

ORACLE®

Oracle Digital Assistant

The Complete Training

Oracle Web SDK



Image courtesy of pixabay.com

Copyright © 2018, Oracle and/or its affiliates. All rights reserved. |

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

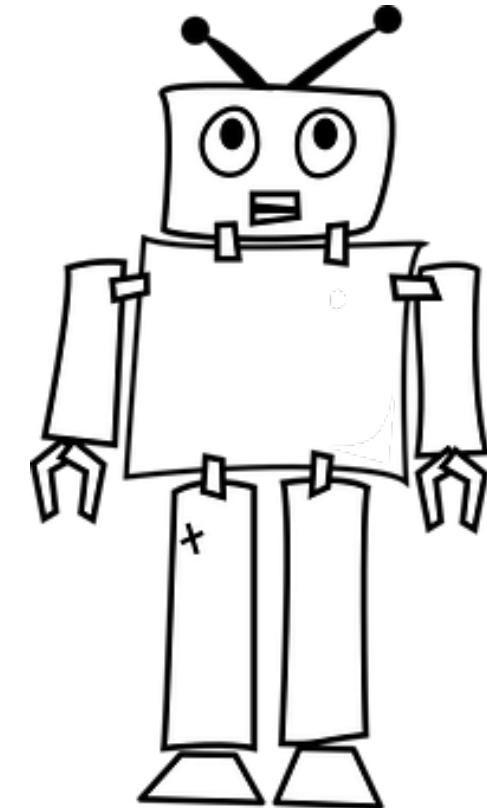
Topic agenda

- 1 ➤ Introduction to Oracle Web SDK
- 2 ➤ Web channel configuration
- 3 ➤ Web SDK channel connection types
- 4 ➤ Demo and Customizing Web SDK

Topic agenda

- 1 ➤ Introduction to Oracle Web SDK
- 2 ➤ Web channel configuration
- 3 ➤ Web SDK channel connection types
- 4 ➤ Demo and Customizing Web SDK

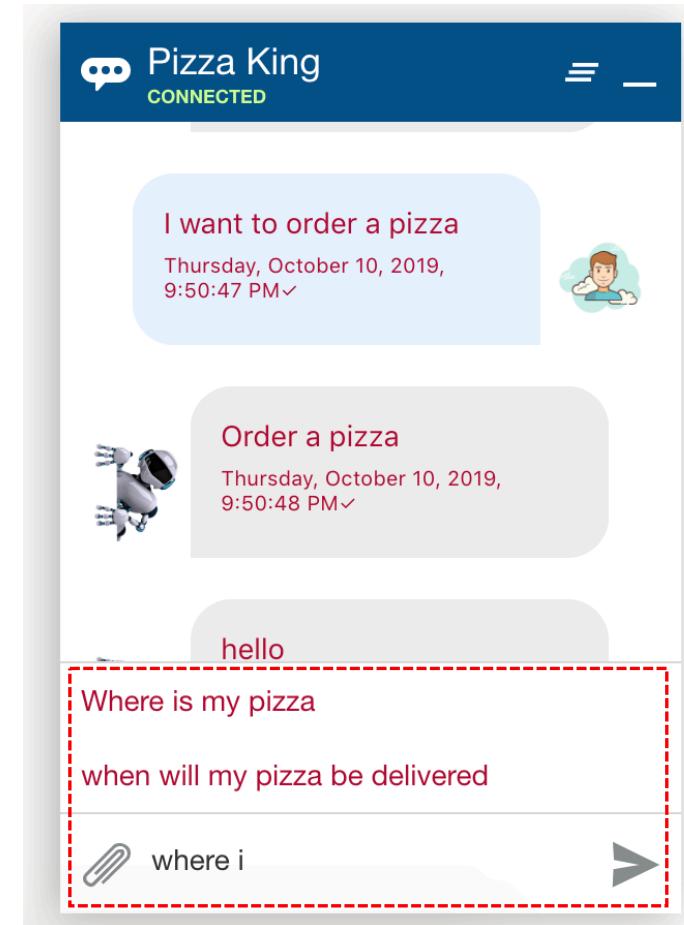
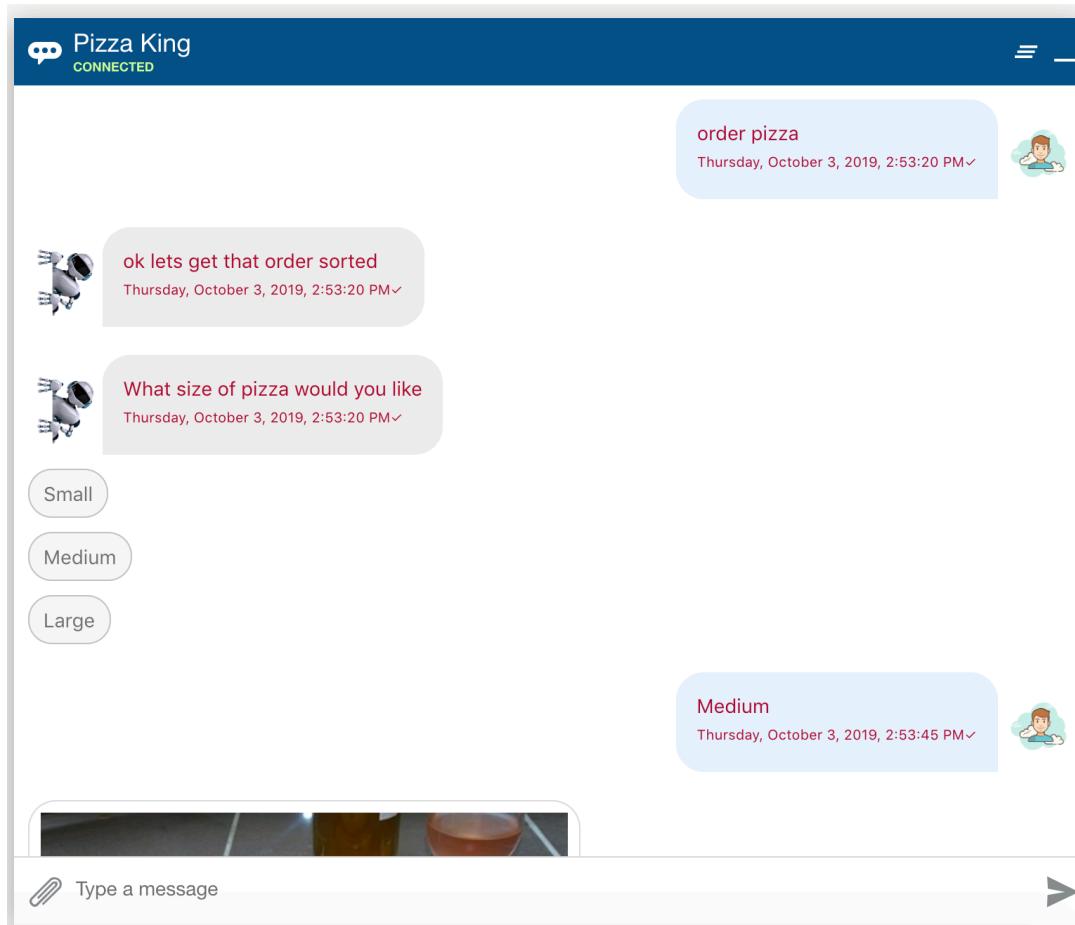
Oracle offers a new Web SDK for integrating Oracle Digital Assistant with web applications and web sites



