



Verbale Riunione 2022-01-27

Informazioni sul Documento

Versione	1.0.0
Data di approvazione	2022-02-06
Approvatori	Michele Gatto
Redattori	Greta Cavedon
Verificatori	Francesco Protopapa
Uso	Interno
Distribuzione	Prof. Vardanega Tullio Prof. Cardin Riccardo Gruppo <i>Dream Team</i>

e-mail: dreamteam.unipd@gmail.com



Registro delle Modifiche

Versione	Data	Nominativo	Ruolo	Descrizione
v1.0.0	2022-02-06	Michele Gatto	Responsabile	Approvazione del documento
v0.1.0	2022-02-03	Greta Cavedon	Analista	Verifica di coerenza e coesione complessiva. Verificatore: <i>Francesco Protopapa</i>
v0.0.1	2022-01-27	Greta Cavedon	Analista	Realizzazione struttura in Latex e stesura documento. Verificatore: <i>Francesco Protopapa</i>



Indice

1	Informazioni Generali	4
2	Ordine del Giorno	4
3	Resoconto	4
3.1	Andamento del PoC	4
3.1.1	API Gateway	4
3.1.2	API Google Maps	5
3.2	Dati da inserire nel database per il PoC	5
3.3	Stato dei Documenti	5
4	Riepilogo delle Decisioni	5



1 Informazioni Generali

- **Luogo:** Meeting Discord
- **Data:** 2022-01-27
- **Ora:** 13:00 - 14:00
- **Partecipanti del Gruppo:**
 - Edoardo Pavan,
 - Francesco Protopapa,
 - Greta Cavedon,
 - Luciano Wu,
 - Michele Gatto.
- **Segretario:** Greta Cavedon

2 Ordine del Giorno

- Andamento del PoC
 - API Gateway
 - API Google Maps
- Dati da inserire nel database per il PoC
- Stato dei Documenti

3 Resoconto

3.1 Andamento del PoC

Nelle ultime settimane ci siamo occupati principalmente del PoC ed in questa riunione abbiamo messo in luce alcune problematiche e cambi di ruolo, in particolare:

- Edoardo Pavan si occuperà delle API Gateway, anziché della parte front-end come stabilito inizialmente, in quanto non erano state prese in carico da nessuno;
- Luciano Wu si occuperà della parte front-end al posto di Edoardo Pavan;
- a Pietro Villatora è stata assegnata la parte relativa alle **API di Google Maps**, poiché possono tornare utili per identificare ciascun locale (ad esempio: è possibile identificare un locale in base alle coordinate geografiche) e nessuno se ne stava occupando.

Gli altri componenti del gruppo continuano a lavorare su ciò che avevano iniziato: Francesco Protopapa e Matteo Basso si stanno occupando di far funzionare il **crawler di Instagram** e di interfacciarlo con i servizi di AWS, Greta Cavedon si sta occupando della parte **front-end**, mentre Michele Gatto sta continuando ad informarsi e sperimentare con il **crawler di TikTok**, considerati anche i consigli dati ieri da parte dell'azienda (*Verbale Esterno 2022-01-26*).

Inoltre, si è discusso se avesse senso o meno utilizzare NodeJS per la parte back-end, piuttosto che usare sempre Python (dato che le Lambda sono comunque in Python), oltre al fatto di usare **AWS Amplify** piuttosto che sfruttare una pipeline CI/CD che carichi la WebApp su S3, sfruttando CloudFront (come suggerito da Michele Massaro di Zero12 in una recente comunicazione avvenuta via *Slack*).

3.1.1 API Gateway

Come esposto nel Verbale esterno 2022-01-26, le API Gateway di AWS servono per interfacciare la WebApp con l'architettura della piattaforma, preservando la sicurezza dei dati presenti nel database. Nel nostro sistema le API Gateway fungono da intermediario, infatti il loro scopo è duale: mostrano ciò che un utente chiede tramite la WebApp (ad esempio, se l'utente effettua una ricerca o vuole visualizzare le informazioni di un locale) e recuperano le informazioni richieste all'interno del database mediante delle chiamate API Rest, sfruttando le Lambda di AWS.

3.1.2 API Google Maps

Le **API di Google Maps** potrebbero tornare utili per il nostro sistema, in quanto permettono di integrare le mappe digitali di Google Maps nella propria piattaforma (nel nostro caso la WebApp), di identificare mediante le coordinate geografiche (per esempio, noi possiamo estrapolare le coordinate, come longitudine e latitudine, con il crawler partendo da un'immagine) e sapere come raggiungere un determinato luogo (per esempio: un ristorante o un bar).

L'unico vincolo da considerare sono le chiamate gratuite che si possono fare con queste API di Google Maps (ossia quante volte posso sfruttare la mappa di Google in un mese per progetto), stimate a **28.500 al mese** (Google mette a disposizione 200\$ al mese di prova per ciascuna azienda - <https://mapsplatform.google.com/pricing/>).

3.2 Dati da inserire nel database per il PoC

Per il PoC si è deciso di realizzare un database che contenesse le informazioni ottenute dal crawler, ma che non fosse troppo complesso. A tal proposito, si è deciso di realizzare una tabella contenente le seguenti informazioni:

- Username utente che ha pubblicato l'immagine,
- Indirizzo dell'immagine analizzata con i servizi di AWS,
- Testo del post presente nell'immagine,
- Nome del luogo in cui è stata pubblicata l'immagine,
- Latitudine,
- Longitudine,
- Risultati ottenuti da Comprehend,
- Risultati ottenuti da Rekognition.

Da quanto è emerso ieri nell'incontro con il proponente, si è deciso di includere i risultati ottenuti dai servizi di AWS direttamente nel database, anziché all'interno di file JSON, per maggior praticità (ad esempio, così risulta molto più semplice ed efficiente effettuare delle ricerche, anziché dover cercare informazioni sui file JSON).

3.3 Stato dei Documenti

Infine, abbiamo parlato dello stato di avanzamento dei documenti: sebbene siano a buon punto, ci sono delle piccole migliorie/aggiunte da apportare (ad esempio: nel *Piano di Qualifica v1.0.0* va terminato il consuntivo, va controllato il preventivo e la parte relativa alle metriche –). Inoltre, vanno sistemati alcuni termini del *Glossario v1.0.0* su tutti i documenti.

4 Riepilogo delle Decisioni

Codice	Decisione
VI_2022-01-27.1	Quali informazioni inserire nel database
VI_2022-01-27.2	Informarsi sulle funzionalità e su come usare le API Gateway offerte da AWS
VI_2022-01-27.3	Considerare l'utilità delle API di Google Maps per il nostro sistema

Tabella 1: Riepilogo delle decisioni riunione interna 2022-01-27