Actions para Google Assistant: seu assistente pessoal

Eduardo Heitor

Big Data Brasil Day– 03/mai/19
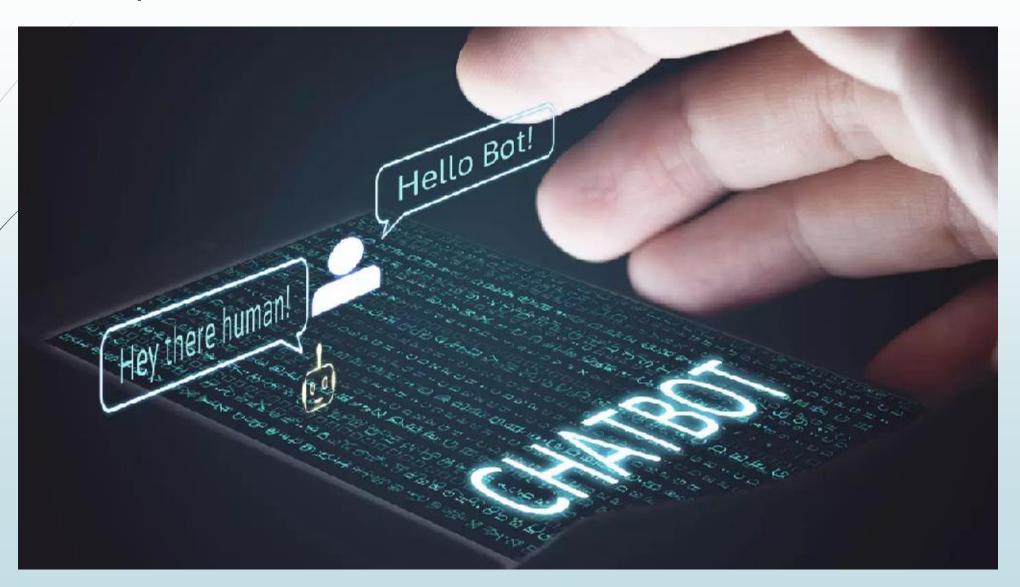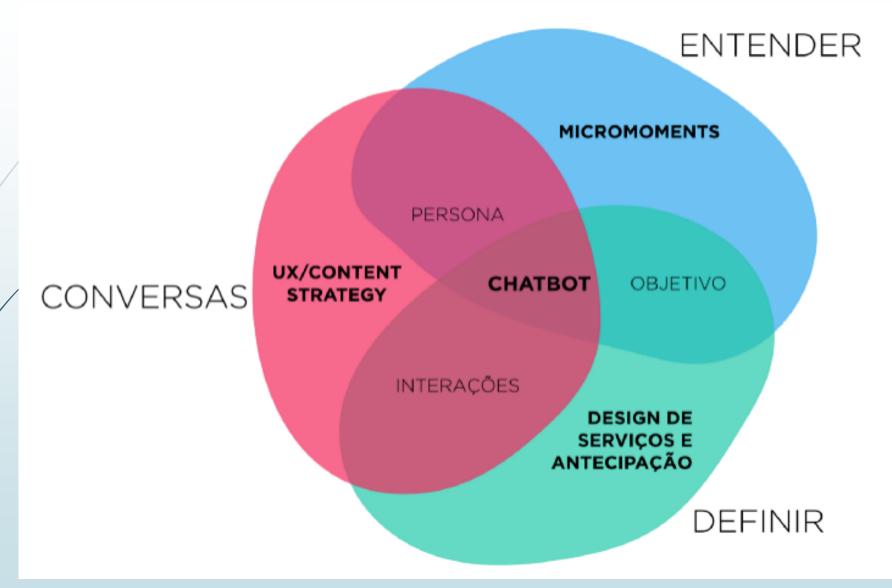
Agenda

- Apresentação
- O que é chatbot
- Google Assistente
- Actions on Google
- Exemplos de uso no Brasil
- Documentação Google
- Exemplo prático para o GDG Ribeirão Preto
- Firebase e arquitetura serverless
- Web services no Actions
- Análise em fluxo contínuo
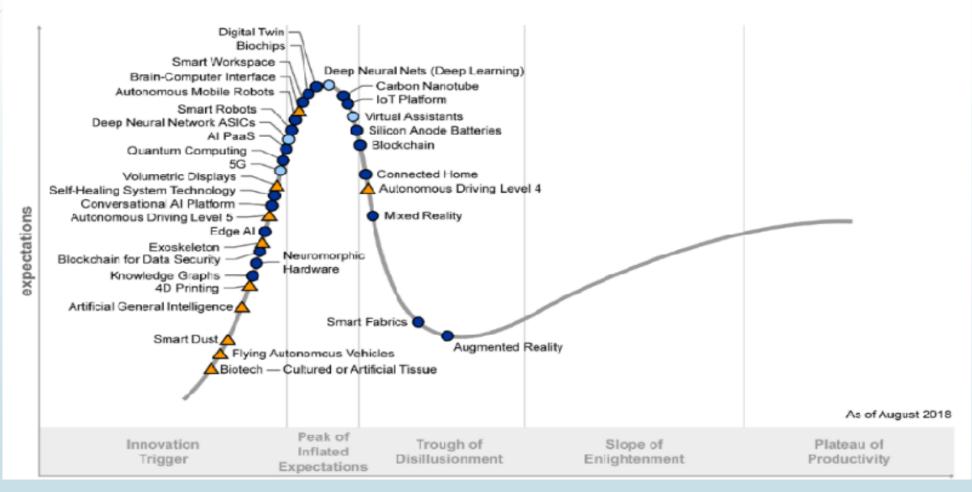
Apresentação – Eduardo Heitor

- Carreira em desenvolvimento na área de Tecnologia da Informação e Comunicação, atuando há 23 anos em empresas como Oi, Ipiranga, Texaco, Bradesco, Itaú, Drogasil, Vivo, Alcatel-Lucent, Genesys, Tivit e Todo, com foco na liderança de equipes em projetos de desenvolvimento de aplicativos de CTI, URA, gravadores digitais e reconhecimento de fala.

- Profissional com certificações da FGV e USP, além de experiência em diversas linguagens de programação

- Atuação como conselheiro da ONG CVU – Centro de Voluntariado Universitário

- Membro da Gestão do Neuron Data Science / USP RP

- Entusiasta de Inteligência Artificial e Aprendizado de Máquina

O que é chatbot?

