

```
<apex:page showHeader="false" sidebar="false" standardStylesheets="false"
applyHtmlTag="true">
  <html>
    <head>
      <meta charset="utf-8" />
      <meta http-equiv="X-UA-Compatible" content="IE=edge"/>
      <meta name="viewport" content="width=device-width, initial-scale=1"/>
      <apex:includeScript value="{!URLFOR($Resource.config)}"/>
      <apex:includeScript value="{!URLFOR($Resource.SkypeBootstrap)}"/>
      <style>
        #conversationWindow{
          height: 88vh;
        }
        button {
          padding: 11px;
          background-color: cornflowerblue;
          border: 1px solid aliceblue;
          border-radius: 5px;
          color: white;
          font-size: 1rem;
        }
        body{
          background-color: #f5fafc;
        }
        input{
          background-color: #fff;
          border: 1px solid #dddbda;
          border-radius: .25rem;
          width: 30%;
          transition: border .1s linear,background-color .1s linear;
          display: inline-block;
          padding: 0 1rem 0 .75rem;
          line-height: 1.875rem;
          min-height: calc(1.875rem + (1px * 2));
        }
        .hideData{
          display:none;
        }
        .showData{
          display:block;
        }
      </style>
    </head>
```

```

<body>
  <div align="center">
    <button id="AdminConsent">Admin Consent</button>
    <button id="btnLog">Sign-in O365</button>
    <button id="initSkypeUI">Initialize SkypeUI</button>
    <button id="loginSkype">Sign-in Skype</button>
    <button id="meetingSkype">Schedule Skype meeting</button>
    <button id="startChat">Start Chat</button>
    <div id="SdkStatus"></div>
    <div id="SigninStatus"></div>
    <div >
      <div>
        <p>Meeting User Name</p>
        <input type="text" label="User Name" name="userName" id="userName"
value="sadanand.kumar@skypecloudanalogy.onmicrosoft.com"/>
      </div>
      <div >
        <p>Meeting Expiration Date</p>
        <input type="date" id="meetingtime" label="Ecpiration Date" />
      </div>
      <div>
        <p>Meeting Subject</p>
        <input type="text" label="Subject" name="subject" id="subject"/>
      </div>
    </div>
  </div>
  <div id="conversationWindow"></div>
  <script type="text/javascript">
    var apiManager = null;
    var client = null;

    (function () {
      //Retrieves access token from URL fragment
      if(location.hash) {
        var hasharr = location.hash.substr(1).split("&");
        hasharr.forEach(function(hashelem) {
          var elemarr = hashelem.split("=");
          if(elemarr[0] == "access_token") {
            console.log('Access Token: ' + elemarr[1]);
          }
        }, this);
      }
    })
  </script>

```

/*

Sign-in to Office 365 with Azure Active Directory
The document below describe authentication

<https://msdn.microsoft.com/en-us/skype/websdk/developwebsdkappsforfbonline#authenticate-a-user-with-office-365-online>

*/

```
document.getElementById('btnLog').onclick = function () {  
    location.assign(  
        config.loginurl+  
        '&client_id='+config.clientid+  
        '&resource='+config.resource+  
        '&redirect_uri='+ config.replyurl  
    );  
};
```

/*

Grant app with Admin Consent
You can see more information in the document below

<https://msdn.microsoft.com/en-us/skype/websdk/developwebsdkappsforfbonline#tenant-administrator-consent-flow>

*/

```
document.getElementById('AdminConsent').onclick= function(){  
  
location.assign('https://login.microsoftonline.com/common/oauth2/authorize?response_type=id_token'+  
    '&client_id='+config.clientid+  
    '&redirect_uri='+config.replyurl+  
    //'&response_mode=form_post'+ //Because it is static web page and cant'  
handle post method, I comment out it.  
    '&nonce=samplestring'+// You can add code to defend Replay Attack with  
nonce property  
    '&resource='+config.resource+  
    '&prompt=admin_consent'  
    );  
};
```

/*

Initializing Skype Web SDK & Conversation Control.
You can learn more with the document below

<https://msdn.microsoft.com/en-us/skype/websdk/useconversationcontrol#initialize-the-skype-we>

b-sdk-api-endpoint

```
        */
        document.getElementById('initSkypeUI').onclick = function () {
            Skype.initialize({apiKey:config.apiKeyCC},function(api){
                apiManager = api;
                client = apiManager.UIApplicationInstance;
                document.getElementById('SdkStatus').innerHTML = 'Skype Web SDK &
Conversation Control Initialize success!';

                //Event handler: whenever app state changes, display its value
                client.signInManager.state.changed(function (state) {
                    console.log(state);
                });

                //Event handler : it is handled when conversations added. ie. incoming call,
outgoing call
                client.conversationsManager.conversations.added(function(conversation){

                    //Event handler : Get notified when conversation control receives an incoming
call
                    conversation.selfParticipant.audio.state.changed(function (newValue, reason,
oldValue) {

                        // 'Notified' indicates that there is an incoming call
                        if(newValue === 'Notified') {
                            var sip = conversation.participants(0).person.id();

                            if (confirm("Would you like to accept this incoming call?")) {
                                setTimeout(function() {
                                    var container = document.getElementById(sip);
                                    if(!container){
                                        container = document.createElement('div');
                                        container.id = sip;

document.getElementById("conversationWindow").appendChild(container);

                                    //render conversation controll
                                    var promise = apiManager.renderConversation(container,{
                                        //Start outgoing call with chat and audio
                                        modalities:['Chat','Audio'],
                                        participants : [sip]
                                    }).then(function(conversation){
                                        console.log('Conversation rendered successfully');
                                        loginSkype();
```

```

    });
    }
    // This accepts an incoming call with audio
    conversation.audioService.accept();
    // To accept an incoming call with video enabled call
    // conversation.videoService.accept()
    }, 0);

    } else {
        // Reject the incoming call
        conversation.audioService.reject();
    }
}
});
}, function(err){
    console.log(err);
});
};

