

Nuove features in Confluence

Nuove features in Confluence

In questo post andremo ad esaminare una piccola novità di Confluence, che aiuta molto gli utilizzatori nello svolgere il proprio lavoro.



Resizable table columns in Confluence

Una delle ultime caratteristiche introdotte, già presenti sulla versione Cloud, è la possibilità di poter inserire delle tavole ridimensionabili, come mostrato dalla seguente GIF animata:



Abbiamo la possibilità di poter disporre di tavole a dimensione fissa o responsive :-). Usando le tavole tutti i giorni, come utente, posso assicurarvi che è una feature che aspettavo da un pò di tempo.

Avendo a disposizione una istanza cloud, che uso regolarmente, l'uso di tavole responsive è UTILE :-D.

Conclusioni

Una piccola novità, un ottimo aiuto per noi utenti di Confluence. Restiamo in attesa di ulteriori ... sorprese.

Reference

Maggiori informazioni sono reperibili [qui](#).

Novità su JIRA Software

Ultime novità su JIRA Software

Prosegue la serie di post dedicati alle ultime novità dei prodotti della Atlassian. Ci occupiamo adesso di JIRA Software.



Cosa abbiamo di nuovo?

Iniziamo con la prima schermata che viene visualizzata all'inizio di tutte le attività è la seguente:



Tutte le prime schermate che venivano proposte, sono state unificate nella precedente schermata, che risulta di più agevole e rapida nell'utilizzo.



Per dare maggiore flessibilità al team Agile, viene messa a disposizione la possibilità di poter gestire e delegare le responsabilità delle Sprint. Di conseguenza, è stata introdotta una nuova permission: **manage sprint**; come mostrato nella precedente Gif.

Questo ovviamente presuppone una grande fiducia ed un team molto affiatato, altrimenti questa opzione potrebbe generare delle situazioni molto inconsistenti.



Altra novità è quella di **Flag and comment** una issue contemporaneamente. In questo modo è possibile marcarla come importante (Flag) e successivamente inserire un commento. Si tratta di una piccola feature, ma utile per coloro che lavorano tutti i giorni su questi punti.

Conclusioni

Si tratta di piccole modifiche, ma che semplificano notevolmente il lavoro.

References

Maggiori informazioni sono disponibili [qui](#).

Evento Atlassian a Bologna

Evento Atlassian a Bologna



Maggiori informazioni sono reperibili [qui](#).

Webinar su Atlassian

Webinar su Atlassian



Segnalo il seguente webinar su Atlassian JIRA Service Desk da parte di [ibuildings](#). Maggiori informazioni sono presenti [qui](#).

Component & Subcomponents – Alternative

Quali alternative esistono?

In questo post andremo ad esaminare quali alternative sono disponibili per l'addon descritto [qui](#). Stiamo parlato di [Component/Subcomponent/Bundle Versions](#).



Che cosa offre?

Questo addon mette a sua volta a disposizione la possibilità di poter inserire una gerarchia di componenti



direttamente dallo screen del dettaglio in maniera molto agevole. In aggiunta abbiamo anche la possibilità di associare delle componenti a specifiche versioni



in maniera molto semplice, da quanto mostra la scheda del Marketplace



Inoltre abbiamo la possibilità di inserire nei Workflow Validator la possibilità di poter eseguire dei controlli sulle versioni da associare in fase di risoluzione di una issue



In aggiunta abbiamo una estensione del JQL per poter referenziare queste informazioni



Conclusioni

L'addon si presenta con delle funzionalità di tutto rispetto. Sono molto interessanti e consentono di poter implementare delle features molto carine.

References

Maggiori informazioni sono reperibili [qui](#).

Component & Subcomponents – Test su strada

Test su strada

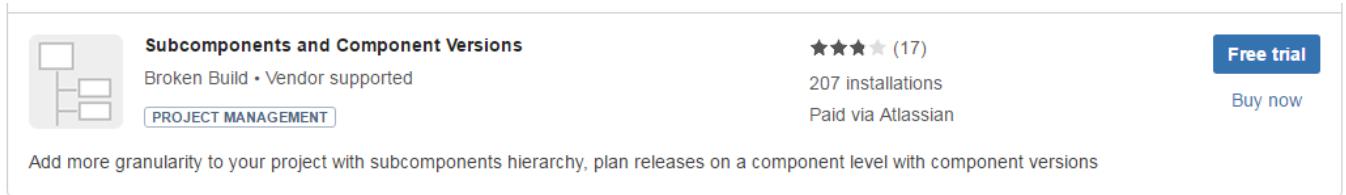
In questo post andremo a fare il test di questo addon, cercando di trovare ciò che ci serve.