Hype Cycle for Emerging Technologies, 2018

fonte: https://www.gartner.com/en/newsroom/press-releases

Chatbots Will Appeal to Modern Workers
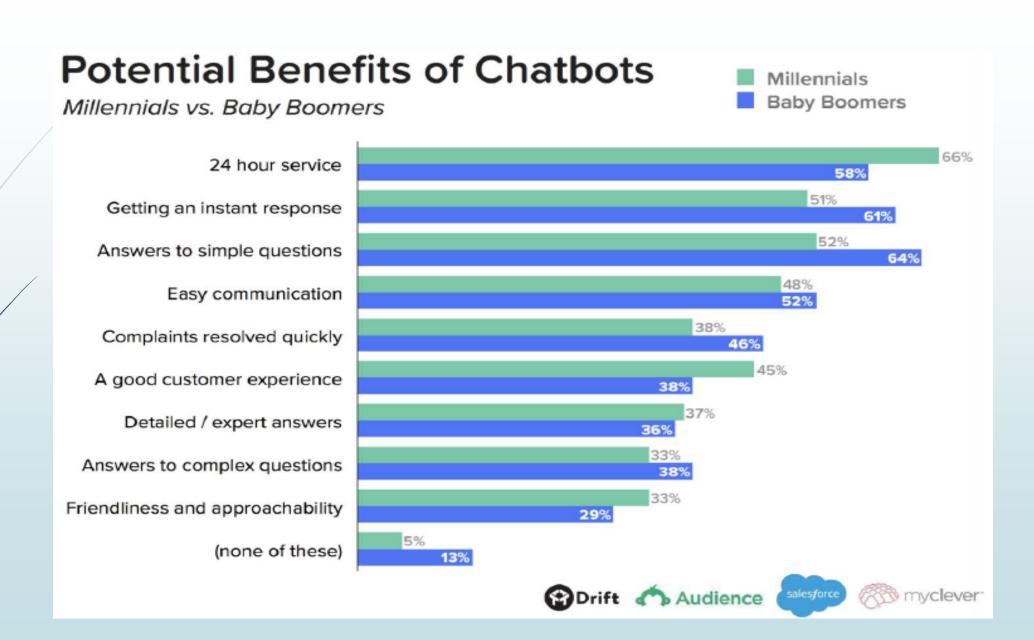
March 28, 2018
Contributor: Laurence Goasduff

TRENDS

Chatbots are on the rise thanks to their ability to mimic conversations and offer instant, digital connections.

This article has been updated from the original, published on June 9, 2017 to reflect new events, conditions or research.
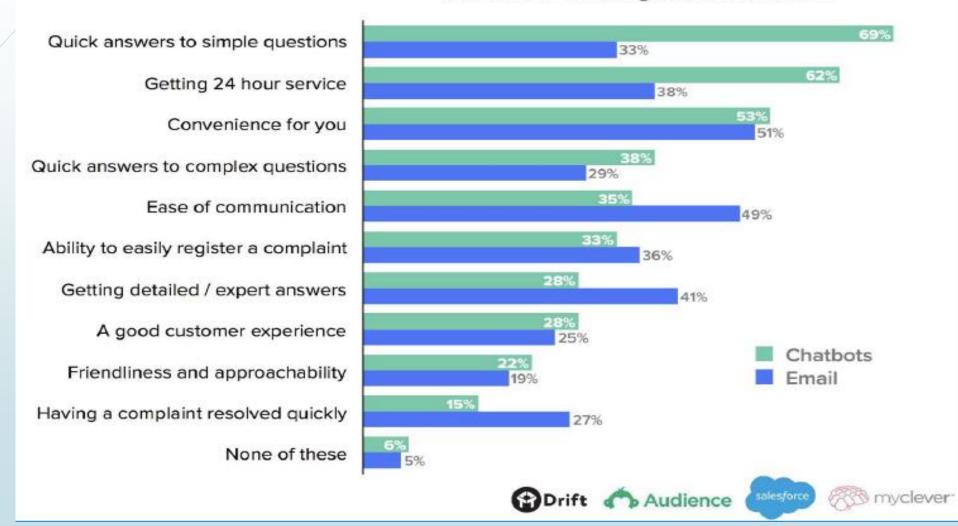
Chatbots continue to be a hot topic among media, end users and vendor communities. This is no surprise as chatbots — which use artificial intelligence (AI) to mimic human conversations — are expected to become widespread within the next two to three years.

"By 2020, over 50% of medium to large enterprises will have deployed product chatbots," said Van Baker, research vice president at Gartner, while at the Gartner Application Architecture, Development & Integration Summit, held March 12-13 in Mumbai.
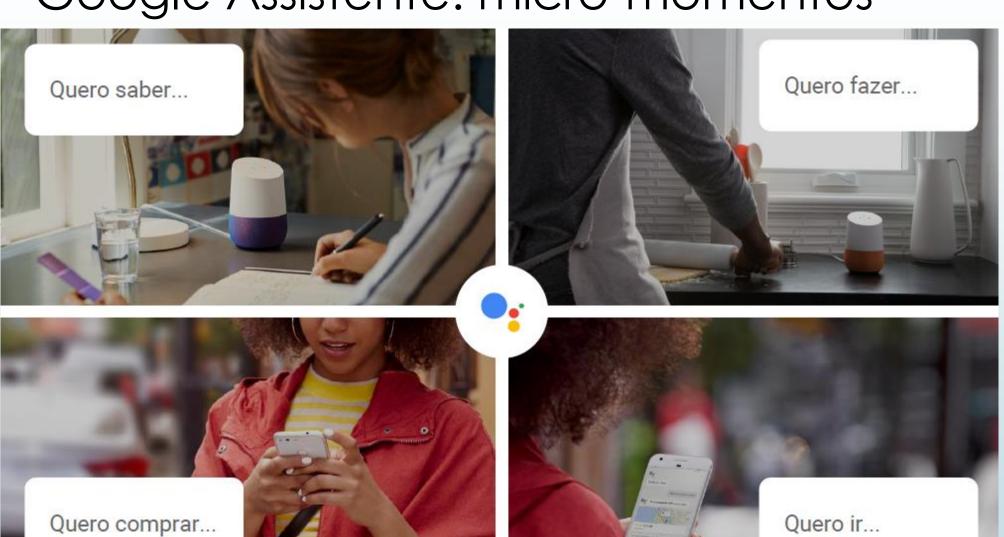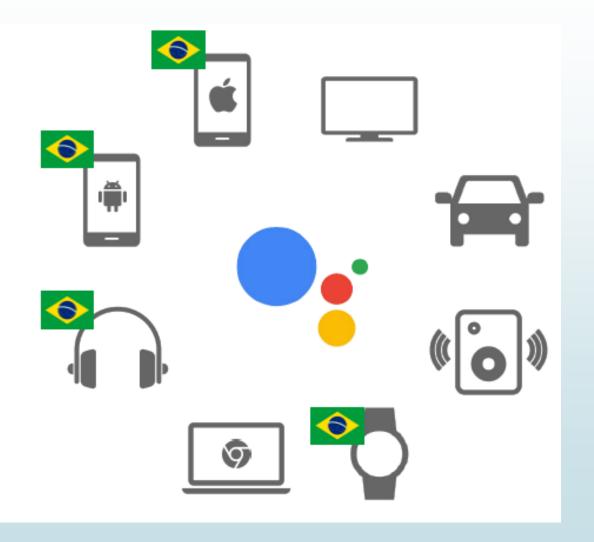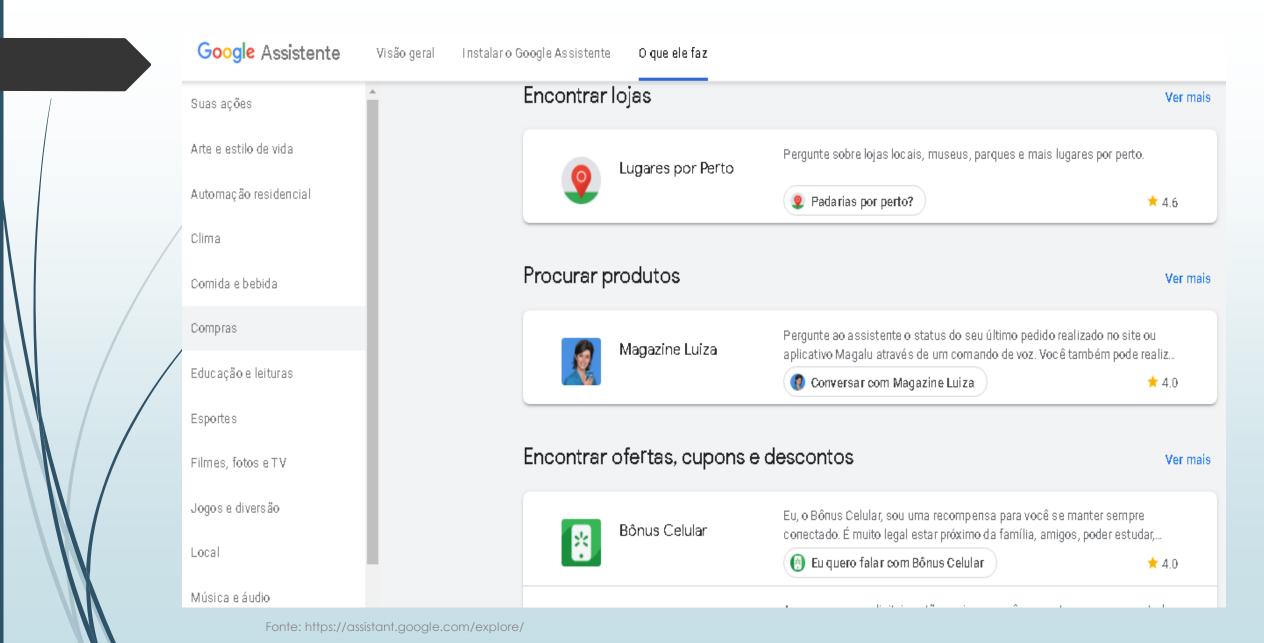
Google Assistente: micro-momentos

