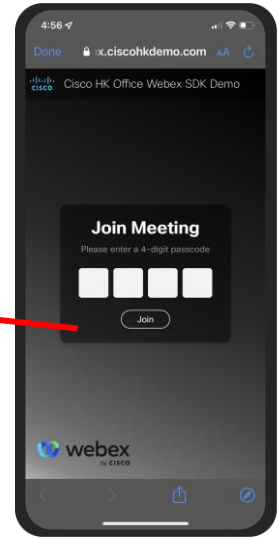
```
$('#audio-control').on('click', () => {  
    meeting.muteAudio().then(() => {  
        console.log('audio is muted');  
    });  
});  
  
$('#video-control').on('click', () => {  
    meeting.muteVideo().then(() => {  
        console.log('video is off');  
    })  
});  
  
$('#share-start').on('click', () => {  
    meeting.shareScreen();  
});  
  
$('#hangup').on('click', () => {  
    meeting.leave();  
});
```



Webex Browser SDK - JavaScript

Set up join meeting handler and join button

```
function joinMeeting(meeting) {  
  return meeting.join().then(() => {  
    const mediaSettings = {  
      receiveVideo: true,  
      receiveAudio: true,  
      receiveShare: true,  
      sendVideo: true,  
      sendAudio: true,  
      sendShare: false  
    };  
  
    return meeting.getMediaStreams(mediaSettings).then((mediaStreams) => {  
      const [localStream, localShare] = mediaStreams;  
      meeting.addMedia({ localShare, localStream, mediaSettings });  
    });  
  });  
}  
  
function startMeeting() {  
  return webex.meetings.create(room_id).then((meeting) => {  
    joinMeeting(meeting);  
    return bindMeetingEvents(meeting);  
  });  
}
```



Webex Browser SDK

Samples

We've put together several samples of the Browser SDK in action. All of the sample code is in GitHub. Take the SDK for a spin with a live demo and check out the sample code:

Sample	Description	Live Demo	Sample Code
Single Party Call	Make or receive 1:1 video calls with another Webex user	Demo	JS HTML
Single Party Call with Mute	Make or receive 1:1 video calls with another Webex user, with the option to mute your audio and/or video streams	Demo	JS HTML
Multi Party Call	Start or join a Webex video call with multiple participants	Demo	JS HTML
Call with Content Sharing	Make or receive video calls with the ability to share your screen or an application	Demo	JS HTML

Meeting Controls

Once in a meeting utilize these controls to manipulate the meeting.

Please note that you must be a meeting moderator in order to lock or record a meeting.

General Controls

meeting.lockMeeting()

meeting.unlockMeeting()

meeting.startRecording()

meeting.pauseRecording()

meeting.resumeRecording()

meeting.stopRecording()

Enter DTMF tones

meeting.sendDTMF()

start receiving transcription (click me before joining)

stop receiving transcription

Transcription Content: Webex Assistant must be enabled, check the console!

Meeting Streams

Local Video

Local Screenshot

Remote Video

Remote Screenshot

Use Case Demo

Call to Action

- Think about the **customized Webex embedded workflow** in your business applications and talk to us!
- Check out **Webex for Developer** website
 - <https://developer.webex.com/>
- **Browser SDK Sample** live demo and GitHub repository
 - <https://webexsamples.github.io/browser-sdk-samples/browser-plugin-meetings/>
 - <https://github.com/WebexSamples/browser-sdk-samples/tree/main/browser-plugin-meetings>
- **Use Case Demo** GitHub repository
 - (to be provided)

Session Surveys

We would love to know your feedback on this session!

- Complete the session surveys in the Cisco Events mobile app. You'll earn some points in the Cisco Live Game and potentially win a prize.
- Complete a minimum of four session and the overall event surveys to claim a Cisco Live cable bag.

Continue your education



Visit the Cisco Showcase for related demos



Book your one-on-one Meet the Expert meeting



Attend the interactive education with DevNet, Capture the Flag, and Walk-in Labs



Visit the On-Demand Library for more sessions at www.CiscoLive.com/on-demand

Cisco Learning and Certifications

From technology training and team development to Cisco certifications and learning plans, let us help you empower your business and career. www.cisco.com/go/certs

Pay for Learning with Cisco Learning Credits

(CLCs) are prepaid training vouchers redeemed directly with Cisco.



Learn

Cisco U.

IT learning hub that guides teams and learners toward their goals

Cisco Digital Learning

Subscription-based product, technology, and certification training

Cisco Modeling Labs

Network simulation platform for design, testing, and troubleshooting

Cisco Learning Network

Resource community portal for certifications and learning



Train

Cisco Training Bootcamps

Intensive team & individual automation and technology training programs

Cisco Learning Partner Program

Authorized training partners supporting Cisco technology and career certifications

Cisco Instructor-led and Virtual Instructor-led training

Accelerated curriculum of product, technology, and certification courses



Certify

Cisco Certifications and Specialist Certifications

Award-winning certification program empowers students and IT Professionals to advance their technical careers

Cisco Guided Study Groups

180-day certification prep program with learning and support

Cisco Continuing Education Program

Recertification training options for Cisco certified individuals





The bridge to possible

Thank you

CISCO *Live!*

DevNet Zone

#CiscoLiveAPJC

CISCO *Live!*



#CiscoLiveAPJC