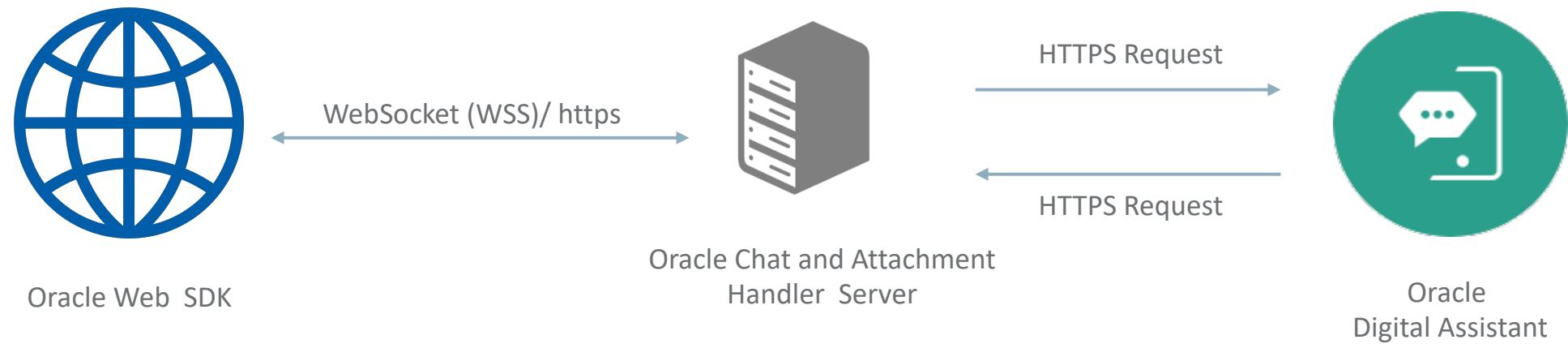
Some of the new features

- Ease of installation
- Support for JWT client authentication
- Configurable components support e.g. timestamp display, loading chat bubble size, font size, chat widget size, padding in message bubbles
- Enable autocomplete feature
- File/document attachment capabilities
- Embedded mode support and much more!!

Sample user interfaces

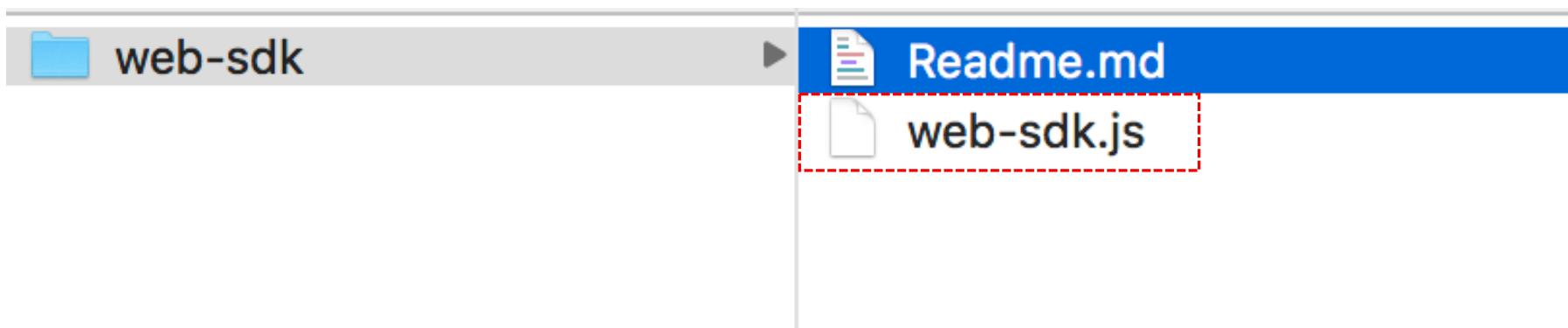


How the Oracle web SDK works



Oracle Web SDK download

- Download SDK
 - From downloads section link
 - Extract the SDK
 - Simple steps to add the client SDK to your app



Topic agenda

- 1 ➤ Introduction to Oracle Web SDK
- 2 ➤ Web channel configuration
- 3 ➤ Web SDK channel connection types
- 4 ➤ Demo and Customizing Web SDK

Create Oracle web channel

Development

Home

Development

Skills

Digital Assistants

Channels

Store

ORACLE® Digital Assistant

Channels

Users Agent Integrations Applications System

+ Channel

No items to display.

Create Channel

* Name: RDH_Oracle_Web

Description: Adding ODA in web page

Channel Type: Oracle Web

* Allowed Domains: Slack, Microsoft Teams, Cortana, Skype For Business, Oracle iOS, Oracle Android, Oracle Web

* Max. Token Expiration (Minutes)

Session Expiration (minutes): Oracle Web

Create

Web channel configuration

Route To CrcPizzaBot DRAFT • 1.1 ▾

Channel Enabled

Reset Sessions

* Name RDH_Oracle_Web

Description *Optional short description for this channel*

Channel Type Oracle Web

? * Allowed Domains *

Secret Key qGg...ZBQGu Reset

Channel Id 890...49afc808

Client Authentication Enabled

Session Expiration (minutes) 60 ▾ Default

List of allowed domains to connect to the Chat Server.
Required, if client authentication is disabled.