Disponível em múltiplos dispositivos

Fonte: https://assistant.google.com/explore/

Actions on Google

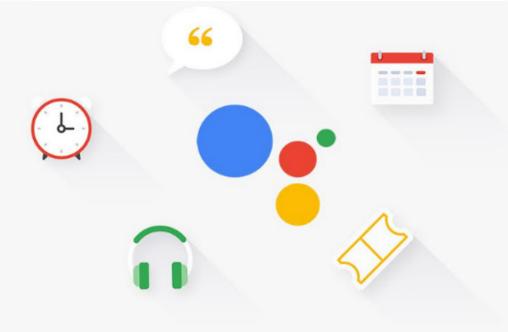actions.google.com

Com acesso a tecnologias do Google

📍 Location & Context awareness

🔴 Knowledge Graph

🟡 Identity

🟢 *Payments
*TBD no BR

🔵 Structured Data Markup

🔴 Natural Language Understanding

🟡 Machine Learning

🟢 Individual speech recognition

Actions on Google

Help users get things done across Google

Engage billions of users across the Google Assistant and soon Google Search, Android, and other surfaces by developing Actions and linking them with our ever-growing intents catalog.

START BUILDING   LEARN MORE

Conversation developers

Design, build, and distribute fully-customizable conversational Actions across Google, using Dialogflow or one of our many templates.

Smart home developers

Build smart home Actions allowing you to connect, query, and control IoT devices through your cloud infrastructure.

Content creators

Make content you've already integrated in Google services (such as AMP news, recipes markup, and podcasts) more discoverable, by claiming and refining automatically generated Actions in our directory.
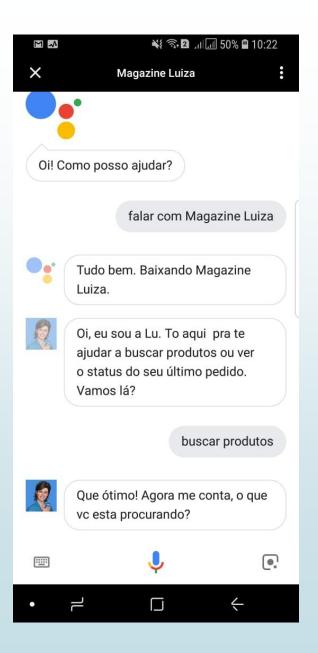
Coming Soon! Android developers

Create an Action by adding an Actions.xml file to your Android app and improve its discovery and re-engagement across multiple, billion-user Google products.

