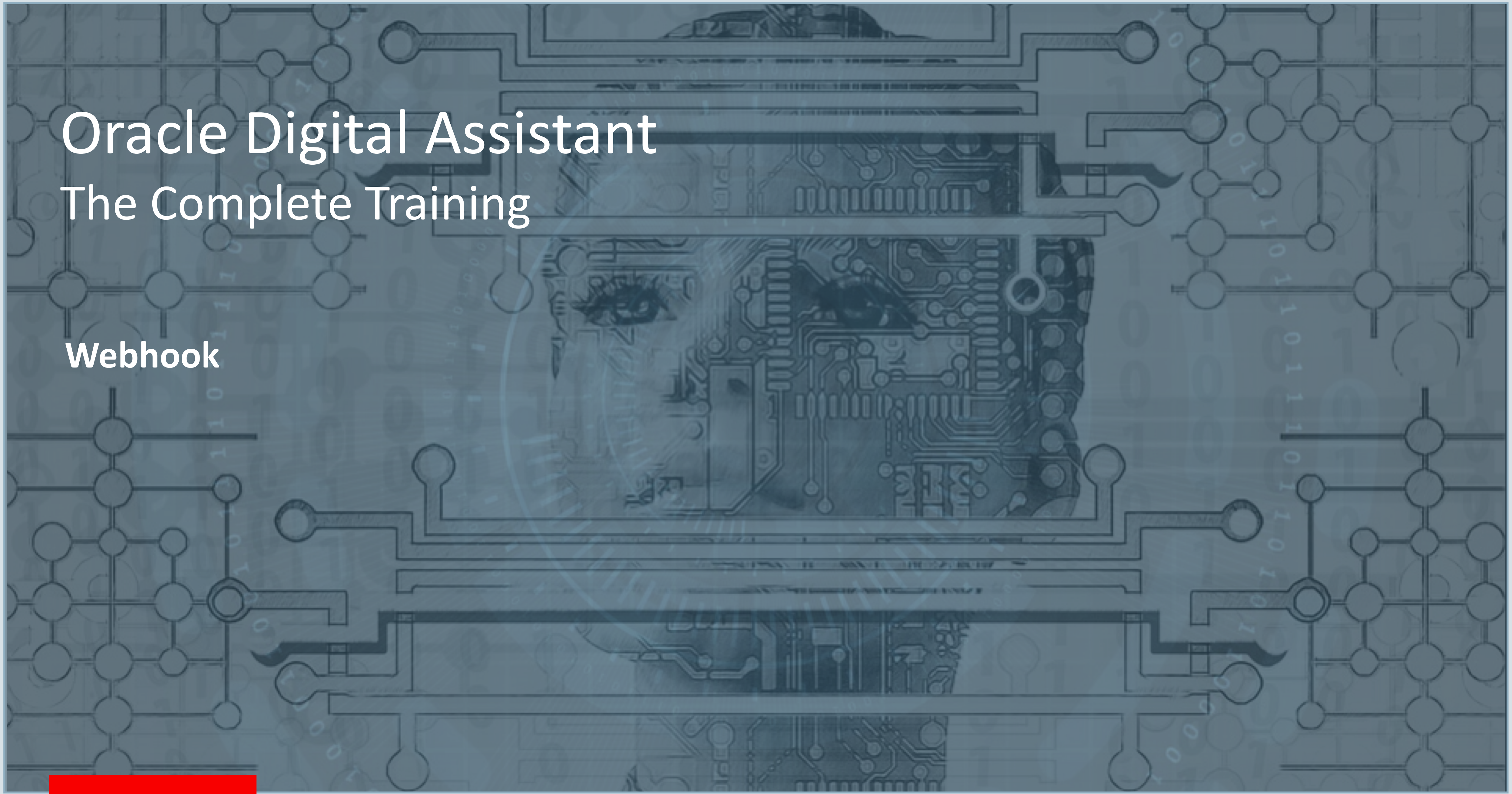


Oracle Digital Assistant

The Complete Training

Webhook

ORACLE

Image courtesy of pixabay.com

Safe Harbor Statement

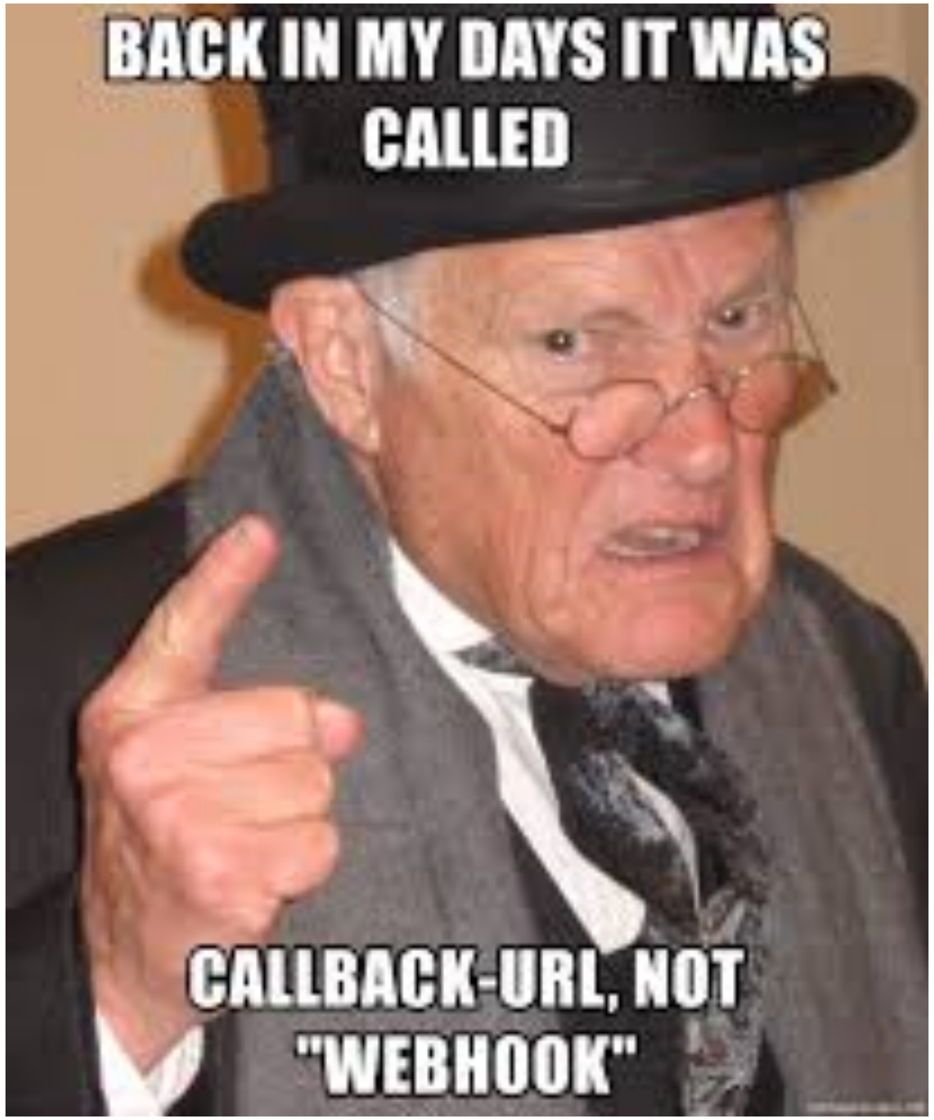
Topic agenda

1. Overview
2. Creating webhook clients with Node.js
3. Creating a webhook channel

Topic agenda

About webhooks and their use with chatbots

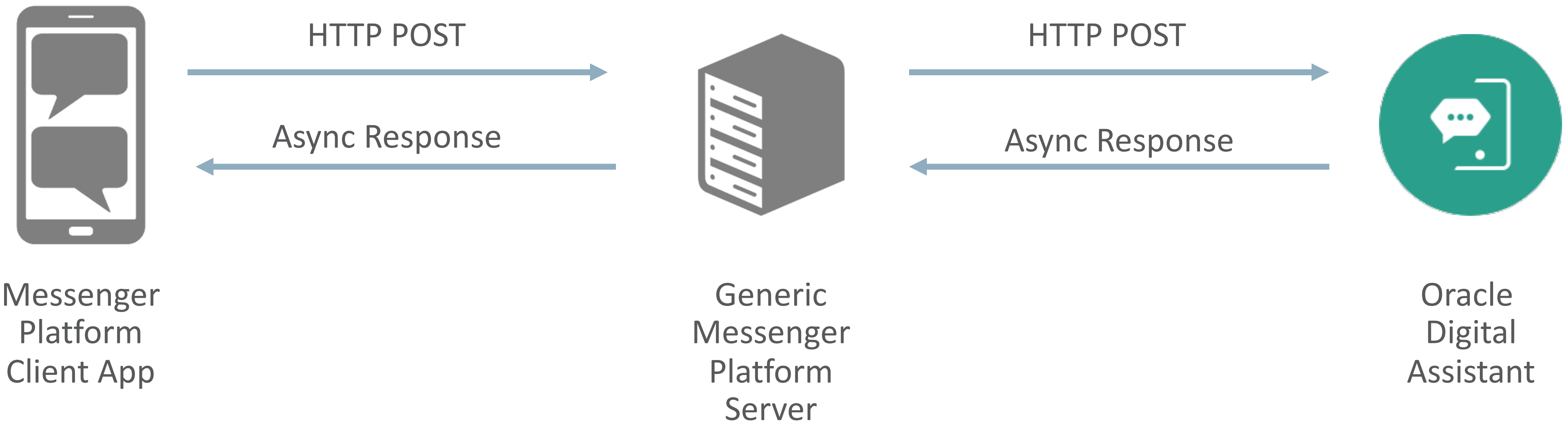
- Programming interfaces implemented by a system on the web
  - Allows other programs to subscribe and receive automated notification
- Accessing chatbots using Webhooks
  - Users access chatbots through a messenger
  - Webhook are "client adapters" that dispatch and manage conversation between messenger channels and chatbots
- Webhooks and Oracle Digital Assistant
  - You use webhooks to access Oracle Digital Assistant for messenger channels that there is no native channel support provided for



ORACLE

*https://memegenerator.net/

Generic HTTP webhook channel



Generic HTTP webhook support

- Bot channel publishes an HTTP Endpoint to receive messages
- You define a response HTTP Endpoint
- Bot will send response messages back to your server

- To verify messages Oracle Digital Assistant generates and uses a secret key
- Caller must supply
  - X-Hub-Signature HTTP header
  - Set to SHA256 signature of payload
  - Secret key used as SHA key
- Uses same mechanism on response
  - Optional for caller to verify payload

ORACLE

Topic agenda

Building steps for Oracle Digital Assistant

- Download Oracle Bots Node.js SDK
- Create a webhook client project
- Implement webhook client code for messages received from messenger
- Create a webhook channel in Oracle Digital Assistant
  - Associate channel with digital assistant or skill
- Reference webhook channel in webhook client

ORACLE

The **Oracle Bots Node.js SDK** makes it easy to build webhook clients for Oracle Digital Assistant.