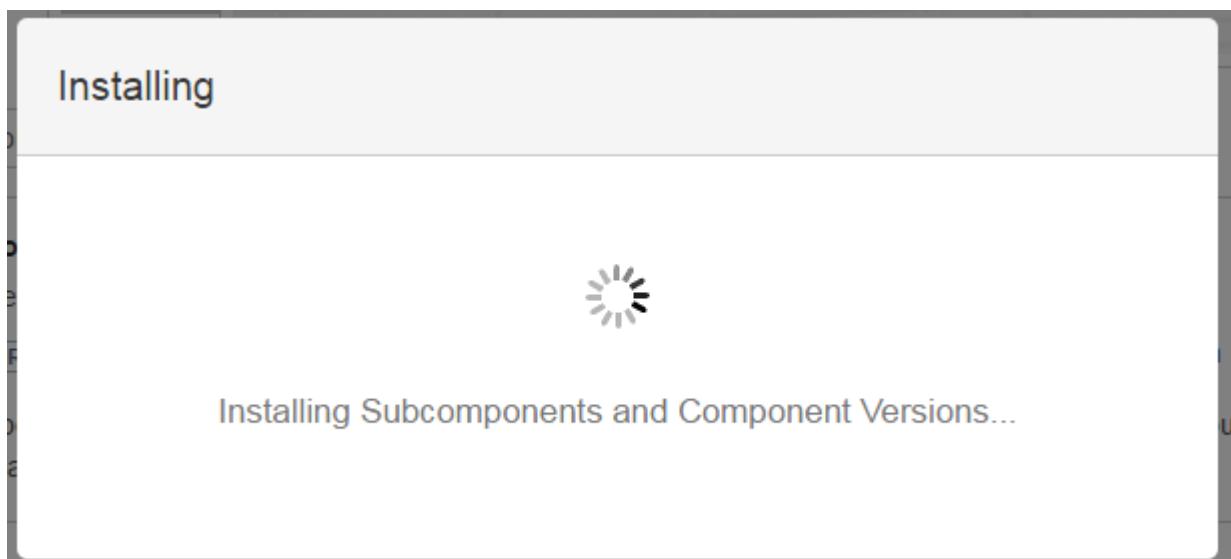
Installazione

Partiamo sempre dalla installazione del nostro addon, che come sempre non ci deve spaventare :-). In questo caso andremo ad utilizzare una installazione di **JIRA CORE**. Selezioniamo il

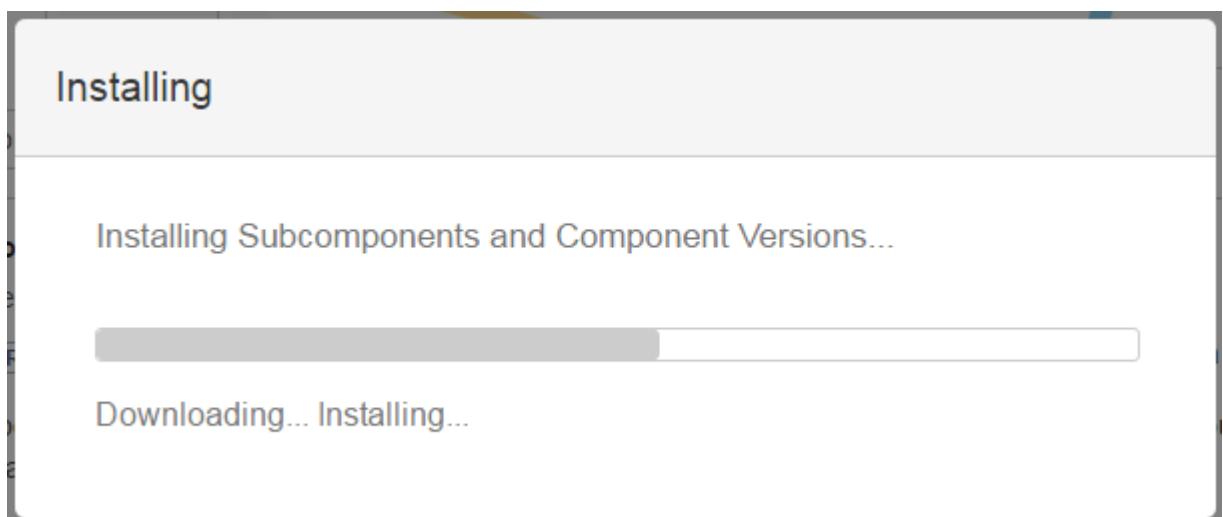
nostro addon e procediamo, selezionando **Free trial** dalla nostra schermata di **Manage Addons**.