Fonte: https://developers.google.com/actions/

Exemplos de uso

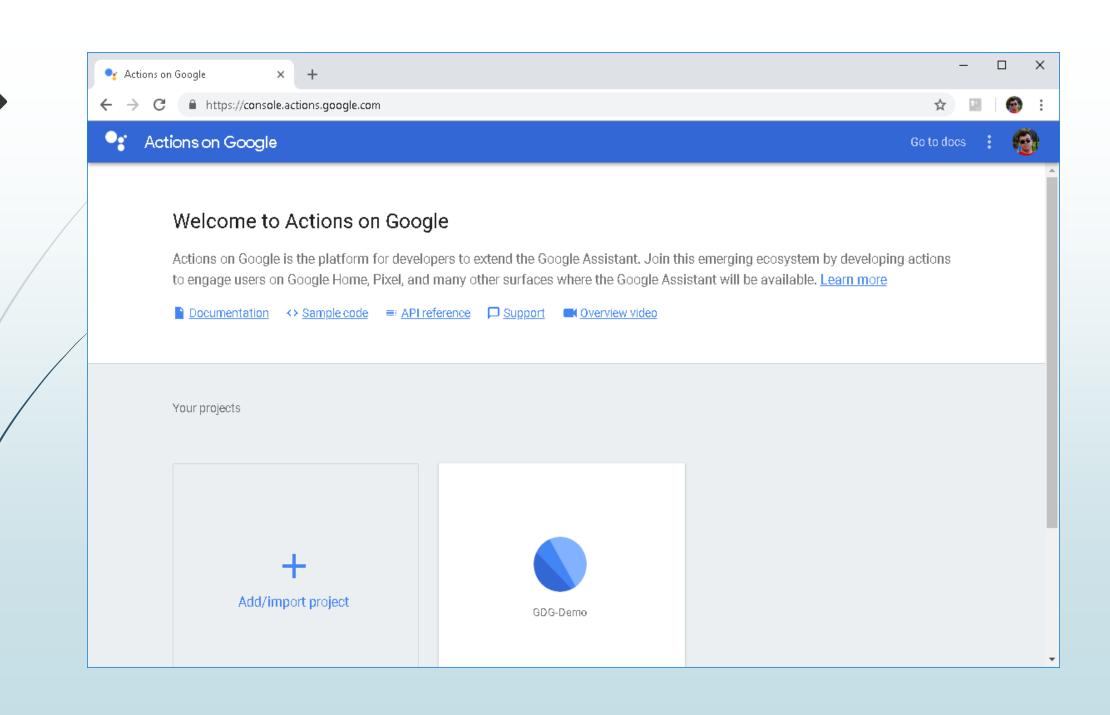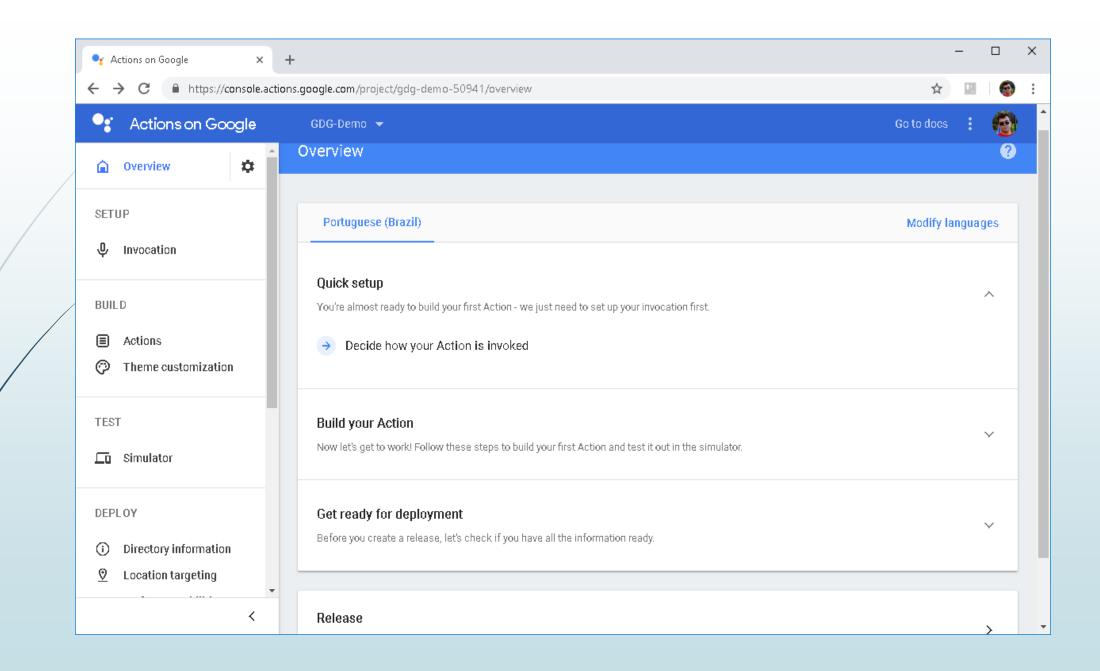➧ OK, Google, falar com Magazine Luiza
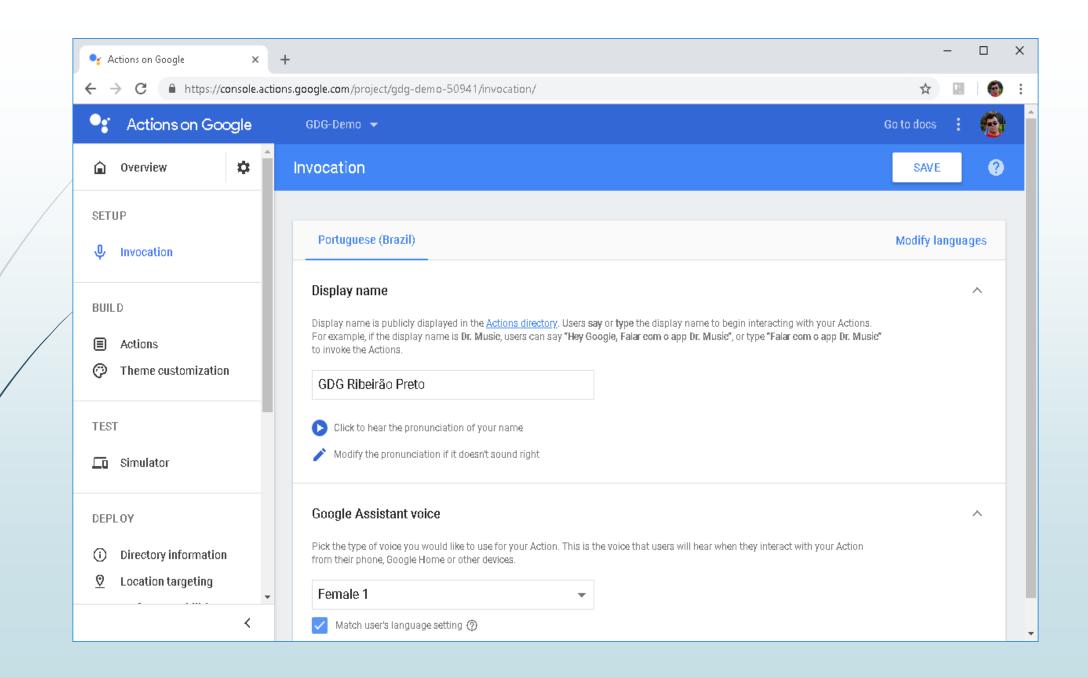
➧ OK, Google, falar com a Caixa

