



dstackgroup@gmail.com

Verbale Interno 2019-03-29

Informazioni sul documento

Nome documento	Verbale Interno 2019-03-29
Data approvazione	2019-03-29
Responsabile	Federico Rispo
Redattore	Eleonora Signor
Verificatore	Niccolò Vettorello
Stato	Approvato
Lista distribuzione	Gruppo <i>DStack</i> <i>Prof. Tullio Vardanega</i> <i>Prof. Riccardo Cardin</i>
Uso	Interno

Sommario

Il presente verbale riporta l'incontro interno effettuato dal gruppo *DStack* il giorno 2019-03-29 presso Torre Tullio Levi Civita via Trieste, 35121 Padova (PD).

Indice

1	Informazioni sulla riunione	1
2	Ordine del giorno	1
3	Resoconto della riunione	2
3.1	Conferma data colloquio <i>Agile</i>	2
3.2	Stesura e-mail da inviare al prof. Riccardo Cardin	2
3.3	Scelta del materiale che si ritiene idoneo da presentare al committente per attestare il raggiungimento della <i>Prododuct Baseline</i>	4
3.4	Tracciamento delle decisioni	5

1 Informazioni sulla riunione

- **Luogo della riunione:** Torre Tullio Levi Civita, aula LabTA
- **Ora di inizio:** 13.30
- **Ora di fine:** 17.30
- **Segretario:** Eleonora Signor
- **Partecipanti:** Tutti i membri del gruppo *DStack*
 - Federico Rispo;
 - Alberto Schiabel;
 - Eleonora Signor;
 - Harwinder Singh;
 - Elton Stafa;
 - Enrico Trinco;
 - Niccolò Vettorello.

2 Ordine del giorno

- Conferma data colloquio *Agile*;
- Stesura e-mail da inviare al prof. Riccardo Cardin;
- Scelta del materiale che si ritiene idoneo da presentare al committente per attestare il raggiungimento della *Prododuct Baseline*.

3 Resoconto della riunione

3.1 Conferma data colloquio *Agile*

A seguito di uno scambio di mail con il prof. Riccardo Cardin in data 2019-03-25, avvenuta con l'uso della mail ufficiale del gruppo, è stata scelta come data ufficiale in cui sostenere il colloquio *Agile* il 2019-04-05 alle ore 12:45. Durante la riunione odierna si sono discusse più nel dettaglio le scelte progettuali decise durante la riunione precedente ed espone nel *Verbale-I-9_2019_03_24*.

3.2 Stesura e-mail da inviare al prof. Riccardo Cardin

Il gruppo *DStack* a seguito di dubbi sorti durante la riunione, in merito alla validità della progettazione adottata per il sistema Butterfly e al materiale da presentare per attestare il raggiungimento della *Product Baseline* durante il colloquio *Agile*, ha deciso di rivolgersi al prof. Riccardo Cardin, nuovamente tramite contatto mail. Durante la suddetta riunione viene stesa una bozza di quanto necessario chiedere al docente, da rifinire ed inviare nei prossimi giorni con priorità immediata, con responsabilità del *Responsabile di Progetto* Federico Rispo.

I punti di interesse riguardano:

- Il vincolo obbligatorio di presentare durante il colloquio con il committente un documento dal nome *Allegato Tecnico*;
- Dubbi riguardanti i design pattern da applicare alla progettazione del sistema Butterfly. Nello specifico riguardanti il design pattern impiegato nell'interazione con il client di terze parti gitlab-4j, in uso per capire il tipo di evento inoltrato da GitLab al Producer dedicato. Nel dettaglio:
"Il client di nostro interesse si aspetta che il chiamante implementi l'interfaccia "WebHookListener", che noi abbiamo implementato in "GitlabWebhookListenerImpl". Tale client emette eventi, che noi adattiamo al nostro contratto pubblico "GitlabWebhookListener" tramite l'implementazione "GitlabWebhookListenerAggregator". Tale "GitlabWebhookListenerAggregator", inoltre, è responsabile di emettere un unico evento a partire da tutti gli eventi definiti nel contratto pubblico "GitlabWebhookListener", per semplificare l'API pubblica esposta al chiamante (che tanto si dovrà solo occupare di spedire l'evento in arrivo al broker, a prescindere dal tipo di evento originale). La nostra idea attuale è mostrata nell'immagine in allegato "gitlab-producer-webhook.png"."
(Qui mostrata sotto)