Lasciamo andare la nostra installazione...



... facciamo eseguire il download e la relativa installazione ...



... inseriamo le credenziali per accedere al **my.atlassian.com** ...

Atlassian account login

Log in with your Atlassian account to start your trial.

Email*

Password*

[Lost password](#)

[Log in](#)

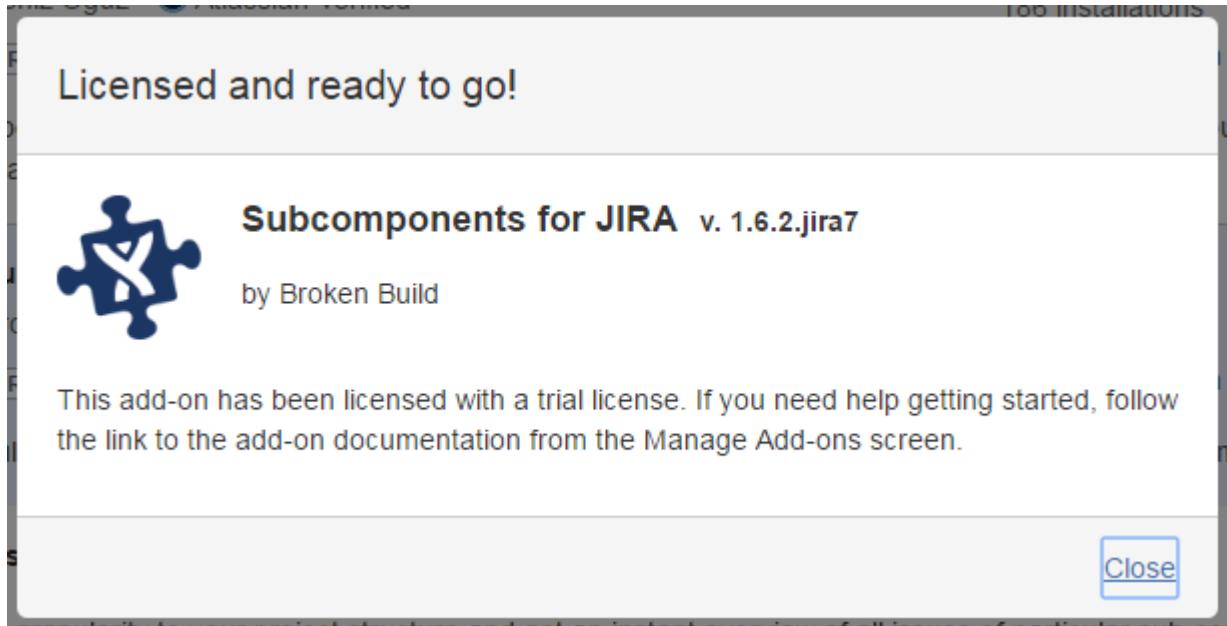
[Cancel](#)

... lasciamo che installi la licenza ...

Free trial

Storing trial license...

... ed il nostro addon è pronto all'uso.



Configurazione

Passiamo alla fase successiva: Come configuriamo questo addon?? La prima cosa che notiamo è che non è presente alcuna sezione di configurazione generale, nella sezione addon. Troviamo la sezione di configurazione tra le opzioni di configurazione del progetto, trattandosi di una configurazione specifica di un progetto.

 **Esempio** Key: ES · Lead:  JIRA Administrator · Pr

[Back to project](#) [Administration](#)

Summary

[Issue types](#)
 └ [Sub-task](#)
 └ [Task](#)

[Workflows](#)

[Screens](#)

[Fields](#)

[Versions](#)

[Components](#)

[Subcomponents](#)

[Component Versions](#)

 **Issue**

Keep tra
type can

Scheme:
ES: Task

Task

Sub-t

 **Workf**

Issues c;
defines t
"Resolve"

Scheme:

Tentiamo una prima configurazione sul nostro progetto di **Esempio**. Supponiamo di avere delle componenti che identificano il nostro sistema da gestire.