Secrets Key used by customer's backend service to
generate and sign JWT tokens for client authentication.
Required if client authentication is enabled.

Used to determine whether SDK is connecting to
client authentication enabled channel.

Web SDK Installation

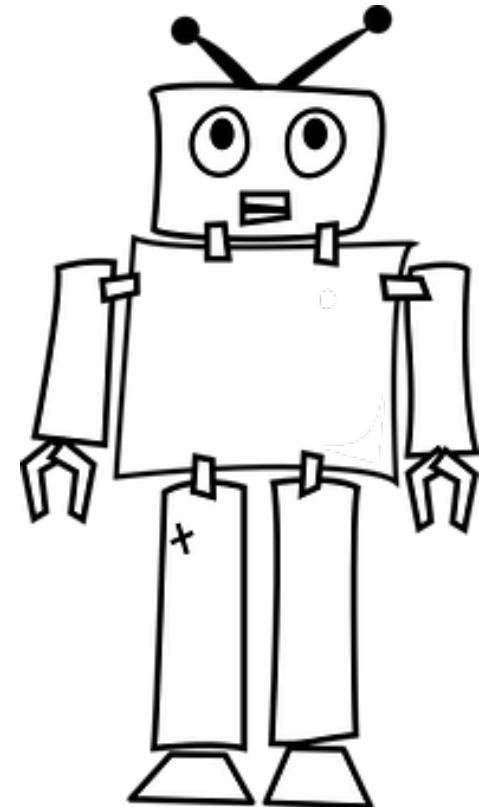
- First, save web-sdk.js in your project directory.
- Add the following code towards the end of the head section on your page and replace **<WebSDK URL>** with location of web-sdk.js.
- Replace **<WebSDK namespace>** with meaningful name, e.g. “Bots”. It will be used to invoke the SDK's public APIs, e.g. Bots.<API>()

```
1 <script>
2     var chatWidgetWebSettings = {
3         URI: '<Chat server URI>',
4         channelId: '<Channel ID>',
5         userId: '<User ID>'
6     };
7     !function(globalObj, doc, library, name) {
8         function initiateSDK() {
9             setTimeout(function() {
10                 globalObj[name] = Object.assign(WebSDK);
11                 globalObj[name] = new globalObj[name](chatWidgetWebSettings);
12                 globalObj[name].connect(); // Connect to chat server
13             }, 2000);
14         }
15         try {
16             var scripts = doc.getElementsByTagName('script')[0],
17                 lib = doc.createElement('script');
18             lib.async = true;
19             lib.src = library;
20             lib.onReady = initiateSDK();
21             scripts.parentNode.insertBefore(lib, scripts);
22         } catch (e) {
23             console.error('Could not load the chat widget');
24         }
25     }(window, document, '<WebSDK URL>', '<WebSDK namespace>');
26 </script>
```

Topic agenda

- 1 ➤ Introduction to Oracle Web SDK
- 2 ➤ Web channel configuration
- 3 ➤ Web SDK channel connection types
- 4 ➤ Demo and Customizing Web SDK

Oracle web channel can be configured to support **client authentication** using **JSON Web Tokens (JWT tokens)**



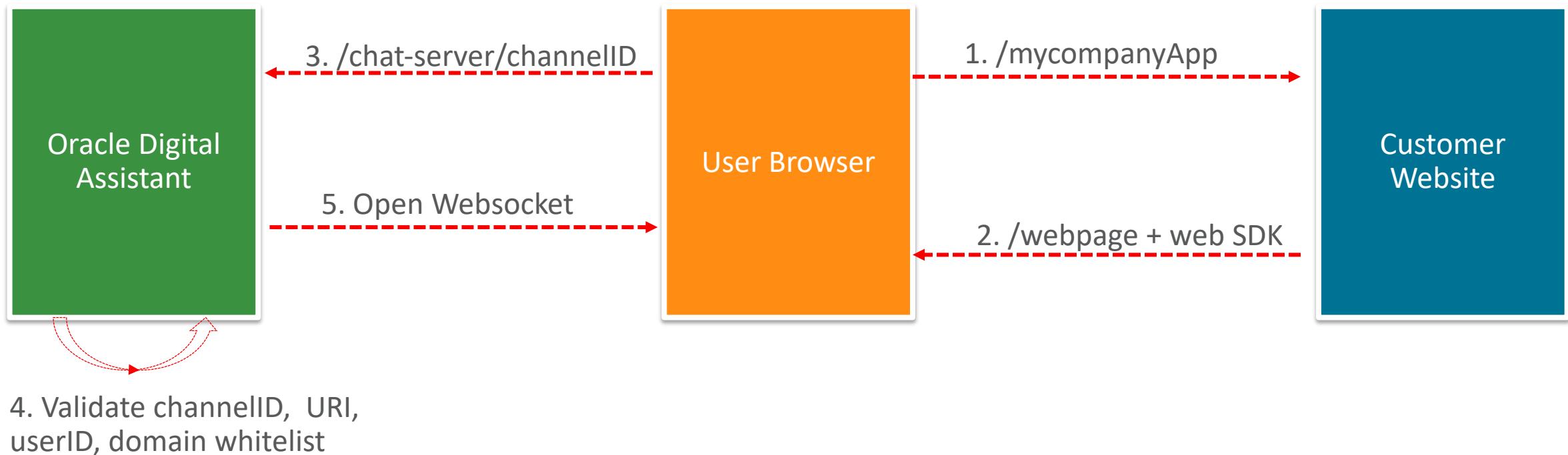
Mode - Client Authentication Disabled

Parameters required:

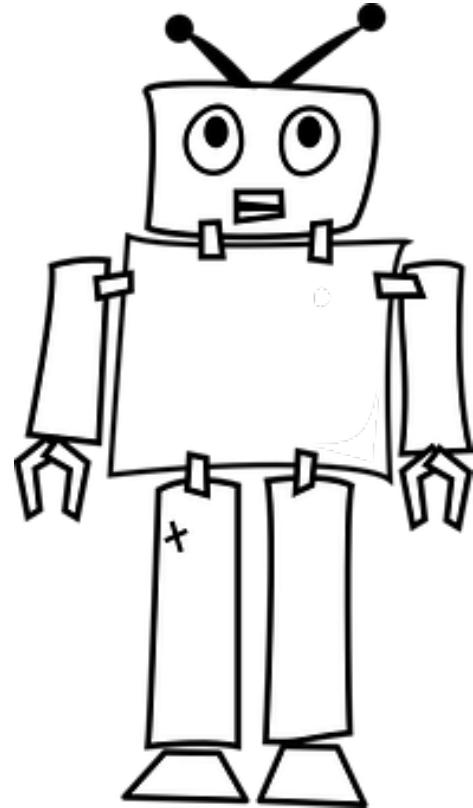
- **URI**: ODA instance ID (required)
- **channelId**: unique identifier for the channel. (required)
- **userId**: used in bots to maintain the conversation session (optional)

```
1  <!doctype html>
2  <html lang="en">
3  <head>
4      <meta charset="utf-8">
5      <title>The ODA Web SDK</title>
6  <script>
7      var chatWidgetSettings = {
8          URI: 'idcs-oda-XXXX-d0.data.digitalassistant.oci.oc-test.com',
9          channelId: '4537071b-XXXX-41e6-81f8-bb4061fbe36b',
10         userId: 'Neo'
11     };
12     !function (globalObj, doc, library, name) {
13         function initiateSDK() {
14             setTimeout(function () {
15                 globalObj[name] = Object.assign(WebSDK);
16                 globalObj[name] = new globalObj[name](chatWidgetSettings);
17                 globalObj[name].connect();
18             }, 2000);
19         }
20         try {
21             var scripts = doc.getElementsByTagName("script")[0],
22                 lib = doc.createElement("script");
23             lib.async = true;
24             lib.src = library;
25             lib.onReady = initiateSDK();
26             scripts.parentNode.insertBefore(lib, scripts);
27         } catch (e) {
28             console.error("Could not load the chat widget");
29         }
30     }(window, document, "web-sdk.js", "Bots");
31 </script>
32 </head>
33 <body>
34     <h3 align="center">The ODA Web SDK</h3>
35 </body>
36 </html>
```

Flow - Client Authentication Disabled



The option to **disable the client authentication** requirement should be used only for web pages and applications that by itself restrict access to authorized users only.



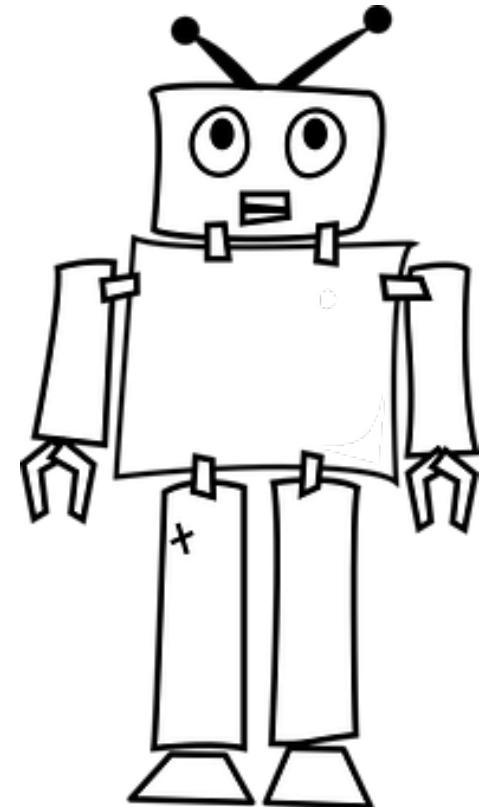
Mode - Client Authentication Enabled

Parameters required:

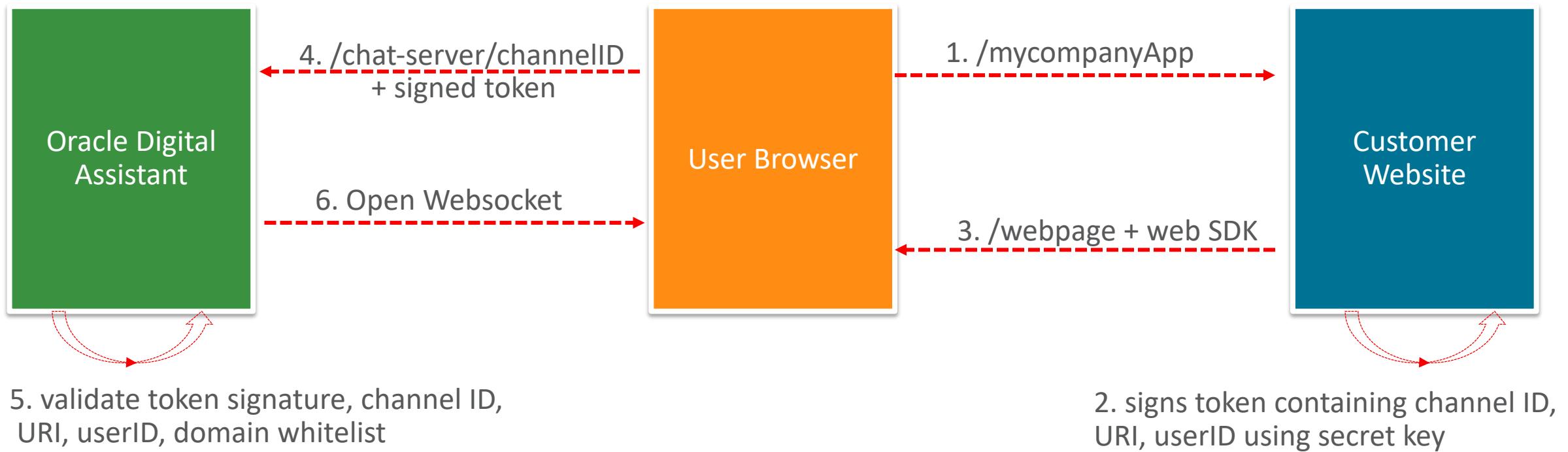
- **URI**: ODA instance ID (required)
- a **signed JWT token** must be set in the Authorization header containing all the required parameters
 - **channelId**
 - **URI**
 - **userId** as token claims.

```
4 ▼   var chatWidgetWebSettings = {  
5     URI: '<Chat server URL>'  
6   };  
7 ▼   var generateToken = function() {  
8     return new Promise((resolve) => {  
9       fetch('https://yourbackend.com/endpointToGenerateJWTToken').then((token) => {  
10         resolve(token);  
11       })  
12     });  
13   };  
14   !function(globalObj, doc, library, name) {  
15     function initiateSDK() {  
16       setTimeout(function() {  
17         globalObj[name] = Object.assign(WebSDK);  
18         globalObj[name] = new globalObj[name](chatWidgetWebSettings, generateToken);  
19         globalObj[name].connect();  
20       }, 2000);  
21     }  
22   };
```