Documentação

- Baixe a amostra do Fatos Sobre o Google: https://g.co/dev/Facts-About-Google

- Visite o Console do Actions: https://console.actions.google.com

- Dê uma olhada na documentação: https://developers.google.com/actions/

- Junte-se à nossa comunidade G+: https://g.co/actionsdev

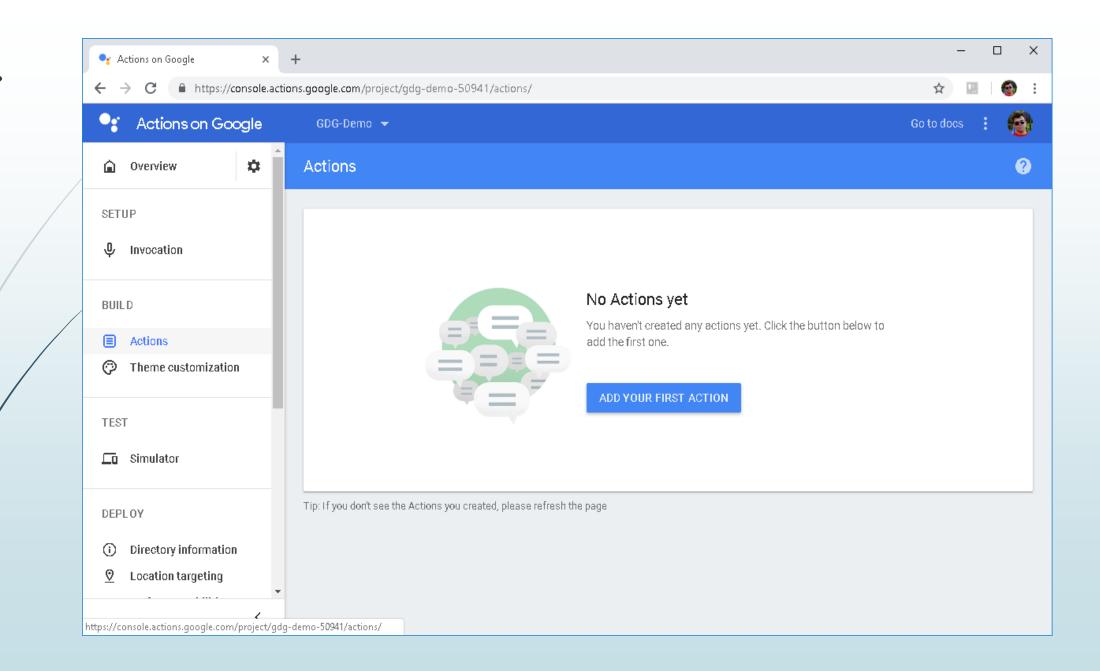- Assistant para seu negócio: https://assistant.google.com/business/
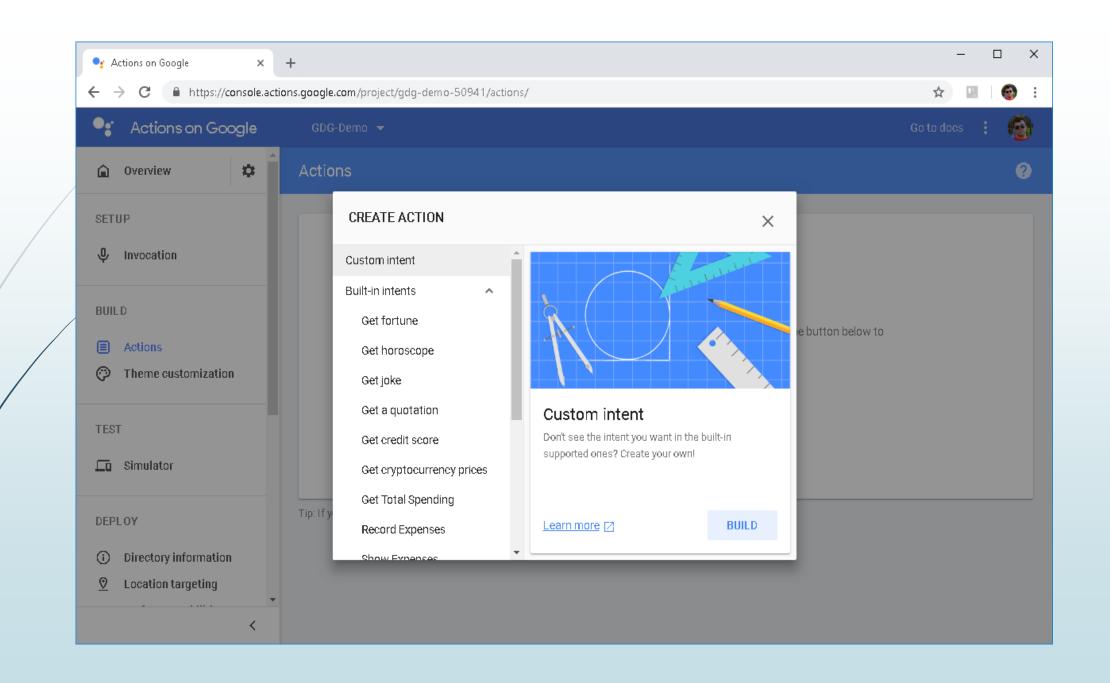
Exemplo prático para o GDG RP

- Vamos criar um aplicativo simples para esta palestra no GDG Ribeirão Preto

- Conceitos básicos:

  - **Intenção:** é o desejo que o chatbot perceberá que o usuário possui ao enviar uma mensagem específica. Por exemplo: ao enviar um "obrigado" a intenção do usuário é agradecer. Sendo assim a intenção para a frase poderia ser "agradecimento".

  - **Entidade:** é o substantivo relacionado ao desejo que foi detectado pelo chatbot. Por exemplo: Se a frase enviada pelo usuário for "gerar novo relatório", a entidade poderia ser "relatório" ou "novo relatório"