Esempio

Key: ES · Lead: JIRA Administrator · Project type: Business · Category: None · URL: No URL

[Back to project](#)

Administration

[Summary](#)

[Issue types](#)

Sub-task

Task

[Workflows](#)

[Screens](#)

[Fields](#)

[Versions](#)

Components

[Subcomponents](#)

[Component Versions](#)

[Users and roles](#)

[Permissions](#)



Components

Projects can be broken down into components, e.g. "Database", "User Inte

	Name	Description	Cor
	<input type="text"/>		
	Supporto	richieste di supporto / manutenzione	
	Backend - JAVA	Definizione del backend Java	
	Database - PLSQL	Definizione delle Stored procedure	
	Database - Struttura	Definizione della struttura del DB	

Supponiamo di avere la necessità di distinguere le richieste di supporto, quali segnalazioni utente o segnalazione divisione interna, etc. Se aggiungiamo nuove componenti, potremmo avere delle difficoltà. Il numero delle componenti si moltiplicherebbe e alla fine si avrebbe difficoltà nel censire ed usare, nonché nell'identificare.

Creiamo le nostre sottocategorie selezionando l'opzione **Subcomponent**. Viene mostrata la form per compilare la nostra gerarchia.

Back to project Administration

Summary

Issue types

Sub-task

Task

Workflows

Screens

Fields

Versions

Components

Subcomponents

Component Versions

Subcomponents

Organize your projects better with subcomponents hierarchy, assign components to issues via subcomponents field picker, search issues using subcomponents() JQL.

Available Components

New component... Add Filter...

Components Hierarchy

Component: Supporto

Issues Versions

Backend - JAVA

Database - PLSQL

Database - Struttura

Webapp

Supporto

Client

Consulenza

Interno

Sistemistica

There are no issues in current component and it's subcomponents. Use subcomponents field picker on issue screen to select components from hierarchy.

Basta un semplice drag 'b' drop per costruire la gerarchia. Possiamo anche estendere direttamente le nostre componenti e agevolmente nella gerarchia.

Ma le *componenti*?

Che cosa succede alle nostre componenti? Come sono organizzate? Fondamentalmente l'addon si innesta nel nostro JIRA e gestisce come un ulteriore meccanismo che inserisce la gerarchia e la gestisce. Infatti, se andiamo a vedere la componenti, vediamo che sono aumentate, come mostrato in figura.

	<input type="text"/>	<input type="text"/>	<input type="text"/>	Project Default (Unassigned)	<button>Add</button>
	Backend - JAVA	Definizione del backend Java	Project Default	<button>Delete</button>	
	Cliente		Project Default	<button>Delete</button>	
	Consulenza		Project Default	<button>Delete</button>	
	Database - PLSQL	Definizione delle Stored procedure	Project Default	<button>Delete</button>	
	Database - Struttura	Definizione della struttura del DB	Project Default	<button>Delete</button>	
	Interno		Project Default	<button>Delete</button>	
	Sistemistica		Project Default	<button>Delete</button>	
	Supporto	richieste di supporto / manutenzione	Project Default	<button>Delete</button>	
	Webapp	Front end web	Project Default	<button>Delete</button>	

Come usiamo questa gerarchia?

Adesso vediamo come usarla. Creiamo una nuova Issue e andiamo ad inserire la nostra gerarchia di componenti

Open Issues Switch filter ▾

Order by Priority ▾

- ES-7
Questo è un esempio
- ES-4
Esempio 4
- ES-6
esempio 6
- ES-5
esempio 5
- ES-3
Esempio 3
- ES-2
Esempio 2
- ES-1

Esempio / ES-7

Questo è un esempio

[Edit](#) [Comment](#) [Assign](#) [More ▾](#) [Done](#) [Admin ▾](#)

Details

Type:	<input checked="" type="checkbox"/> Task
Status:	TO DO (View Workflow)
Priority:	↑ Medium
Resolution:	Unresolved
Component/s:	<input type="text"/>
Labels:	None Add Edit Delete

Description

Task di esempio

Se andiamo ad editare il campo **Component/s** notiamo, come mostrato dalla figura precedente, una nuova icona che indica la gerarchia.