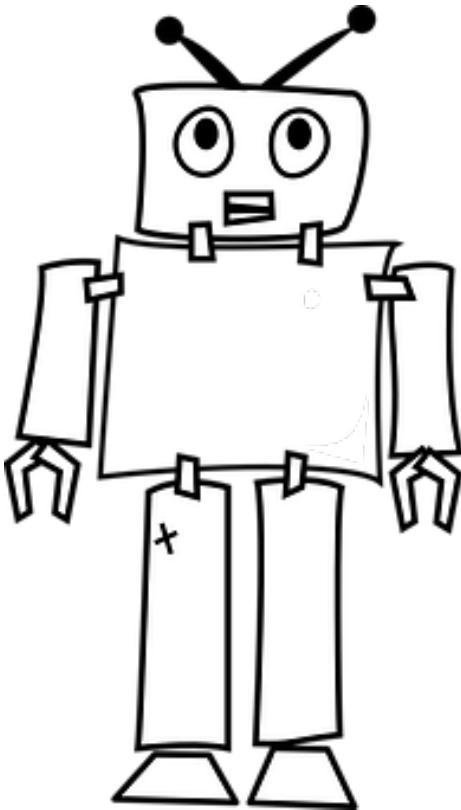
The JWT token **generation and signing**
must happen on the customer **backend server** and never on the client either web or mobile



Flow - Client Authentication Enabled



The client authentication enabled mechanism adds an extra layer of security over the connection, ensuring that unauthorised connections are not established

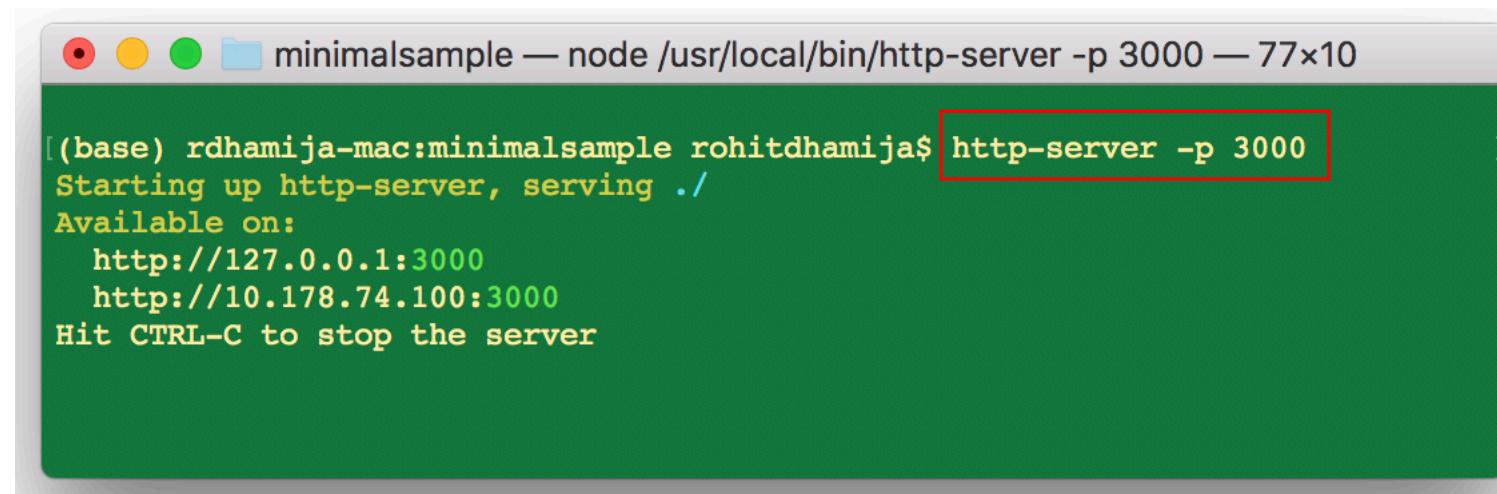


Topic agenda

- 1 ➤ Introduction to Oracle Web SDK
- 2 ➤ Web channel configuration
- 3 ➤ Web SDK channel connection types
- 4 ➤ Demo and Customizing Web SDK

Running locally

- Host the page with any web server tool you want!
- Some popular tool for local testing are JavaScript HTTP server, and Python web server

