



MashUp

Software Engineering Group

info@mashup-unipd.it

Informazioni Documento

Nome documento	<i>Verbale del 2015-02-18</i>
Versione	<i>1.0.0</i>
Data redazione	2015-02-19
Redattori	Cusinato Giacomo
Verificatori	Roetta Marco
Approvazione	Santacatterina Luca
Lista distribuzione	<i>MashUp</i> <i>Prof. Tullio Vardanega</i> <i>Prof. Riccardo Cardin</i> <i>Dott. David Santucci - Zing Srl</i>
Uso	Interno

Sommario

Questo documento riassume il secondo incontro tra il proponente e il gruppo *MashUp*.

1 Informazioni generali

1.1 Dettagli incontro

- **Data:** 2015-02-18
- **Ora di inizio:** 10:30
- **Ora di fine:** 13:00
- **Luogo:** Tenuta Ca' Tron, Via Sile, 41 - 31056 Roncade, Treviso
- **Partecipanti:** Carnovalini Filippo, Ceccon Lorenzo, Cusinato Giacomo, Facchin Nicola, Roetta Marco, Santacatterina Luca, Tesser Paolo

1.2 Ordine del giorno

Di seguito sono elencati gli argomenti che il gruppo ha trattato con il proponente.

- Raffinamento dei moduli che compongono l'applicativo;
- Specifica sulle metriche ricavate dai social network a supporto delle View;
- Valutazione di servizi web a supporto dell'analisi dei dati ricavati;

2 Riassunto della riunione

2.1 Risposte all'ordine del giorno

Le risposte di seguito fornite non sono la trascrizione esatta di quanto detto al momento, ma una elaborazione finale in accordo con il proponente.

Esse saranno fonte di requisiti e di casi d'uso che verranno descritti in maniera più dettagliata nel documento di *Analisi dei Requisiti v2.0.0*.

- La struttura del sistema ha subito delle modifiche che rendono l'applicativo più efficiente per il suo funzionamento nella Google Cloud Platform. Si è deciso infatti che ogni modulo che interrogherà una certa API sarà inserito in una piattaforma web differente (Google App Engine differente). Questi moduli conterranno a sua volta un modulo Miner, un modulo Processor, un Cron e un Database NoSQL come in precedenza. In questo modo, oltre a distribuire il carico di lavoro che può causare il superamento della soglia gratuita del servizio, ogni modulo risulterà più estensibile essendo indipendente dagli altri;
- Con l'aiuto del proponente sono state discusse le prime analisi effettuate sulla ricerca della tipologia di dati da ricavare tramite le API esposte dai vari social network. In particolare modo, sono state individuate le prime metriche che andranno a definire le funzionalità rese disponibili dalle View e che individueranno i dati grezzi necessari alle Recipe;
- Il proponente ha fornito un elenco di servizi web che potrebbero facilitare l'analisi dei dati ricavati dai social network o, in alcuni casi, fornire funzionalità ed informazioni aggiuntive al risultato finale esposto dalle View, grazie alle API esposte da ognuno di essi. Tali servizi sono:
 - **PageSpeed Insights**: servizio fornito da Google per l'analisi della velocità ed il rendimento di un sito web;
 - **MozRank**: fornisce informazioni sull'importanza di un sito web a seconda della qualità pagine che contengono uno o più link a tale sito;
 - **Feedly**: espone delle API che forniscono dati e statistiche sull'uso dei feed all'interno del servizio, inclusi topic, categorie, tag ed interessi popolari tra gli utenti;
 - **Ritetag**: fornisce analisi e statistiche sul trend e la popolarità di determinati hashtag;
 - **Klout**: fornisce analisi sull'influenza generata da parte di una pagina o di un profilo attivo in un social network;
 - **ShareCount**: traccia l'andamento di un URL nei vari social network;
 - **W3C**: fornisce errori sul codice front-end di un determinato sito web;
 - **GTMetrix**: fornisce analisi sulla pesantezza e la velocità di una pagina web.