Subcomponents:

Filter... [x](#)

- ▼ Supporto
 - Cliente
 - Consulenza
 - Interno
 - Sistemistica

Use context menu for smart selection [Apply](#) [Cancel](#)

Se lo selezioniamo possiamo andare a selezionare le nostre componenti.

Subcomponents:

- ▼  **Supporto**
-  Cliente
- ▼  **Consulenza**
-  Interno
-  Sistemistica

Use context menu for smart selection

Se andiamo a confermare, il risultato è il seguente:

Open Issues Switch filter ▾

Order by Priority ▾
<input checked="" type="checkbox"/> ES-7 Questo è un esempio
<input checked="" type="checkbox"/> ES-4 Esempio 4
<input checked="" type="checkbox"/> ES-6 esempio 6
<input checked="" type="checkbox"/> ES-5 esempio 5
<input checked="" type="checkbox"/> ES-3 Esempio 3
<input checked="" type="checkbox"/> ES-2 - - -

 Esempio / ES-7
Questo è un esempio

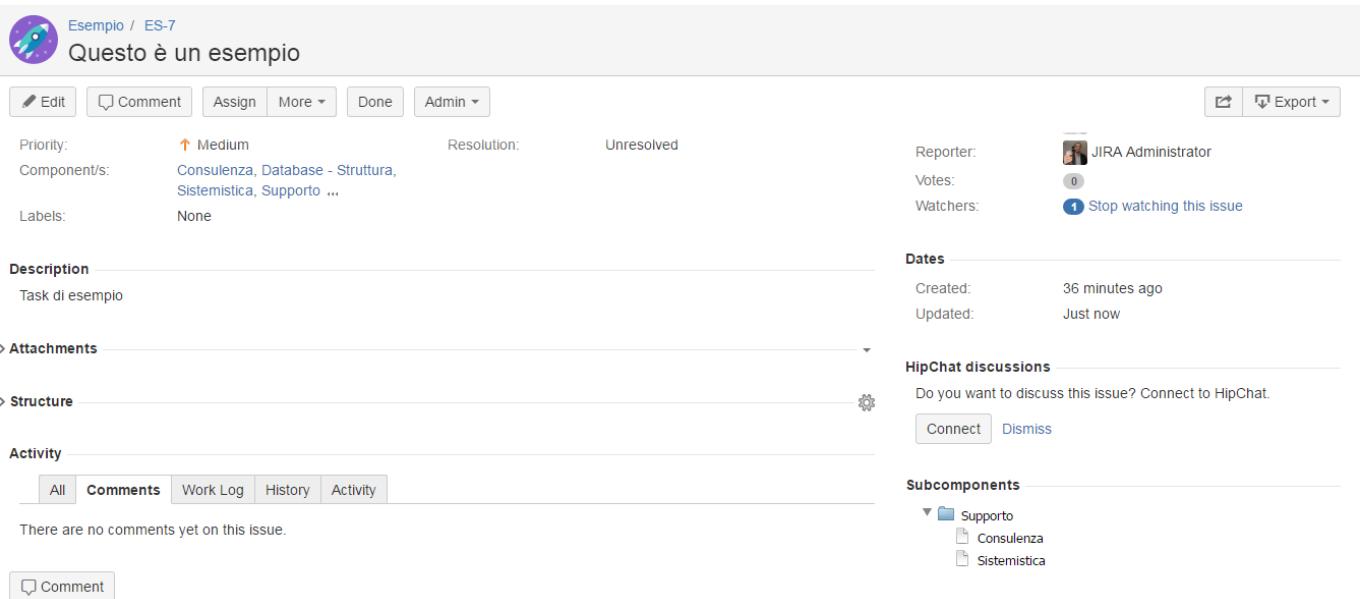
Details

Type:	<input checked="" type="checkbox"/> Task
Status:	TO DO (View Workflow)
Priority:	↑ Medium
Resolution:	Unresolved
Component/s:	Consulenza, Supporto
Labels:	None

ovvero vediamo le due componenti presenti, ma se andiamo in edizione del campo, abbiamo comunque la visione della gerarchia.

Che altro?