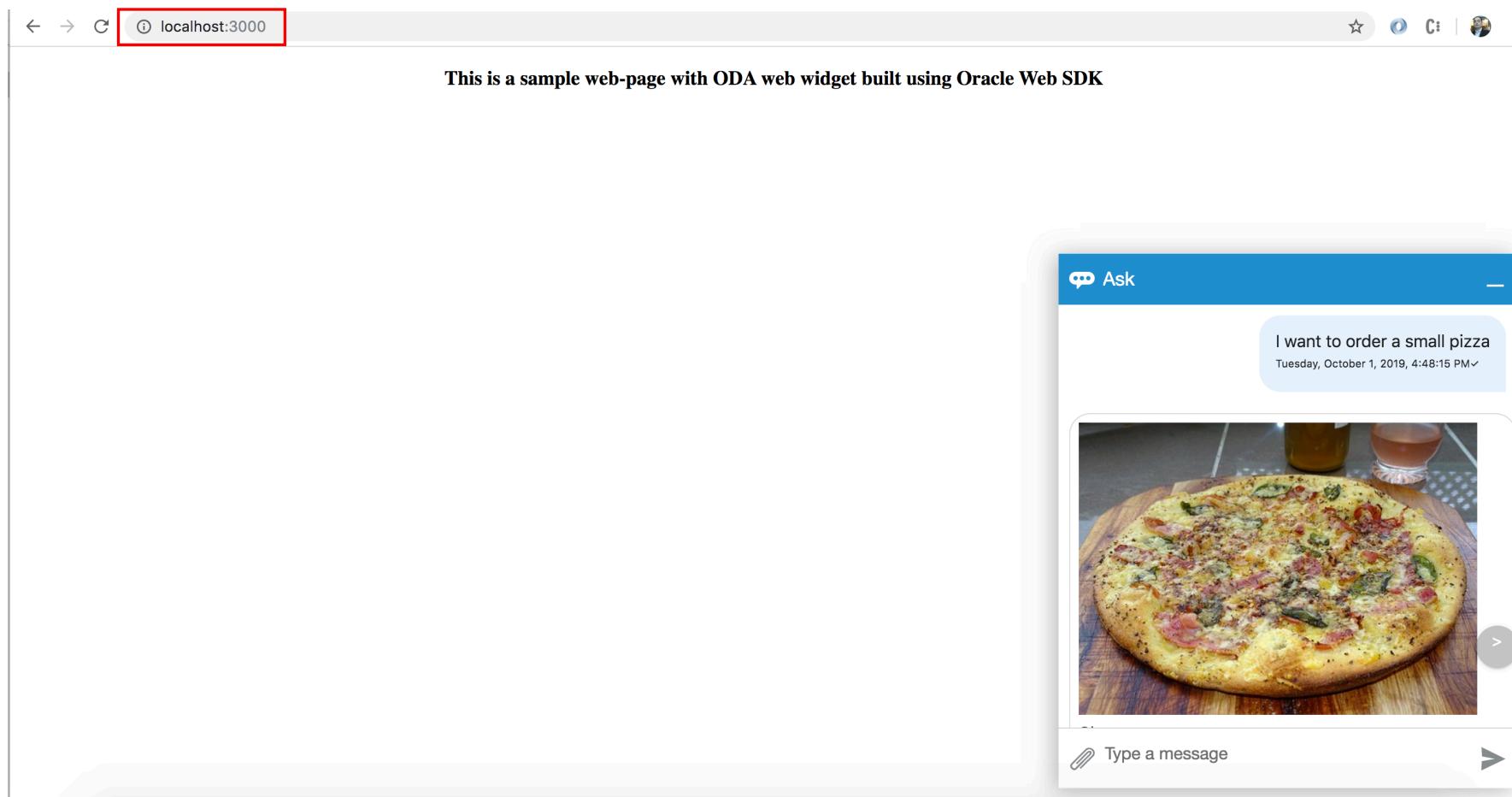


A screenshot of a macOS terminal window titled "minimalsample — node /usr/local/bin/http-server -p 3000 — 77x10". The window contains the following text:

```
(base) rdhamija-mac:minimalsample rohitdhamija$ http-server -p 3000
Starting up http-server, serving .
Available on:
  http://127.0.0.1:3000
  http://10.178.74.100:3000
Hit CTRL-C to stop the server
```

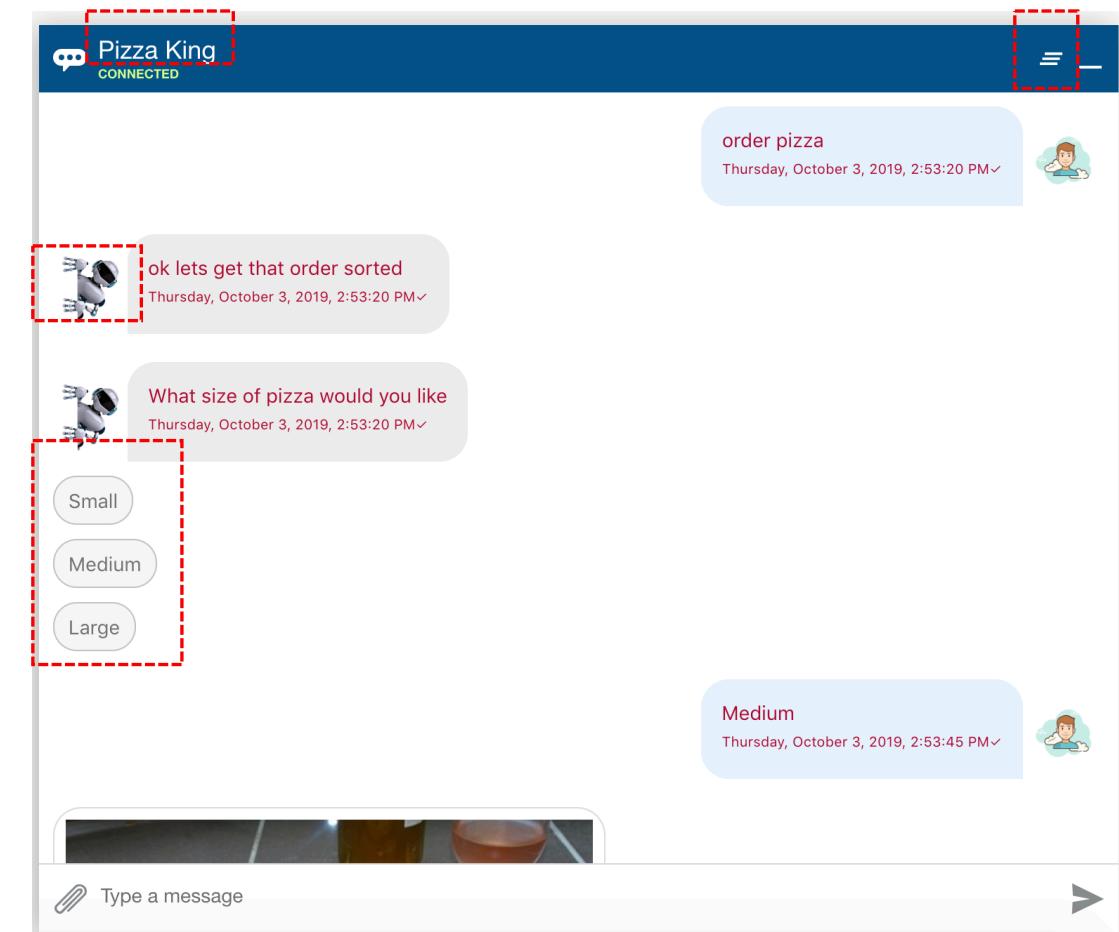
The command "http-server -p 3000" is highlighted with a red box.