  - **Diálogo:** é onde se cria as respostas que o chatbot retornará ao detectar uma intenção e/ou uma entidade. Por exemplo: se o chatbot detectar que a intenção é "agradecimento" ele poderia responder "obrigado por conversar comigo".
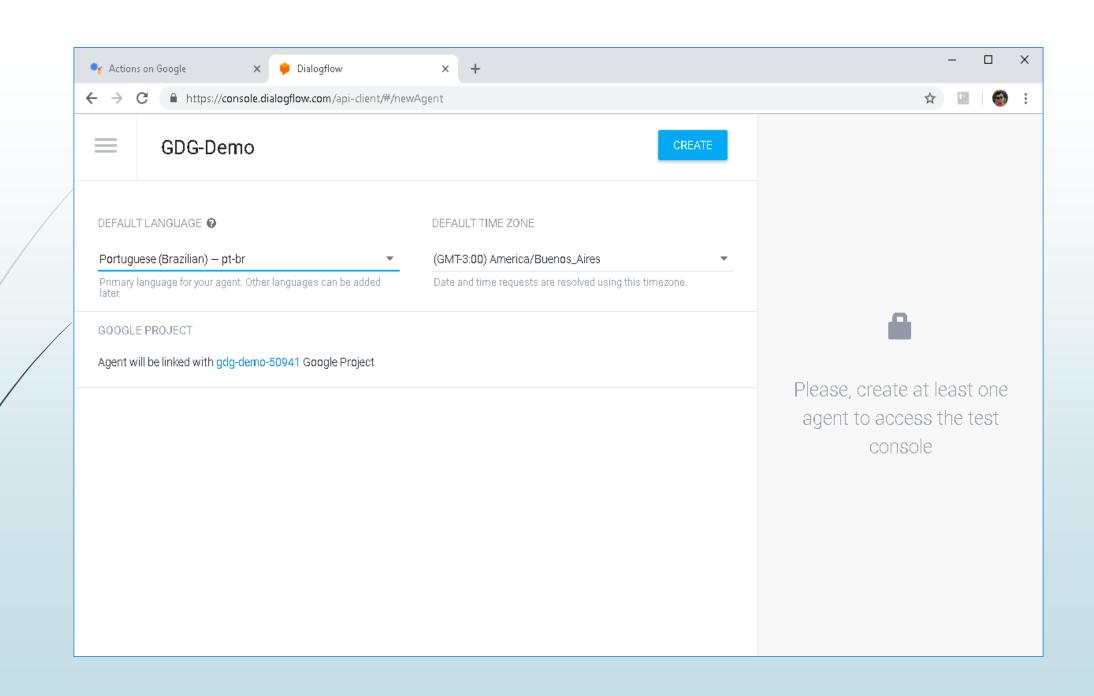
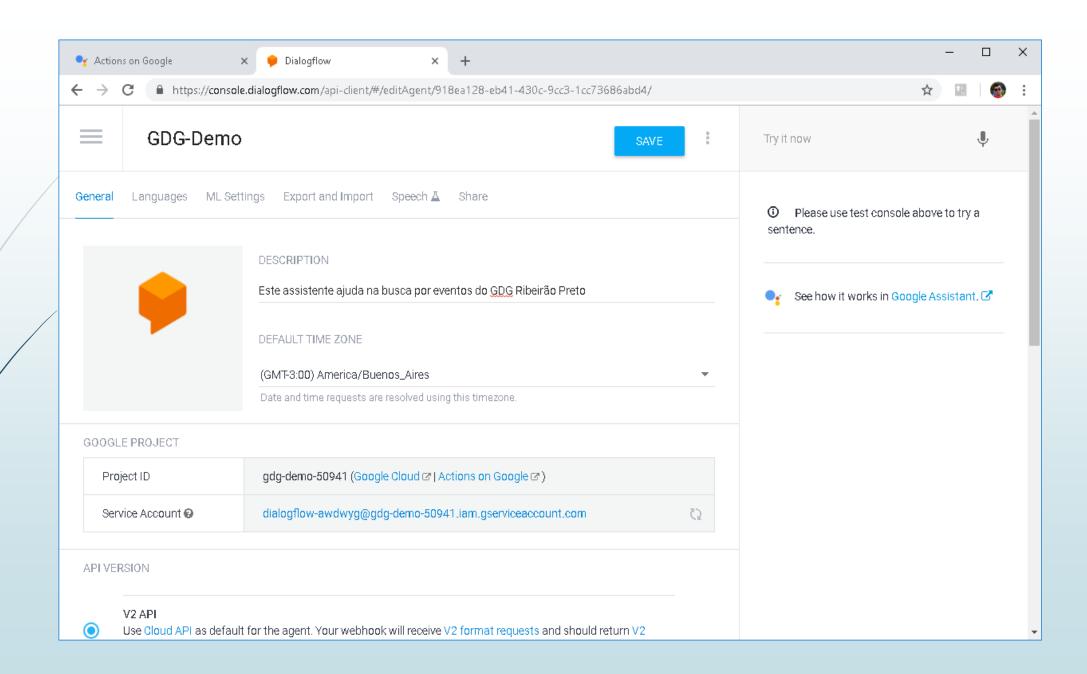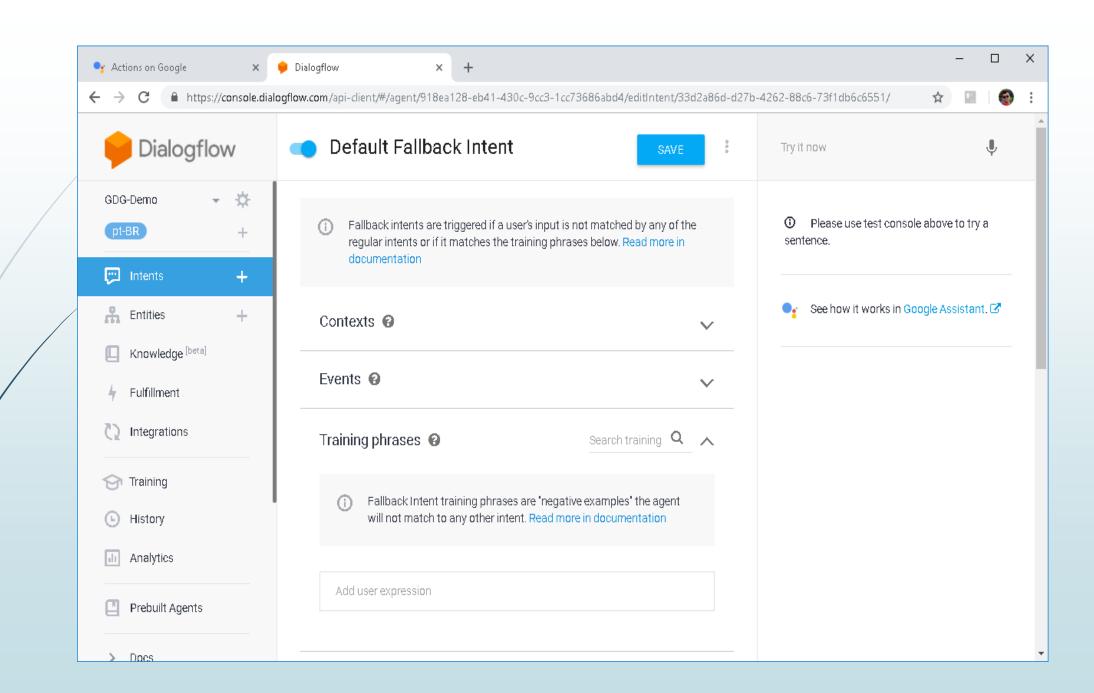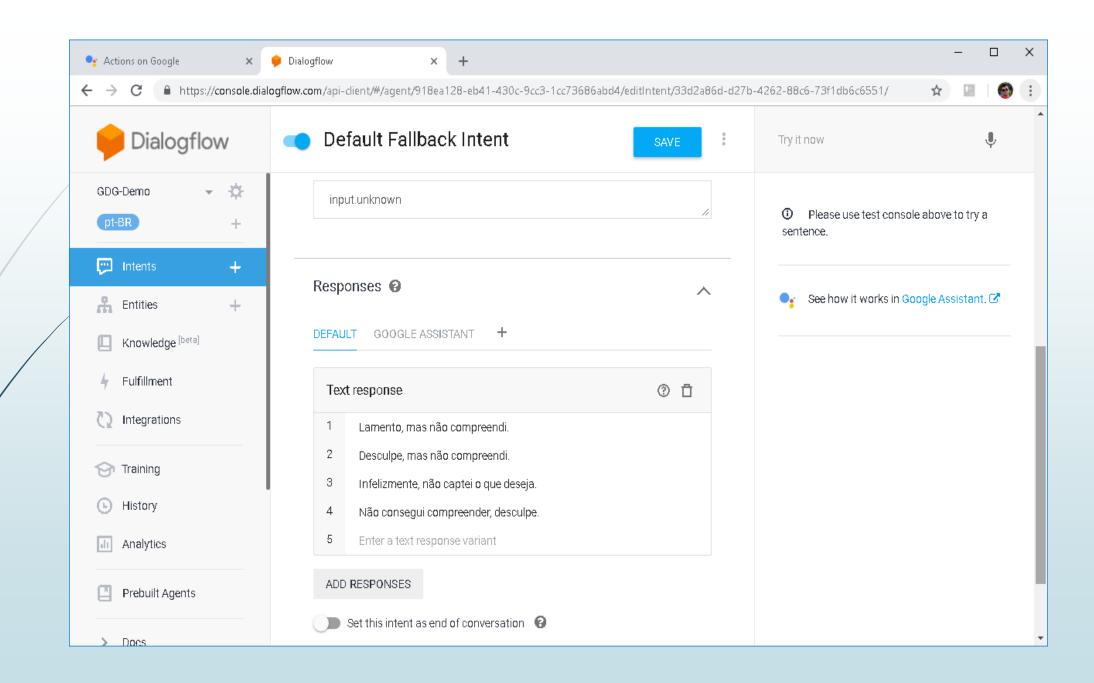Fonte: https://codelabs.developers.google.com/codelabs/actions-1