Se andiamo nella screen di dettaglio, possiamo vedere che in basso a destra abbiamo il dettaglio delle subcomponents.



The screenshot shows a JIRA issue detail page for an example issue (ES-7). The page includes fields for Priority (Medium), Component/s (Consulenza, Database - Struttura, Sistemistica, Supporto ...), Resolution (Unresolved), Reporter (JIRA Administrator), Votes (0), Watchers (1), and Dates (Created: 36 minutes ago, Updated: Just now). The Activity section shows no comments yet. The Subcomponents section lists 'Supporto', 'Consulenza', and 'Sistemistica' under a collapsed 'Supporto' category. The page also features sections for Attachments, Structure, and HipChat discussions.

Conclusioni

Abbiamo visto un esempio di utilizzo di questo addon. Abbiamo visto che cosa fa e come lavora. Sicuramente ci sarà utile per meglio classificare le nostre componenti. Questo componente è disponibile sia per installazioni **Server** che per le installazioni **cloud**.

References

Maggiori informazioni sono reperibili su:

- [Server](#)
- [cloud](#)

Ultime novità su Confluence

Ultime novità

In questo post andremo a visionare alcune tra le ultime novità di Confluence.

In dettaglio... per gli amministratori

Segnaliamo alcune novità su Audit Log per gli amministratori, dove adesso è possibile vedere le variazioni eseguite anche a livello di permission per singolo utente/gruppo, come mostrato dalla seguente immagine.

Audit Log

The audit log gives you a history of changes to your Confluence site. It can be very useful for tracking down things like permissions, global settings or add-on changes.

Filter by keyword Filter by Time: Within the last 10 minutes Export Settings

Time	User	Event type	Change	Item affected	Actions
3 Feb, 2016 10:37:03	Josh User	Permissions	Space permission added	Group: confluence-users	Show more
3 Feb, 2016 10:36:06	System	Users and groups	Group deleted	Group: developers	Show more
3 Feb, 2016 10:35:53	System	Users and groups	User added to group	Group: administrators	Show more
3 Feb, 2016 10:32:00	Administrator	Global Administration	Color scheme modified		Show more
3 Feb, 2016 10:32:00	Administrator	Global Administration	Site logo changed		Show more
3 Feb, 2016 10:31:57	Administrator	Global Administration	Favicon changed		Show more
3 Feb, 2016 10:31:44	Administrator	Global Administration	Global settings changed		Show more
3 Feb, 2016 10:31:36	Administrator	Global Administration	Global settings changed		Show more

Filter it
Dig into the log by keyword or by time

More control
Export the whole log or change how long to keep events

Get detailed
See the details of each change

Viene inoltre segnalata una importante novità: gli amministratori possono gestire a livello di space i watcher. Prima era possibile gestire solo per singola page. Questo apre il passo per una importante novità: Una gestione centralizzata di queste funzioni ☐

Si segnala anche importanti novità per quanto riguarda i file audio/video: La macro Multimedia riesce ad usare i nuovi tag HTML5 `<video>` ed `<audio>`. Viene inoltre consentito anche la possibilità di poter gestire (anche in anteprima) i file formato Mp4 e Mp3.

Punto di attenzione, ma per la nuova versione 6.0: Sarà rimosso il documentation Theme.