Running sample web-app locally



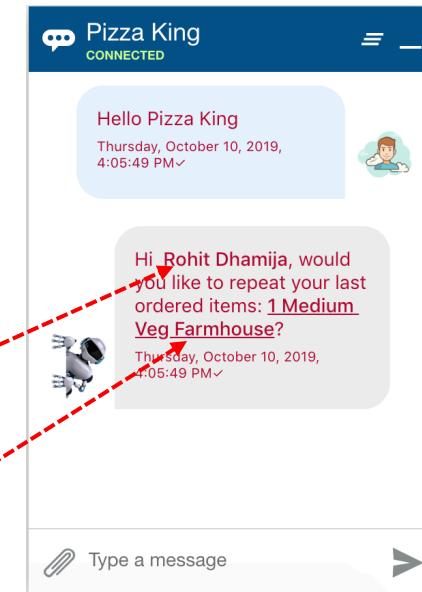
Customize web widget

```
8     var chatWidgetSettings = {  
9         URI: 'idcs-oda-48a59dc1d22c4fb88ce12887752f5e88-d0.data.digitalassistant.oci.oc-test.com',  
10        channelId: '4537071b-b4e4-4fa6-81f8-bb4061fbe36b',  
11        userId: 'Neo',  
12        font: '14px "Helvetica Neue", Helvetica, Arial, sans-serif',  
13        locale: 'en-us',  
14        enableClearMessage: true,  
15        showConnectionStatus: true,  
16        isDebugEnabled: true,  
17        height: '90vh',  
18        width: '60vw',  
19  
20        "i18n": {  
21            "en-us": {  
22                "chatTitle": "Pizza King"  
23            }  
24        },  
25        "colors": {  
26            "branding": "#01538A",  
27            "text": "#B20637"  
28        },  
29        botButtonIcon: 'images/Robot.png',  
30        botIcon: 'images/1\bot.png',  
31        personIcon: 'images/userIcon.png',  
32        displayActionsAsFills: true  
33    };
```



Updating and displaying user information

```
36 ▼         function initiateSDK() {
37 ▼             setTimeout(function () {
38                 globalObj[name] = Object.assign(WebSDK);
39                 globalObj[name] = new globalObj[name](chatWidgetSettings);
40                 globalObj[name].connect().then(() => {
41                     Bots.sendMessage('Hello Pizza King');
42                     Bots.updateUser({
43                         profile: {
44                             givenName: 'Rohit',
45                             surname: 'Dhamija',
46                             email: 'rohit.dhamija@oracle.com',
47                             properties: {
48                                 lastOrdredItems: "1 Medium Veg Farmhouse"
49                             }
50                         }
51                     });
52             }, (reason) => {
53                 console.log("Connection failed");
54                 console.log(reason);
55             });
56         }, 2000);
57     }
```

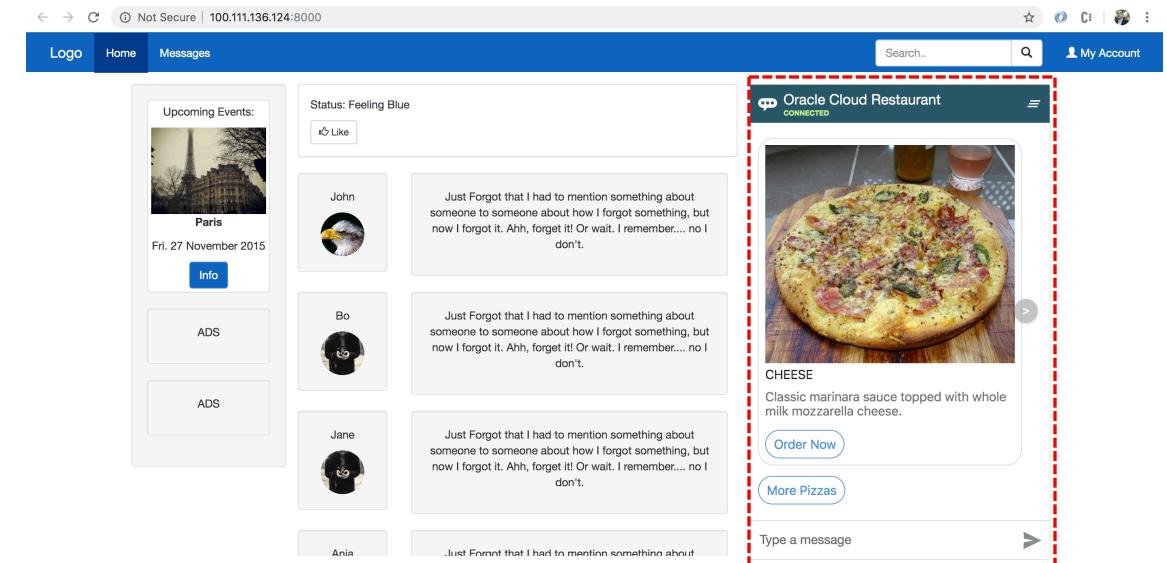


```
30     start:
31         component: "System.Output"
32         properties:
33             text: "Hi <b> ${profile.givenName} ${profile.surname}</b>, would you like to repeat your last ordered items:
34             <b><u>${profile.properties.value.lastOrdredItems}</u></b>?"
35             transitions:
36                 return: "done"
```

Embedded mode

```
chatWidgetSettings = {  
    URI: 'idcs-oda-XXX-d0.data.digiXX.oci.oc-test.com',  
    channelId: '25f5898b-XXXX-a91c-6faf8a8c01ca',  
    showConnectionStatus: true,  
    enableClearMessage: true,  
    width: '420px',  
    height: '650px',  
    enableTimestamp: false,  
    openChatOnLoad: true,  
    initUserHiddenMessage: 'Show Menu',  
    embedded: true,  
    targetElement: 'container-div',  
    enableAttachment: false,  
    displayActionsAsPills: true,  
    enableLocalConversationHistory: true,  
    colors: {  
        branding: '#2c5967'  
    },  
    i18n: {  
        'en-us': {  
            chatTitle: 'Oracle Cloud Restaurant'  
        }  
    }  
};
```

```
<div id="container-div" class="col-sm-3 well"  
    style="height: 600px; width: 380px; padding: 0; text-align: initial">  
</div>
```