2.2 Specifica sui moduli individuati

La struttura dell'applicativo è stata aggiornata in modo tale da distribuire lo spazio riservato al database ed il carico di lavoro affidato al sistema. Ogni modulo responsabile di ricavare i dati da un determinato social network sarà infatti inserito in

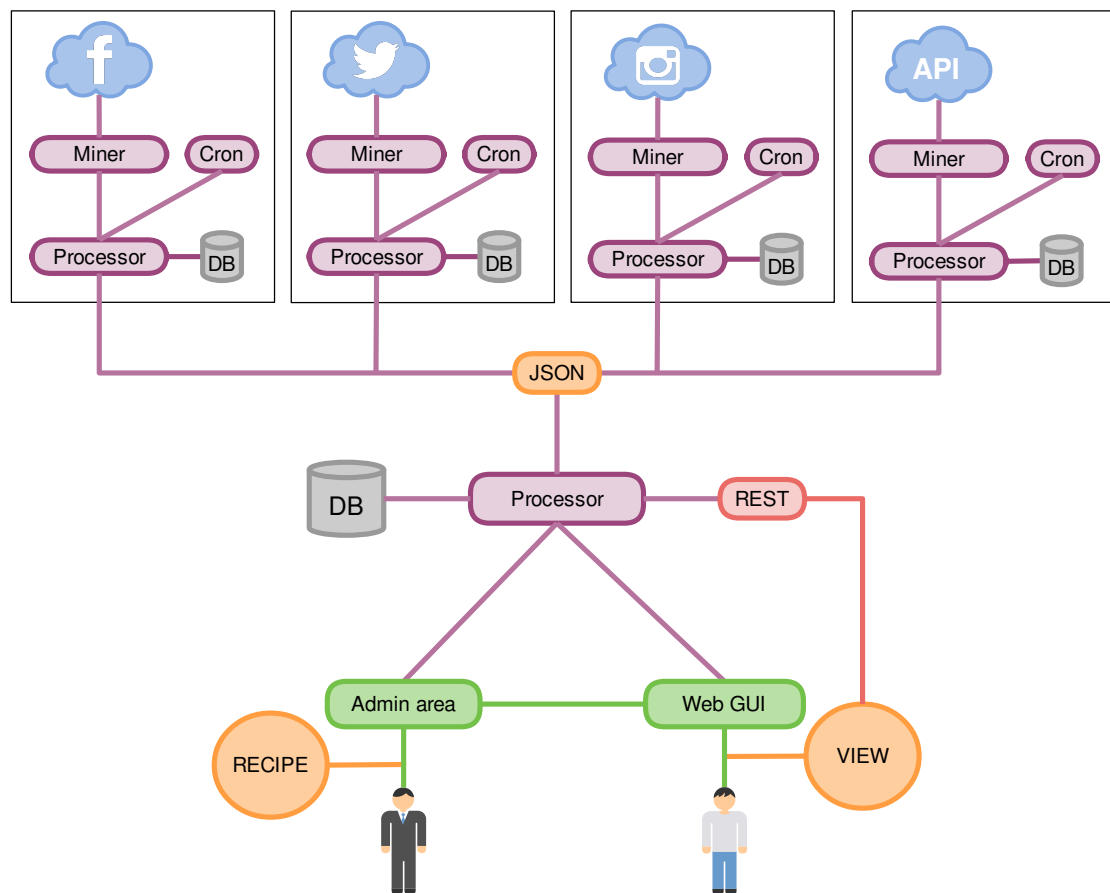


Figura 1: Moduli applicativo

un'infrastruttura differente che richiederà la creazione di più account nella piattaforma di Google.

Ogni modulo, quindi, conterrà la struttura Miner, Cron e Processor stabilita in precedenza ed un database NoSQL che conterrà i dati grezzi. Il vantaggio di questa soluzione è data dal fatto che ogni modulo risulta indipendente dai restanti e potrà essere progettato in modo specifico per le funzionalità che dovrà fornire. Inoltre, sebbene la possibilità di superamento della soglia gratuita del servizio offerto da Google non dovrebbe causare problemi, la distribuzione del sistema su più account renderà tale problematica meno rilevante.

Tutti i moduli saranno gestiti da un Processor centrale, il quale, interagirà con un database NoSQL principale che memorizzerà i dati di configurazione dell'applicativo ed i riferimenti ai dati dei database secondari di ogni modulo sopracitato. Il Processor centrale, inoltre, si occuperà di elaborare i dati grezzi ed esporli alla WebGUI tramite dei servizi di tipo REST.