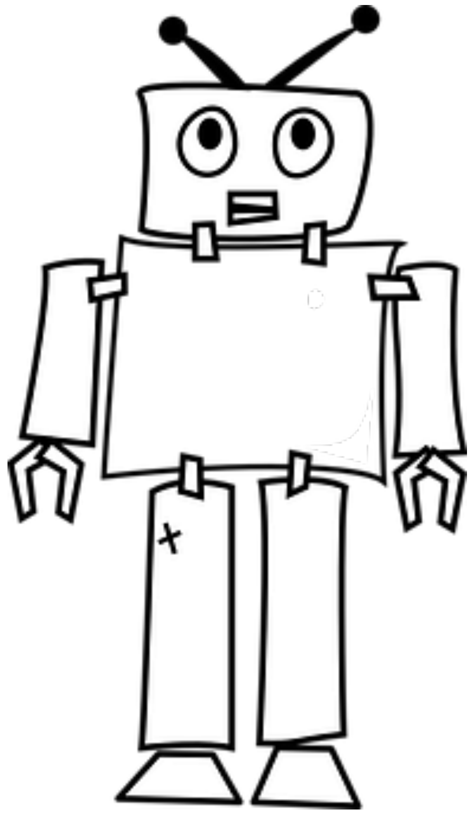


Image courtesy of pixabay.com

Oracle Bots Node.js SDK

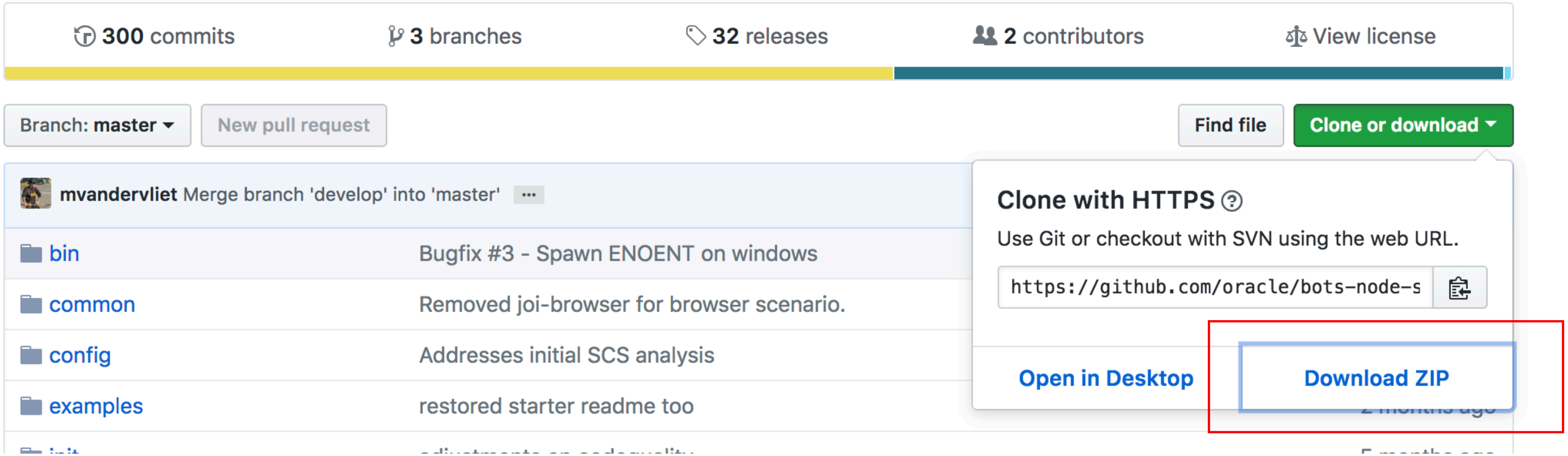
https://github.com/oracle/bots-node-sdk

- Bots Node.js SDK functionality
  - Custom component development
  - Webhook client development
- Webhook development support
  - Assists in building webhook clients that dispatch between messengers and bots
    - E.g. Alexa integration
  - Generates the SHA256 signature and sets X-Hub-Signature header

Getting started with your webhook development 1 of 4

Download Ortacle Bots Node.js SDK

Oracle Bots Node.js SDK https://oracle.github.io/bots-node-sdk/



mvandervliet Merge branch 'develop' into 'master'

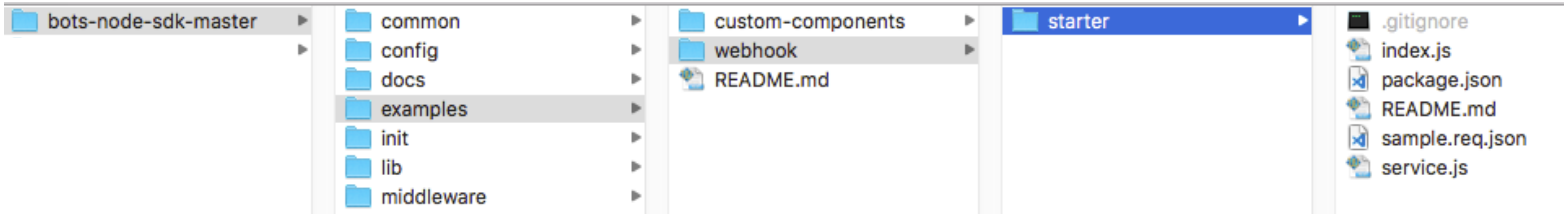
bin	Bugfix #3 - Spawn ENOENT on windows
common	Removed joi-browser for browser scenario.
config	Addresses initial SCS analysis
examples	restored starter readme too

Use Git or checkout with SVN using the web URL.

```
https://github.com/oracle/bots-node-s
```

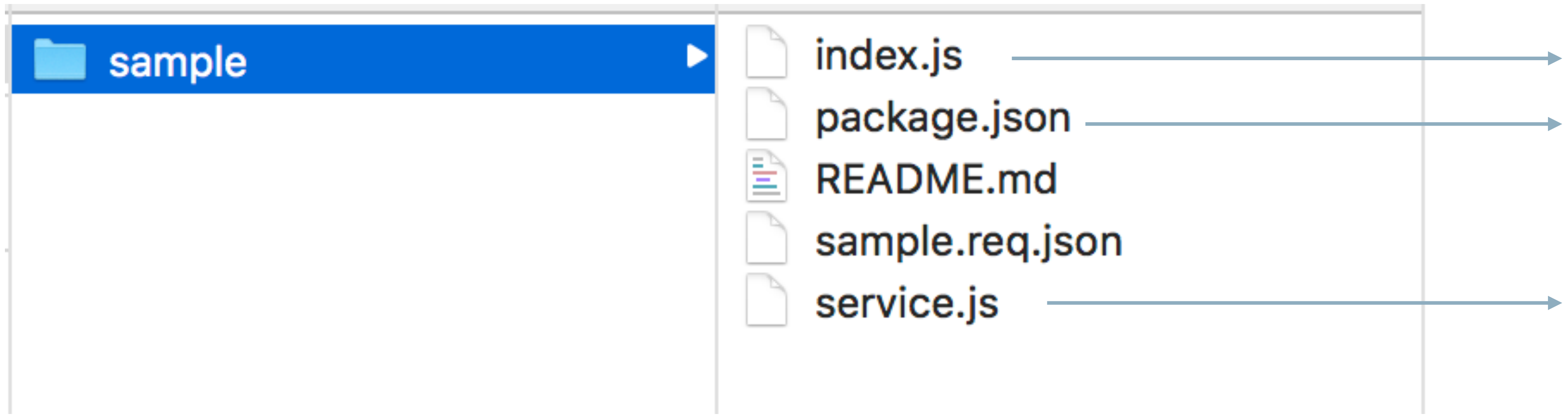
Getting started with your webhook development 2 of 4

