

MegAlexa

Aricchitore di skill di Amazon Alexa

VERBALE ESTERNO DEL 2019-02-15

GRUPPO ZEROSEVEN



Versione	1.0.0
Data Redazione	2019-02-17
Redazione	Bianca Andreea Ciuche
Verifica	Matteo Depascale
Approvazione	
Uso	Esterno
Distribuzione	Prof. Tullio Vardanega Prof. Riccardo Cardin Gruppo ZeroSeven Zero12 s.r.l.
Email di contatto	zerosevnswe@gmail.com

Versione	Data	Descrizione	Autore	Ruolo
0.1.0	2019-02-20	Verifica verbale	Matteo Depa-scale	Verificatore
0.0.2	2019-02-17	Stesura verbale	Bianca Andreea Ciuche	Analista
0.0.1	2018-12-08	Creazione template documento	Ludovico Brocca	Amministratore

Tabella 1: Diario delle modifiche

Indice

1	Informazioni sulla riunione	2
1.1	Motivo della riunione	2
2	Resoconto	2
2.1	Argomenti discussi	2
2.2	Tracciamento decisioni	3

1 Informazioni sulla riunione

1.1 Motivo della riunione

E' stata richiesta questa riunione per aggiornare la Proponente sul lavoro che stiamo svolgendo e per chiarire alcuni dubbi riguardanti le tecnologie che sono sorte lavorando sul prototipo.

- **Luogo e data:** Google Hangout, Venerdì 15 Febbraio 2019;
- **Ora di inizio:** 17:00;
- **Ora di fine:** 17:20;
- **Partecipanti:**
 - Gian Marco Bratzu;
 - Mirko Franco;
 - Ciuche Bianca;
 - Ludovico Brocca;
 - Matteo Depascale;
 - Stefano Zanatta;
 - Sig. Stefano Dindo.

2 Resoconto

2.1 Argomenti discussi

Nel corso della riunione effettuata mediante Hangouts, è stato mostrato all'azienda il mock dell'applicazione. Inoltre, sono state discusse varie problematiche emerse durante la programmazione del prototipo.

- Si è discusso sulla possibilità di integrare il codice delle funzioni lambda con github per avere un codice più controllato senza effettuare ogni volta il caricamento tramite lo zip. Git hub ha un'area di configurazione che si interfaccia con *AmazonWebService_G*, quindi si può, dopo aver eseguito un commit far partire uno script che si collega al git di una lambda che effettua lo zip e la carica tramite l' *API_G* di Amazon. Oppure c'è la possibilità di usare Code Commit, la quale è stata scartata dal gruppo preferendo github

- Si è discusso sullo sviluppo del voice dialog flow. Questi devono essere composti da tutte le possibili iterazioni necessarie per il progetto. Bisogna inoltre creare dei scenari in cui si deve chiarire quali workflow l'utente ha a disposizione.
- Si è discusso su come avviene in dettaglio il login. E' corretto eseguire la registrazione e il login tramite un *account_G* Amazon. Inoltre è possibile interrogare *Amazon_G*, usando un token, per avere i dati dell'utente provenienti da Amazon. Dopo aver effettuato il login una prima volta, successivamente è possibile effettuare il login automatico grazie al token. Il token ha un tempo di durata durante il quale rimane attivo.
- Si è discusso sull'utilizzo di *DynamoDB_G*. E' corretto scaricare le classi che JSON genera in automatico, queste classi rispecchiano la struttura del nostro database.
- Per testare il codice in *NodeJs_G* è stato consigliato l'utilizzo di due framework: *Mocha_G* assieme alla libreria *Chai_G* oppure *Asser Istanbul_G*.

2.2 Tracciamento decisioni

Codice	Decisione
01-VER_2019-02-15	Sviluppare due blocchi per la consegna del prototipo.
02-VER_2019-02-15	Sviluppare il voice dialog flow con una struttura a dialogo.
03-VER_2019-02-15	Mocha: framework utilizzato per effettuare i test sul codice in NodeJS.

Tabella 2: Tracciamento decisioni