Firebase e arquitetura serverless

Serverless 1: Backend as a Service (BaaS)

Bancos de dados, serviços de autenticação, analytics e outros acessíveis diretamente via APIs

Firebase e arquitetura serverless

Firebase e arquitetura serverless

Serverless 2: Function as a Service (FaaS)

Lógica server-side disparada por eventos e que executa em containers stateless gerenciados por um terceiro (provedor do serviço de FaaS)

Firebase e arquitetura serverless

DevOps, NoOps e Fully Managed

Movimento em busca de automação de produção de software e redução(ou eliminação) da necessidade de tarefas operacionais e recursos computacionais visíveis

Firebase e arquitetura serverless

BaaS e FaaS - Vantagens

- Custos de escalabilidade reduzidos (paga somente por tempo de execução e memória alocada para a função ou quantidade de operações e armazenamento)

- Custos operacionais reduzidos ou gratuitos

- Custos de desenvolvimento reduzidos
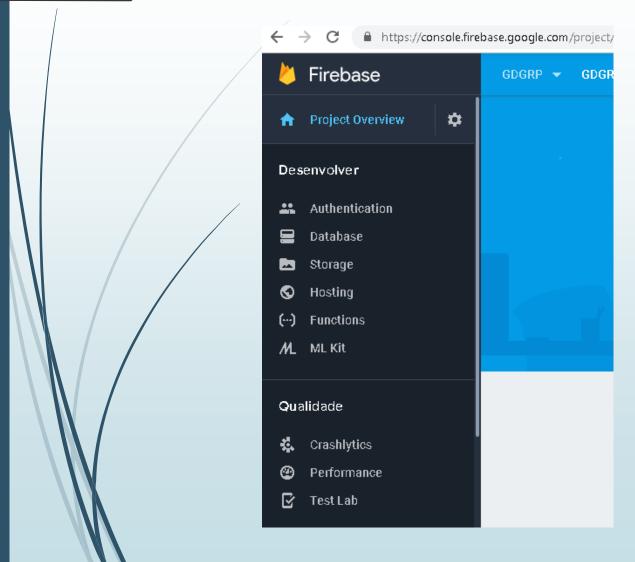
http://martinfowler.com/articles/serverless.htm

Google

Firebase e arquitetura serverless
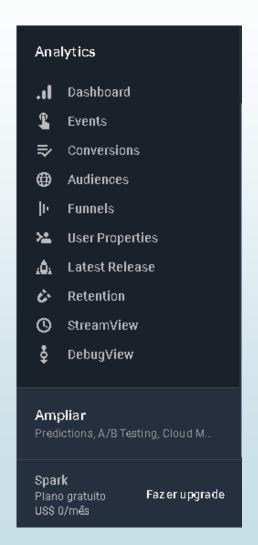
BaaS e Faas - Desvantagens

- Lock-in de provedores Cloud (mitigação através de frameworks open source como Serverless Framework*)

- Repetição de lógica nas camadas cliente, no caso de BaaS

- Sempre 'stateless'
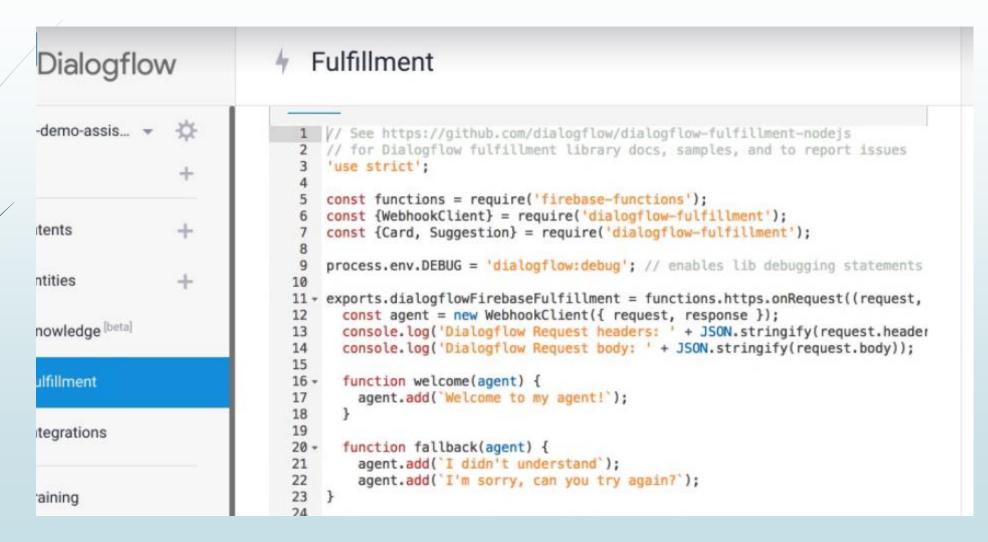
*https://github.com/serverless/serverless

Google

Firebase e arquitetura serverless

Dialogflow com webhook (fulfillment)

Integrate web services (API) to Google actions

Integrate web services to Google actions
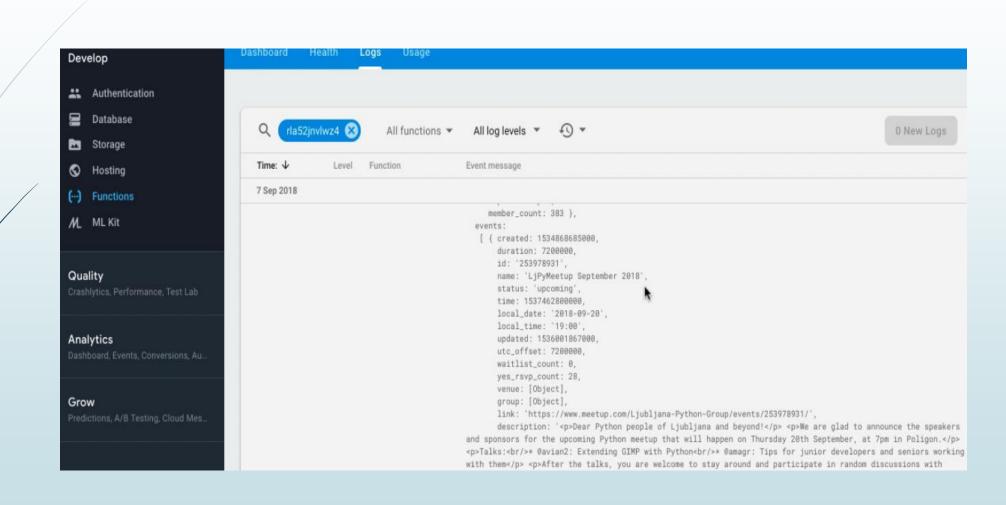
```
index.js ×    config.js ×    package.json ×
 7
 8    process.env.DEBUG = 'dialogflow:debug'; // enables lib debugging statements
 9
10    var admin = require("firebase-admin");
11
12    var serviceAccount = require("./config/udemy-demo-assistant-7912e-firebase-adminsdk-63ims-24dd853435.json");
13
14
15    if (!config.MEETUP_KEY) {
16        throw new Error('missing MEETUP_KEY');
17    }
18
19    const requestAPI = require('request-promise');        ←→
20
21    admin.initializeApp({
22        credential: admin.credential.cert(serviceAccount),
23        databaseURL: "https://udemy-demo-assistant-7912e.firebaseio.com"
24    });
25
26    exports.dialogflowFirebaseFulfillment = functions.https.onRequest((request, response) => {
27        const agent = new WebhookClient({ request, response });
28        console.log('Dialogflow Request headers: ' + JSON.stringify(request.headers));
29        console.log('Dialogflow Request body: ' + JSON.stringify(request.body));
30
31        function welcome(agent) {
32            agent.add(`Welcome to my agent!`);
33        }
34
```