```

/*

Sign-in Skype

You can learn more from the link below

<https://msdn.microsoft.com/en-us/skype/websdk/developwebsdkappsforfbonline#support-for-oauth2-in-internet-explorer>

```

*/
document.getElementById('loginSkype').onclick = function (){
    var params =
    {
        "client_id": config.clientid,
        "origins":
["https://webdir.online.lync.com/autodiscover/autodiscover.service.svc/root"],
        "cors": true,
        "version": config.appName+'1.0.0',
        "redirect_uri":
"https://sadanand-dev-ed--c.ap5.visual.force.com/apex/Skypeintegration"
    };

    client.signInManager.signIn(params).then(function(){
        document.getElementById('SignInStatus').innerHTML = 'Signed in as ' +
client.personsAndGroupsManager.mePerson.displayName();
    });
}

```

```

    },function(error){
        console.log(error);
    });
};

```

```

/*

```

Render conversation control on outgoing call
 You can get more information with the reference below.

<https://msdn.microsoft.com/en-us/skype/websdk/useconversationcontrol#render-conversation-control-on-outgoing-call>

```

    */
    document.getElementById('startChat').onclick=function(){
        //show prompt in the browser
        var input = prompt('SIP URIs of the
participants:', 'sip:sadanand.kumar@skypecloudanalogy.onmicrosoft.com');

        if(!input)
            return;
        var uris = input.split(',').map(function(s){return s.trim();});
        console.log('participants:', uris);

        var container = document.getElementById(input);
        if(!container){
            container = document.createElement('div');
            container.id = input;
            document.getElementById("conversationWindow").appendChild(container);
        }
        //render Conversation Control in a web page.
        var promise = apiManager.renderConversation(container,{
            //Start outgoing call with chat window
            modalities:['Chat'],
            participants: uris
        });
    };
    /* schedule skype meeting start*/
    document.getElementById('meetingSkype').onclick=function(){
        try{
            var userName = "";
            userName += document.getElementById('userName').value;
            if(userName != undefined && userName != null &&
/^\\w+([\\.-]?\\w+)*@\\w+([\\.-]?\\w+)*\\.\\w{2,3}+$/ .test(document.getElementById('userName').value)
        ){

```

```

        if (document.getElementById('meetingtime').value != undefined &&
(document.getElementById('meetingtime').value).match(/^\d\d\d\d-(0?[1-9]|1[0-2])-(0?[1-9]|12)[0
-9]|3[01])$/)) {
            if (document.getElementById('subject').value != undefined &&
!(document.getElementById('subject').value).match(/^\s*$/)) {
                Date.prototype.addDays = function(days) {
                    var date = new Date(this.valueOf());
                    date.setDate(date.getDate() + days);
                    return date;
                }
                var conversation;
                var meeting = client.conversationsManager.createMeeting();

                /* meeting.subject('Planning meeting');
var dateData = new Date();
console.log(dateData.addDays(1));
meeting.expirationTime(dateData.addDays(1));
meeting.joinUrl.get().then(uri => {
    console.log('-Uri-joinUrl-');
    console.log(uri);
});

meeting.onlineMeetingUri.get().then(uri => {
    console.log('-Uri--');
    console.log(uri);
    conversation = client.conversationsManager.getConversationByUri(uri);
});
console.log('conversation');
console.log(conversation);*/
                var inputDate = document.getElementById('meetingtime').value;
                var day;
                if(inputDate != null){
                    day = inputDate.split('-')[2];
                }
                var dateData = new Date();
                meeting.subject(document.getElementById('subject').value);
                meeting.expirationTime(dateData.addDays(day));
                var meetingID = meeting.onlineMeetingUri.get().then(function(uri){
                    var newDiv = document.createElement('div');

document.getElementById("conversationWindow").appendChild(newDiv);
                    var renderConvDfd = apiManager.renderConversation(newDiv,{

```

```

        conversationId:uri,
        modalities:['Chat']
    }).then(function(conversation){
        console.log('--userName--->>>'+userName);
        conversation.participants.add(userName);
    });
    //monitor(renderConvDfd, 'render meeting');
});
//monitor(renderConvDfd,'meeting scheduled compleated');
/* meeting.onlineMeetingUri.get().then(uri => {
    conversation = client.conversationsManager.getConversationByUri(uri);
});*/
    }else{
        alert('Pleas Enter the Subject');
    }
    }else{
        alert('Pleas Enter the valid expiration date');
    }
    }else{
        alert('Pleas Enter the valid User Name');
    }
}catch(ex){
    console.log('----Exception-->>>'+ex);
}
};
var newDate = new Date();
var curretDate;
curretDate = newDate.getFullYear();
if((newDate.getMonth().toString()).length == 1){
    curretDate += '-0'+newDate.getMonth();
}else{
    curretDate += '-' +newDate.getMonth();
}
if((newDate.getDay().toString()).length == 1){
    curretDate += '-0'+newDate.getDay();
}else{
    curretDate += '-' +newDate.getDay();
}
/* if((newDate.getHours().toString()).length == 1){
    curretDate += 'T0'+newDate.getHours();
}else{
    curretDate += 'T'+newDate.getHours();
}
}

```



```
        if((newDate.getMinutes().toString()).length == 1){
            curretDate += ':0'+(newDate.getMinutes());
        }else{
            curretDate += ':'+(newDate.getMinutes());
        }*/
        document.getElementById("meetingtime").min = curretDate;
        document.getElementById("meetingtime").value = curretDate;
    }());
</script>
</body>
</html>
</apex:page>
```