Extract ZIP and navigate to bots-node-sdk-master/examples/webhook/starter folder



Getting started with your webhook development 3 of 4

Copy content of started folder to your custom project folder



index.js → Node Express server

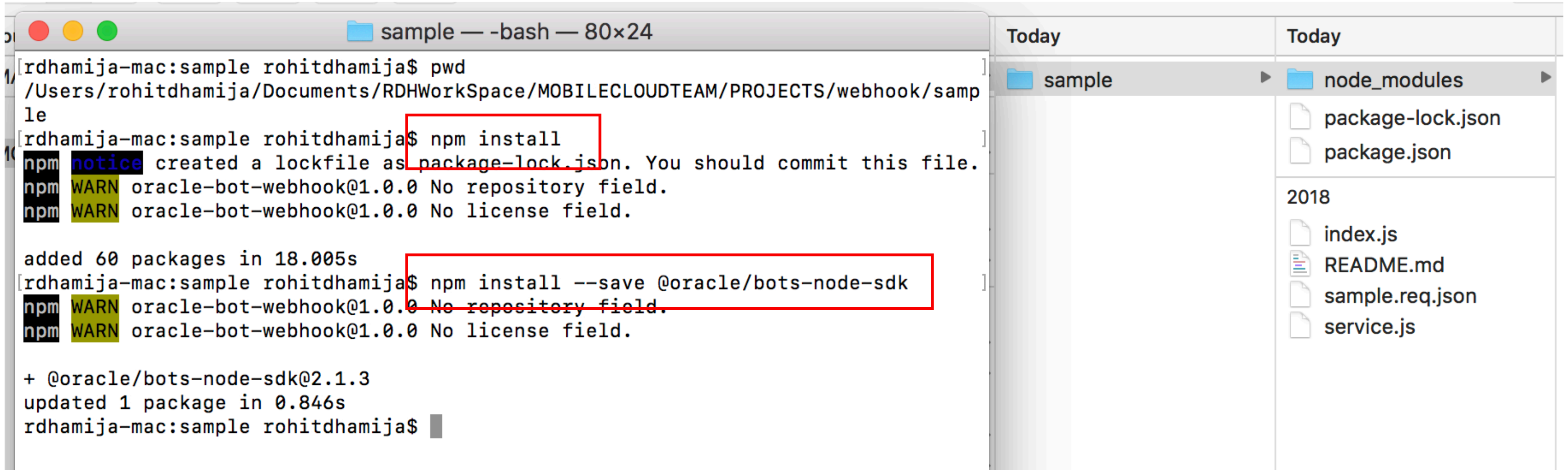
package.json → Node dependencies

README.md

sample.req.json

service.js → Webhook client

Getting started with your webhook development 4 of 4



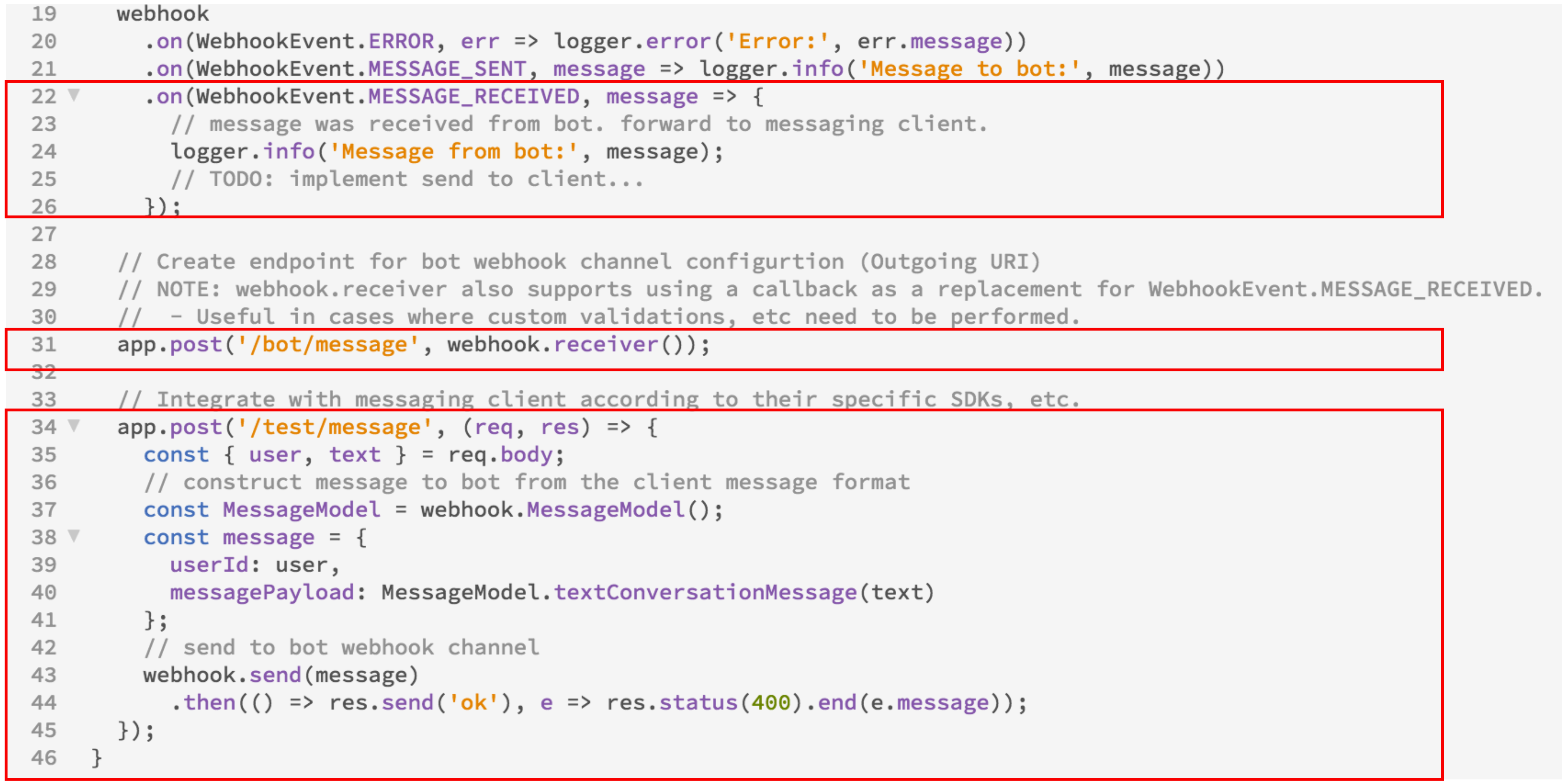
```
rdhamija-mac:sample rohitdhamija$ pwd
/Users/rohitdhamija/Documents/RDHWorkSpace/MOBILECLOUDTEAM/PROJECTS/webhook/samp
le
rdhamija-mac:sample rohitdhamija$ npm install
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN oracle-bot-webhook@1.0.0 No repository field.
npm WARN oracle-bot-webhook@1.0.0 No license field.

added 60 packages in 18.005s
rdhamija-mac:sample rohitdhamija$ npm install --save @oracle/bots-node-sdk
npm WARN oracle-bot-webhook@1.0.0 No repository field.
npm WARN oracle-bot-webhook@1.0.0 No license field.

+ @oracle/bots-node-sdk@2.1.3
updated 1 package in 0.846s
rdhamija-mac:sample rohitdhamija$
```