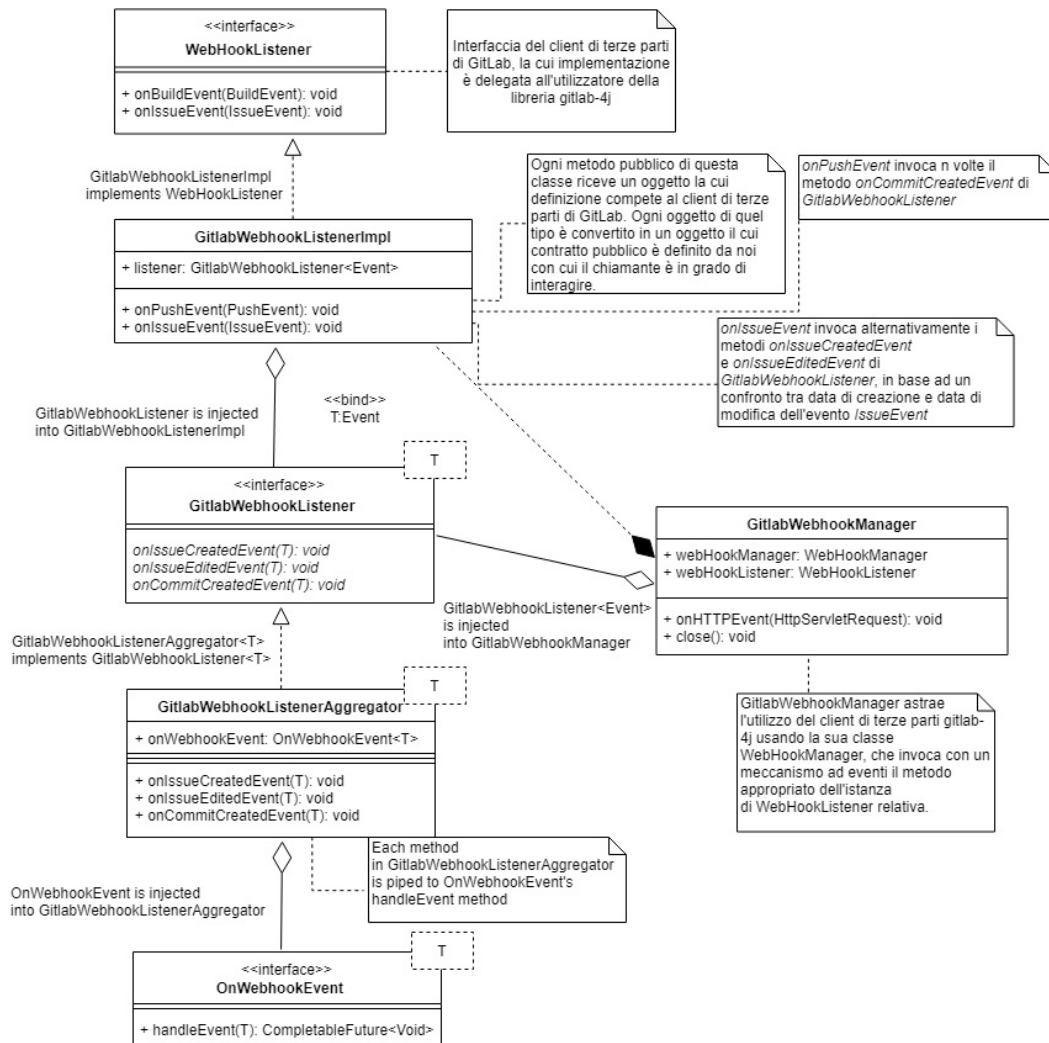


Figura 1: Diagramma delle classi per rappresentare Adapter Pattern e Event-Aggregator Pattern