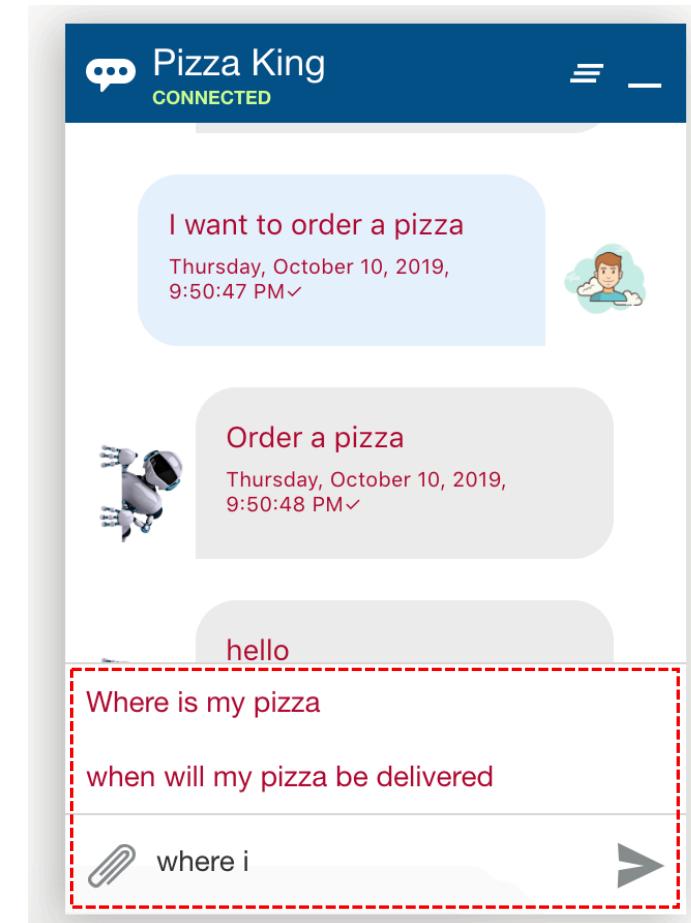
Enable auto complete

- Navigate to Intents
- Under Auto-Complete Suggestions section - Provide utterances to be used for auto complete suggestions.

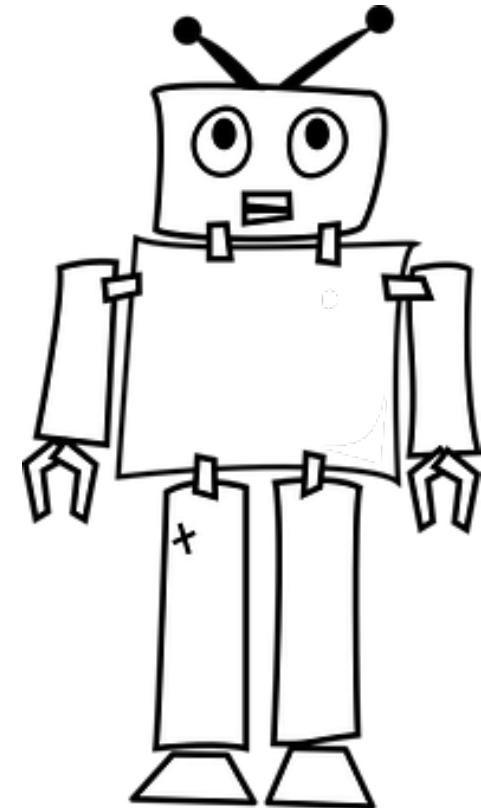
The screenshot shows the Oracle AI Platform interface for creating an intent. On the left, there's a sidebar with various icons for managing intents, entities, and other components. The main area is titled 'Intent' and has a 'More' dropdown. A 'Filter' input field is at the top. Below it, there's a 'Sort By' dropdown set to 'Created Ascending'. The list of intents includes 'CheckOrder' (selected), 'OrderPizza', and 'Welcome'. To the right of the list, there are sections for 'Description' and 'Examples'. The 'Examples' section contains a 'Filter' input, a text area for 'Enter your example utterances here.', and a list of five examples: 'I want to check my order status', 'I want to confirm my pizza order', 'When will my pizza be delivered', 'Where is my order', and 'Where is my pizza'. Below this is a pagination bar 'Page 1 of 1 (1-5 of 5 items)'. At the bottom of the intent list, there's a section titled 'Auto-Complete Suggestions' with a 'Filter' input and a text area for 'Enter your suggestions here'. This section also contains three suggestions: 'I want to check my order', 'when will my pizza be delivered', and 'Where is my pizza'. This entire 'Auto-Complete Suggestions' section is highlighted with a red dashed border. There's another pagination bar 'Page 1 of 1 (1-3 of 3 items)' at the bottom of this section. In the top right corner of the main area, there's a 'Try It Out!' button.

Enable auto complete

```
9 ▼     var chatWidgetSettings = {  
10        URI: 'idcs-oda-48a59dc1d22c4fb88ce12887752f5e88-d0.data.digitalassistant.oci.oc-test.com',  
11        channelId: 'ca6ed977-1593-4028-b27e-3702b0778745',  
12        userId: 'rdh',  
13        font: '14px "Helvetica Neue", Helvetica, Arial, sans-serif',  
14        locale: 'en-us',  
15        enableClearMessage: true,  
16        showConnectionStatus: true,  
17        isDebugMode: true,  
18        height: '60vh',  
19        width: '20vw',  
20      };  
21      enableAutocomplete: true,  
22    };  
23    "i18n": {  
24      "en-us": {  
25        "chatTitle": "Pizza King"  
26      }  
27    },  
28    "colors": {  
29      "branding": "#01538A",  
30      "text": "#B20637"  
31    },  
32    botButtonIcon: 'images/Robot.png',  
33    botIcon: 'images/Robot.png',  
34    personIcon: 'images/userIcon.png',  
35    displayActionsAsPills: true  
36  };
```



User has to enter a minimum of **three characters** in order to get the autocomplete suggestions



Browser Support

- The SDK is supported on following browsers on Mac, Windows, and Linux:
 - Firefox
 - Chrome
 - Safari
 - Edge
- For mobile platforms the following browsers are supported:
 - Android browser
 - iOS Safari

* *Internet explorer is not officially supported*

Integrated Cloud Applications & Platform Services

ORACLE®