- Install project
  - npm install
  - Npm install –save @oracle/bots-node-sdk

Exploring the 'service.js' webhook client



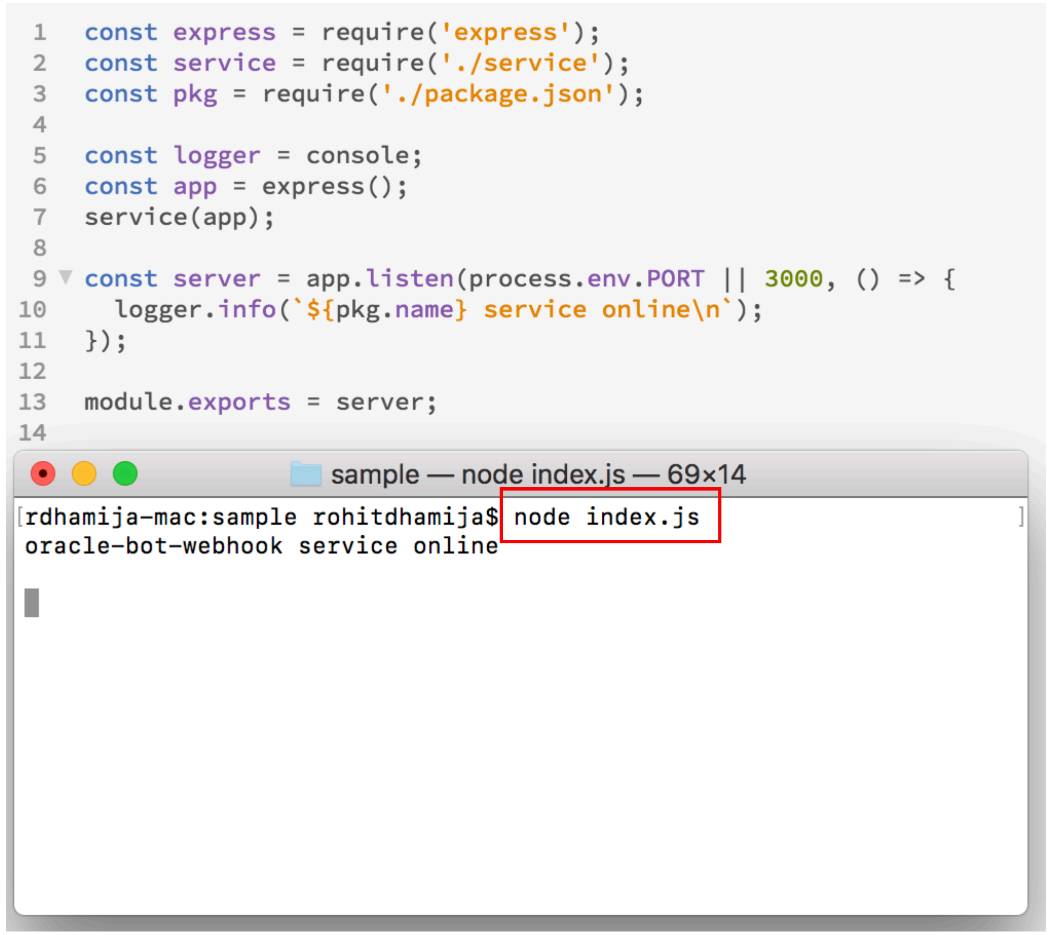
```
  webhook
    .on(WebhookEvent.ERROR, err => logger.error('Error:', err.message))
    .on(WebhookEvent.MESSAGE_SENT, message => logger.info('Message to bot:', message))
    .on(WebhookEvent.MESSAGE_RECEIVED, message => {
      // message was received from bot. forward to messaging client.
      logger.info('Message from bot:', message);
      // TODO: implement send to client...
    });

  // Create endpoint for bot webhook channel configurtion (Outgoing URI)
  // NOTE: webhook.receiver also supports using a callback as a replacement for WebhookEvent.MESSAGE_RECEIVED.
  //  - Useful in cases where custom validations, etc need to be performed.
  app.post('/bot/message', webhook.receiver());

  // Integrate with messaging client according to their specific SDKs, etc.
  app.post('/test/message', (req, res) => {
    const { user, text } = req.body;
    // construct message to bot from the client message format
    const MessageModel = webhook.MessageModel();
    const message = {
      userId: user,
      messagePayload: MessageModel.textConversationMessage(text)
    };
    // send to bot webhook channel
    webhook.send(message)
      .then(() => res.send('ok'), e => res.status(400).end(e.message));
  });
}
```

ORACLE

Running the project



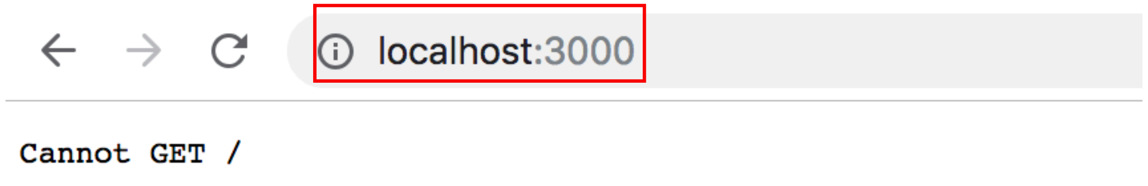
```
const express = require('express');
const service = require('./service');
const pkg = require('./package.json');

const logger = console;
const app = express();
service(app);

const server = app.listen(process.env.PORT || 3000, () => {
  logger.info(`${pkg.name} service online\n`);
});

module.exports = server;
```