"Considerati anche i commenti all'interno dell'UML allegato, possiamo dire che "GitlabWebhookListenerImpl", classe annidata in "GitlabWebhookManager", funga da Adapter, in quanto astrae l'utilizzo della dipendenza di terze parti gitlab-4j? O dovremmo limitarci a descriverne il funzionamento senza citare l'Adapter Pattern? Inoltre, riteniamo che "GitlabWebhookListenerAggregator" possa essere descritto dall'Event Aggregator Pattern, poiché riduce tutti i possibili eventi ad un evento unico a cui la classe chiamante può sottoscrivere. Tale intuizione è corretta?";

- La correttezza dell'affermazione nel definire il nostro sistema come:
"Rete di microservizi la cui interazione con l'utente avviene tramite API REST e la cui comunicazione interna al sistema si basa sul Messaging Pattern 'Publish/Subscribe' nella sua accezione topic-based, in cui il message broker intermedio è Apache Kafka". Faremmo quindi riferimento alla "MICROSERVICES ARCHITECTURE" e alla "API REST TOPOLOGY";
- Se esistono preferenze in merito all'impiego della versione UML1 o UML2 ("Lollipop") per rappresentare l'aggregazione tra una classe e un'interfaccia

implementata da un'altra classe.

3.3 Scelta del materiale che si ritiene idoneo da presentare al committente per attestare il raggiungimento della *Prododuct Baseline*

Il gruppo ha ritenuto possibile la stesura di un documento formale esterno l'*Allegato Tecnico v1.0.0*, da inviare al committente una volta terminato il colloquio *Agile*. In attesa di delucidazioni in merito da parte del committente, come indicato nella sezione precedente, si è deciso di stilare una prima bozza dell'indice.

Indice Allegato Tecnico

- **Introduzione:** composta da
 - Scopo del documento;
 - Scopo del prodotto;
 - Glossario;
 - Riferimenti informativi;
 - Riferimenti normativi.
- **Architettura del prodotto:** composta da
 - Descrizione generale dell'architettura che si è deciso di modellare nel progetto Butterfly;
 - Diagramma dei package per rappresentare l'architettura generale;
 - Diagramma di sequenza generale per rappresentare le interazioni che coinvolgono l'intera architettura, dalla ricezioni di un messaggio da parte del Producer all'inoltro del messaggio verso il Consumer appropriato;
 - Breve elenco dei Design Patterns notevoli utilizzati, senza entrare nello specifico;
 - Descrizione dell'architettura in uso nello User Manager;
 - Diagrammi dei package per rappresentare l'architettura dello User Manager;
 - Descrizione dell'architettura nei servizi di produzione e consumo;
 - Descrizione del producer;
 - Diagrammi dei package relativi ai producer;
 - Diagrammi di sequenza che mostrano le interazioni che avvengono da quando GitLab emette un messaggio a quando il producer di competenza lo riceve;
 - Descrizione del consumer;
 - Diagrammi dei package relativi ai consumer;
 - Digrammi di sequenza che mostrano le interazioni che avvengono da quando un consumer specifico riceve la notifica di un evento a quando questo è inviato al bot di Telegram;
 - Design Patterns notevoli utilizzati con spiegazione di Strategy Pattern, Observer Pattern, Command Pattern e Template Method Pattern;
 - Uso dei diagrammi di classe per ogni pattern esposto.

3.4 Tracciamento delle decisioni

Codice Identificativo	Descrizione
VI-2019-03-29-1	Il <i>Responsabile di Progetto</i> Federico Rispo ha l'onere di contattare tramite e-mail il committente prof. Riccardo Cardin a nome di <i>DStack</i> in modo da chiarire i dubbi sorti durante la suddetta riunione
VI-2019-03-29-2	Si è scelto l'indice generale dell' <i>Allegato Tecnico</i> , la cui utilità deve venire confermata dal committente

Tabella 2: Decisioni Riunione Interna del 2019-03-29