Integrate web services to Google actions

```
131    // // Uncomment and edit to make your own Google Assistant intent handler
132    //    // uncomment `intentMap.set('your intent name here', googleAssistantHandler);`
133    //    // below to get this function to be run when a Dialogflow intent is matched
134    //    function googleAssistantHandler(agent) {
135    //        let conv = agent.conv(); // Get Actions on Google library conv instance
136    //        conv.ask('Hello from the Actions on Google client library!') // Use Actions on Google library
137    //        agent.add(conv); // Add Actions on Google library responses to your agent's response
138    //    }
139        // See https://github.com/dialogflow/dialogflow-fulfillment-nodejs/tree/master/samples/actions-on-google
140        // for a complete Dialogflow fulfillment library Actions on Google client library v2 integration sample
141
142    //    Run the proper function handler based on the matched Dialogflow intent name
143        let intentMap = new Map();
144        intentMap.set('Default Welcome Intent', welcome);
145        intentMap.set('Default Fallback Intent', fallback);
146        intentMap.set('music vote', voting);
147        intentMap.set('vote results', voteResults);|
148        intentMap.set('show meetups', showMeetups);    ◄──────►
149
150    // intentMap.set('your intent name here', yourFunctionHandler);
151    // intentMap.set('your intent name here', googleAssistantHandler);
152        agent.handleRequest(intentMap);
153    });
154
```

Integrate web services to Google actions

```
26    exports.dialogflowFirebaseFulfillment = functions.https.onRequest((request, response) => {
27        const agent = new WebhookClient({ request, response });
28        console.log('Dialogflow Request headers: ' + JSON.stringify(request.headers));
29        console.log('Dialogflow Request body: ' + JSON.stringify(request.body));
30
31        function welcome(agent) {
32            agent.add(`Welcome to my agent!`);
33        }
34
35        function fallback(agent) {
36            agent.add(`I didn't understand`);
37            agent.add(`I'm sorry, can you try again?`);
38        }
39
40        function showMeetups(agent) {
41            displayMeetup();
42        }
43
44        async function voteResults(agent) {
```

Integrate web services to Google actions

```
35    function fallback(agent) {
36        agent.add(`I didn't understand`);
37        agent.add(`I'm sorry, can you try again?`);
38    }
39
40    function showMeetups(agent) {
41        displayMeetup();
42    }
43
44    function displayMeetup() {
45        return requestAPI('https://api.meetup.com/find/upcoming_events?' +
46            '&sign=true&photo-host=public&lon=14.493240&page=30&lat=46.048226&key=' +
47            config.MEETUP_KEY)
48            .then(function (data) {
49                let meetups = JSON.parse(data);
50                console.log(meetups);
51            });
52    }
53
```

Integrate web services to Google actions

Integrate web services to Google actions

```
63    async function displayMeetup() {
64        if (conv.data.meetupData.length === 0 ) {
65            await getMeetupData();
66            return buildSingleMeetupResponse();
67        } else {
68            return buildSingleMeetupResponse();
69        }
70    }
71
72
73    function buildSingleMeetupResponse() {
74        let responseToUser;
75        if ( conv.data.meetupData.length === 0 ) {
76            responseToUser = 'No meetups available at this time!';
77            conv.ask(responseToUser);
78        } else {
79            let meetup = conv.data.meetupData[0];
80            responseToUser = ' Meetup number 1 ';
81            responseToUser += meetup.name;
82            responseToUser += ' by ' + meetup.group.name;
83
84            let date = new Date(meetup.time);
85            responseToUser += ' on ' + date.toDateString() + '.';
86
87            conv.ask(responseToUser);
88            |
89        }
90        return conv;
91    }
92
93    function getMeetupData() {
94        return requestAPI('https://api.meetup.com/find/upcoming events?' +
```

Integrate web services to Google actions

```
"local_date":"2018-09-20",
"local_time":"19:00",
"updated":1536001867000,
"utc_offset":7200000,
"waitlist_count":0,
"yes_rsvp_count":32,
"venue":{ ⊞ },
"group":{ ⊟
    "created":1347628935000,
    "name":"Ljubljana Python User Group",
    "id":4926572,
    "join_mode":"open",
    "lat":46.060001373291016,
    "lon":14.510000228881836,
    "urlname":"Ljubljana-Python-Group",
    "who":"Pythonistas",
    "localized_location":"Ljubljana, Slover
    "region":"en_US",
    "timezone":"Europe/Ljubljana"
},
"link":"https://www.meetup.com/Ljubljana-F
"description":"<p>Dear Python people of Lj
We are glad to announce the speakers and sp
Talks:
lavian2: Extending GIMP with Python<br/>* @a
More info and RSVP on <a href=\"https://www
ents/253978931\" class=\"linkified\">https:
"visibility":"public"
⊟
"created":1519847206000
```

```javascript
    conv.ask(responseToUser);
} else {
    let meetup = conv.data.meetupData[0];
    responseToUser = ' Meetup number 1 ';
    responseToUser += meetup.name;
    responseToUser += ' by ' + meetup.group.name;

    let date = new Date(meetup.time);
    responseToUser += ' on ' + date.toDateString() + '.';

    conv.ask(responseToUser);

    if (conv.surface.capabilities.has('actions.capability.SCREEN_OUTPUT')) {

        let image = 'https://raw.githubusercontent.com/jbergant/udemydemoimg/master/meetup.png';
        conv.ask(new BasicCard({
            text: meetup.description,
            subtitle: 'by ' + meetup.group.name,
            title: meetup.name,
            buttons: new Button({
                title: 'Read more',
                url: meetup.link,
            }),
            image: new Image({
                url: image,
                alt: meetup.name,
            }),
            display: 'CROPPED',
        }));
    }
}
return conv;
```

Muito Obrigado !

- Eduardo Heitor
  - https://linkedin.com/in/eduardoheitor/
  - https://www.facebook.com/eduardo.heitor