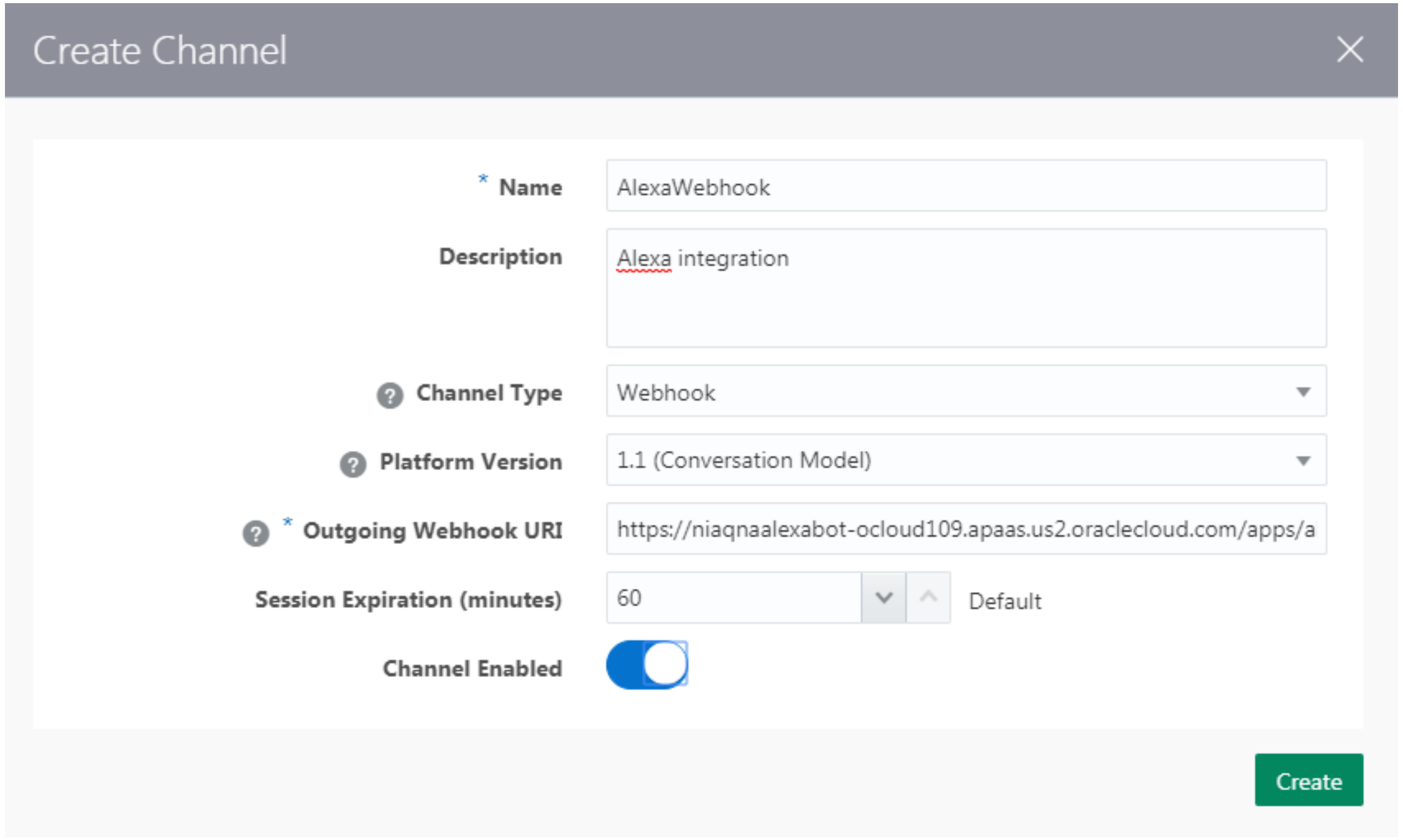
```
rdhamija-mac:sample rohitdhamija$ node index.js
oracle-bot-webhook service online
```



```
Cannot GET /
```