Conclusioni

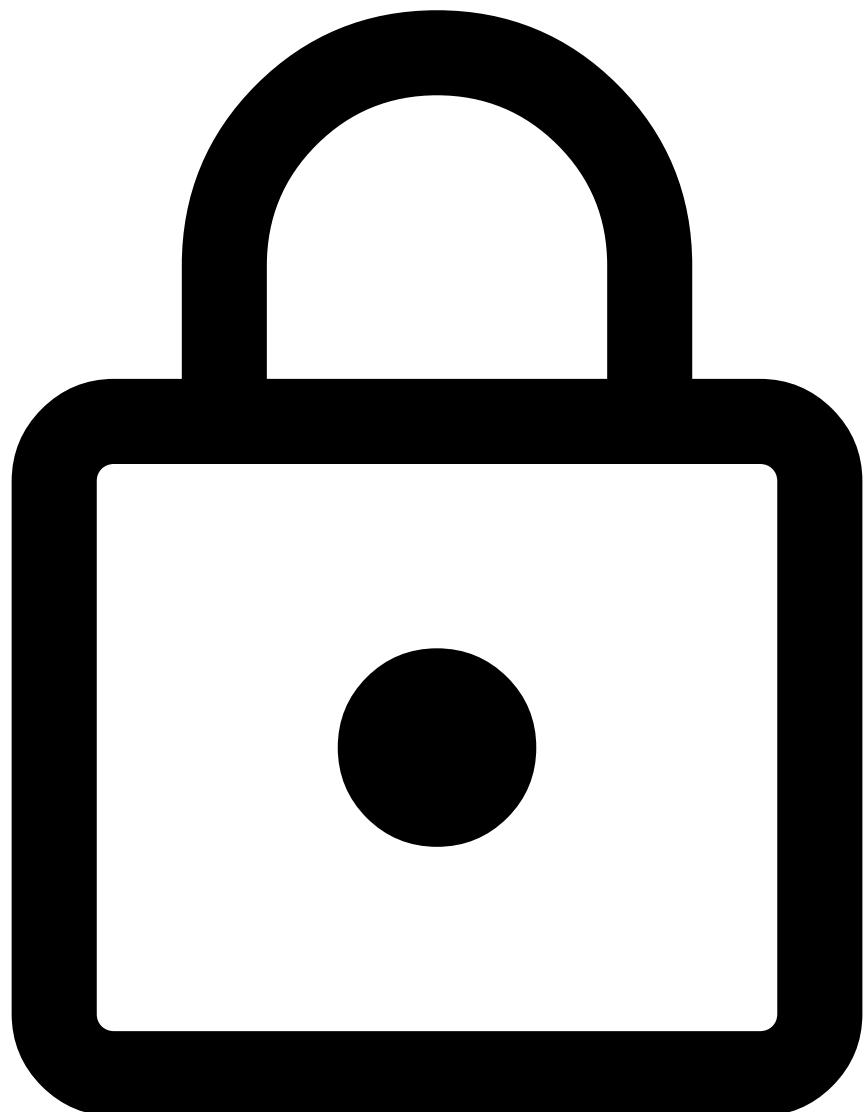
Importanti novità sono previste per la versione 6.0 ma queste ultime segnalate non fanno altro che accrescere la nostra attesa.

Reference

Maggiori informazioni disponibili [qui](#).

Atlassian a Milano

Evento Atlassian a Milano



Maggiori informazioni sono disponibili [qui](#).

Gerarchia di Versioni in JIRA – First look

Altro addon interessante

In questo post andremo a vedere un altro addon molto interessante, prodotto sempre da [Broken Build](#). Abbiamo già visto un altro addon (T0 D0: Inserire il link).



Che cosa offre?

Questo addon permette di poter creare delle gerarchie di versioni, molto utili nello sviluppo di software.



Come notiamo, abbiamo lo stesso layout dell'addon già esaminato (T0 D0: Inserire il link al primo post).



Si integra molto con le funzionalità standard di JIRA, come possiamo ben vedere □



Vediamo che questo addon ci permette di poter generare delle gerarchie di versioni.



Possiamo anche eseguire le ricerche attraverso JQL □

Conclusioni

Abbiamo visto un addon veramente interessante, soprattutto per coloro che sviluppano software, in particolare per organizzare meglio il tutto.

Components & Subcomponents – First look

Estendiamo le componenti

In questo post andremo ad esaminare un nuovo addon, che permette di poter gestire meglio le componenti, aggiungendo una marcia in più: **una gerarchia**.



First look



Questo addon permette di poter estendere la piatta lista delle componenti, consentendo di poter creare delle gerarchie di componenti e sottocomponenti.



Riusciamo facilmente ad associare le sottocomponenti attraverso una treeview molto semplice da usare.



Riusciamo anche ad eseguire delle ricerche per sottocomponente ☐



La stessa view della issue viene estesa in maniera opportuna, consentendo la visualizzazione di entrambe le nuove informazioni.



Queste informazioni sono facilmente fruibili anche sotto **JIRA Service Desk** .



Conclusione

Abbiamo scoperto un nuovo addon che permette di poter gestire in maniera più attinente alla realtà le componenti di un progetto. Possiamo, grazie a lui, creare delle gerarchie e costruire meglio le nostre issue. Nei prossimi post andremo a testarlo sul campo e vedremo anche meglio le caratteristiche.

Reference

Maggiori informazioni sull'addon sono presenti [qui](#).