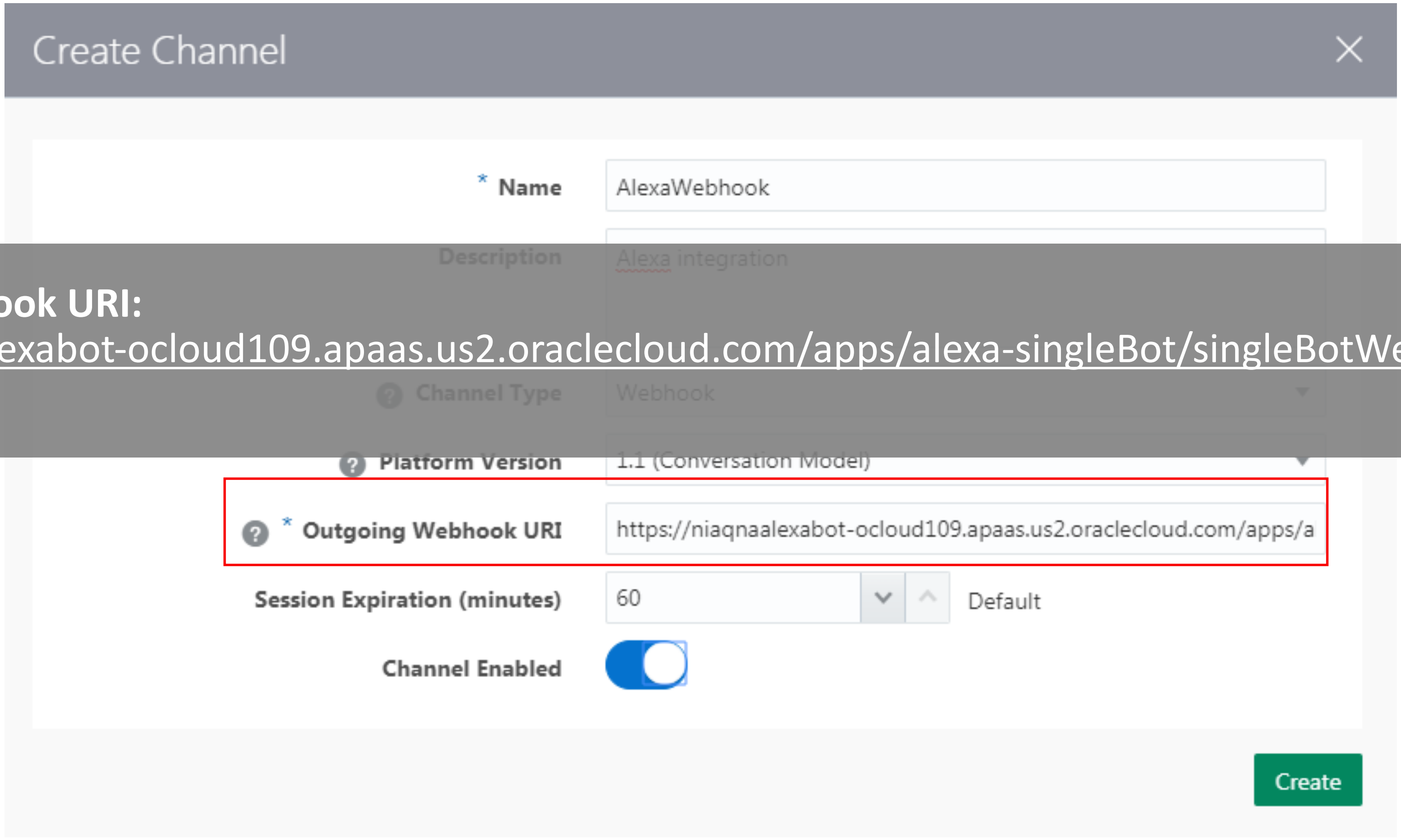
Topic Agenda

Oracle Digital Assistant – Creating a webhook channel



* Name	AlexaWebhook
Description	Alexa integration
Channel Type	Webhook
Platform Version	1.1 (Conversation Model)
* Outgoing Webhook URI	https://niaqnaalexabot-ocloud109.apaas.us2.oraclecloud.com/apps/a
Session Expiration (minutes)	60 Default
Channel Enabled	✓

Oracle Digital Assistant – Creating a webhook channel

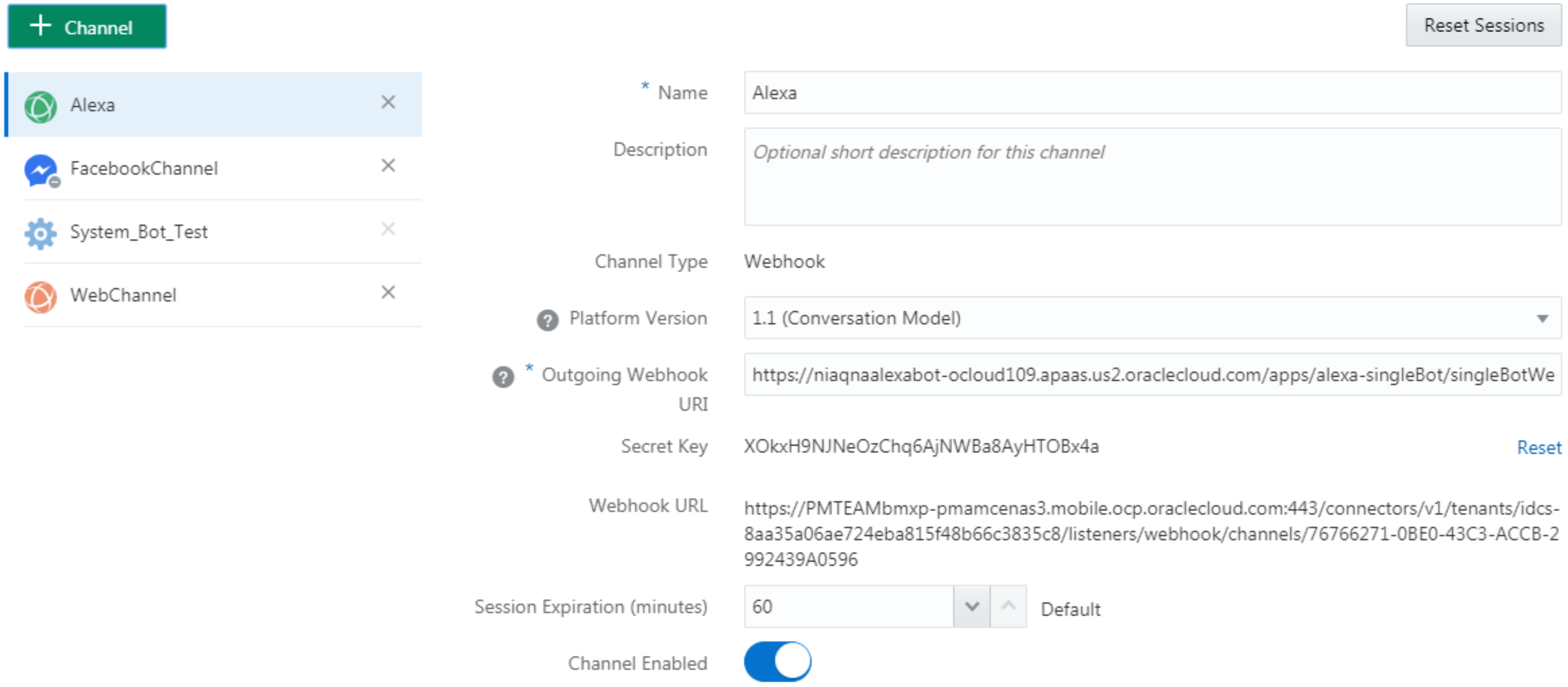


Create Channel

* Name	AlexaWebhook
Description	Alexa integration
Channel Type	Webhook
Platform Version	1.1 (Conversation Model)
* Outgoing Webhook URI	https://niaqnaalexabot-ocloud109.apaas.us2.oraclecloud.com/apps/a
Session Expiration (minutes)	60 Default
Channel Enabled	

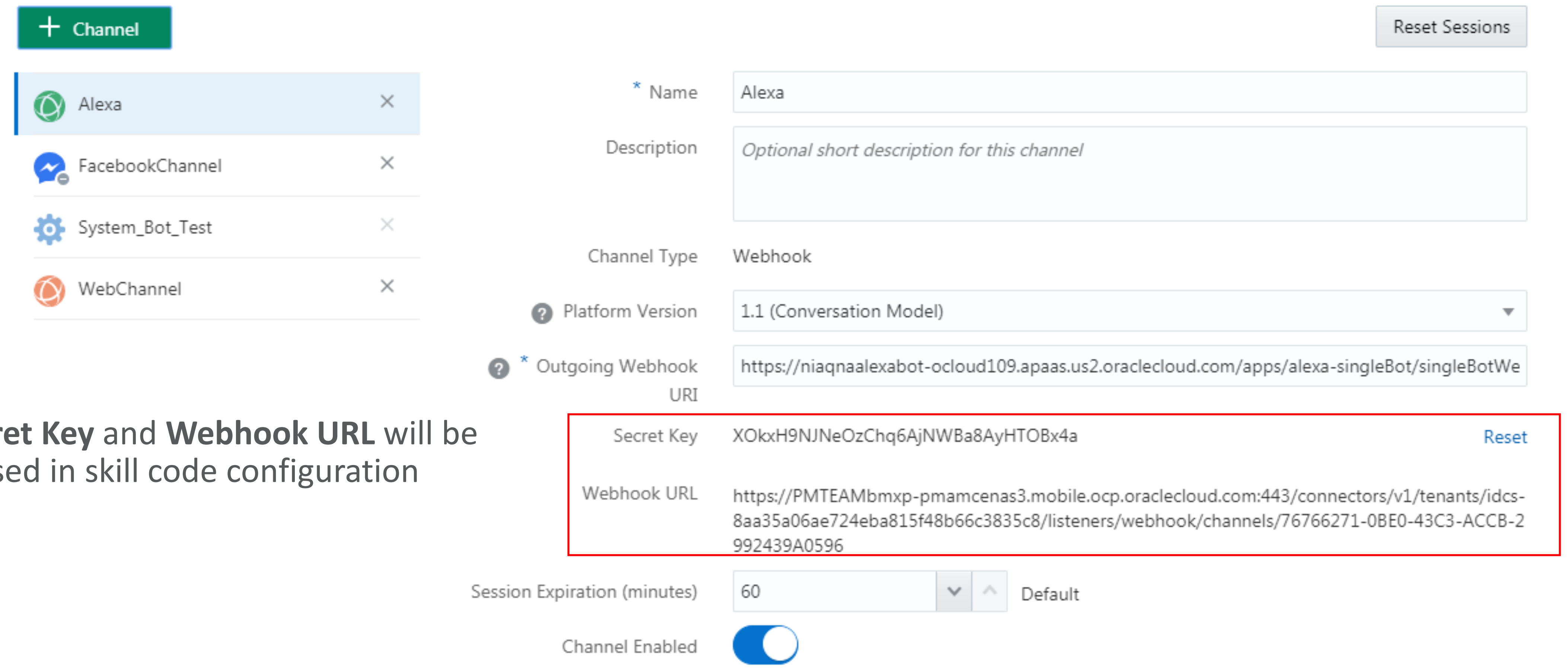
Outgoing Webhook URI:
https://niaqnaalexabot-ocloud109.apaas.us2.oraclecloud.com/apps/alexa-singleBot/singleBotWebhook/messages

Oracle Digital Assistant – Creating a webhook channel



* Name	Alexa
Description	*Optional short description for this channel*
Channel Type	Webhook
Platform Version	1.1 (Conversation Model)
* Outgoing Webhook URI	https://niaqnaalexabot-ocloud109.apaas.us2.oraclecloud.com/apps/alexa-singleBot/singleBotWe
Secret Key	XOkxH9NJNeOzChq6AjNWBa8AyHTOBx4a Reset
Webhook URL	https://PMTEAMbmxp-pmamcenas3.mobile.ocp.oraclecloud.com:443/connectors/v1/tenants/idcs-8aa35a06ae724eba815f48b66c3835c8/listeners/webhook/channels/76766271-0BE0-43C3-ACCB-2992439A0596
Session Expiration (minutes)	60 Default
Channel Enabled	

Oracle Digital Assistant – Creating a webhook channel



* Name	Alexa
Description	*Optional short description for this channel*
Channel Type	Webhook
Platform Version	1.1 (Conversation Model)
* Outgoing Webhook URI	https://niaqnaalexabot-ocloud109.apaas.us2.oraclecloud.com/apps/alexa-singleBot/singleBotWe
Secret Key	XOkxH9NJNeOzChq6AjNWBa8AyHTOBx4a
Webhook URL	https://PMTEAMbmxp-pmamcenas3.mobile.ocp.oraclecloud.com:443/connectors/v1/tenants/idcs-8aa35a06ae724eba815f48b66c3835c8/listeners/webhook/channels/76766271-0BE0-43C3-ACCB-2992439A0596
Session Expiration (minutes)	60 Default
Channel Enabled	

Secret Key and **Webhook URL** will be passed in skill code configuration

You find a complete **webhook example for Alexa** voice integration **on** the **Oracle TechExchange** blog

https://blogs.oracle.com/mobile/tech-exchange

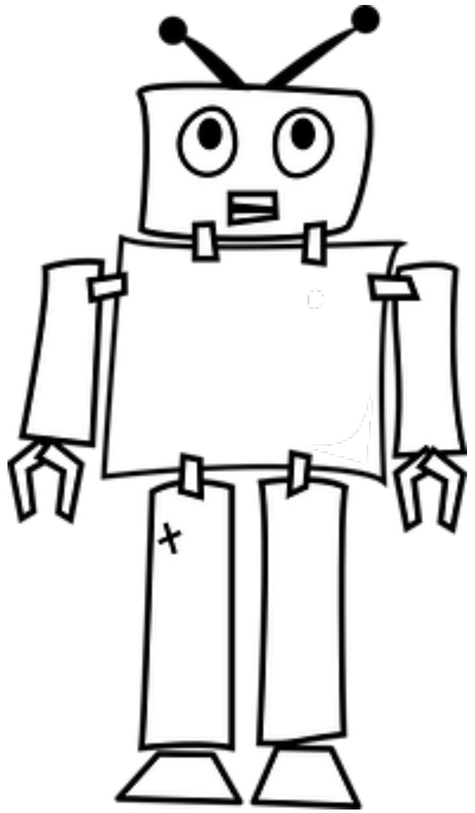


Image courtesy of pixabay.com

Integrated Cloud

Applications & Platform Services

Oracle Digital Assistant Hands-On

TBD