

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](#).

Component & Subcomponents – Alternative

Quali alternative esistono?

In questo post andremo ad esaminare quali alternative sono disponibili per l'addon descritto [qui](#). Stiamo parlando 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**.



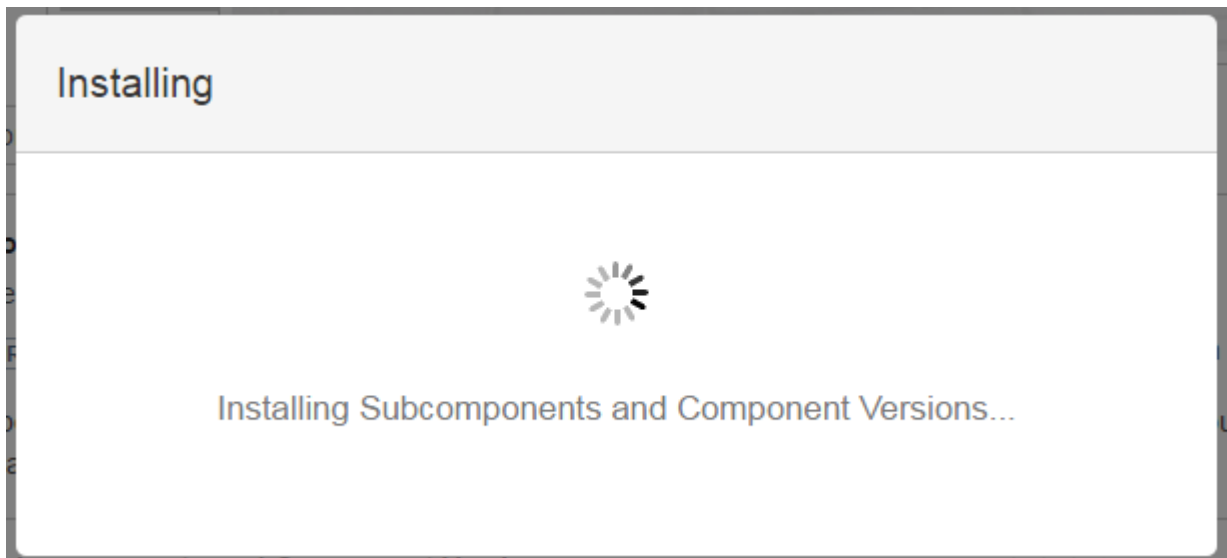
Subcomponents and Component Versions
Broken Build • Vendor supported
[PROJECT MANAGEMENT](#)

★★★★☆ (17)
207 installations
Paid via Atlassian

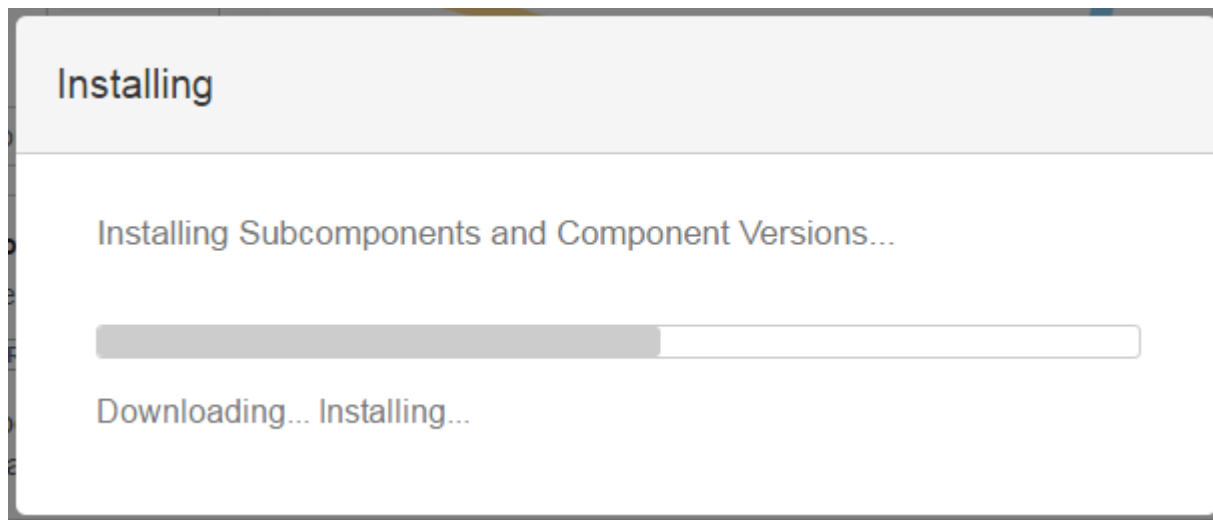
[Free trial](#)
[Buy now](#)

Add more granularity to your project with subcomponents hierarchy, plan releases on a component level with component versions

Lasciamo andare la nostra installazione...



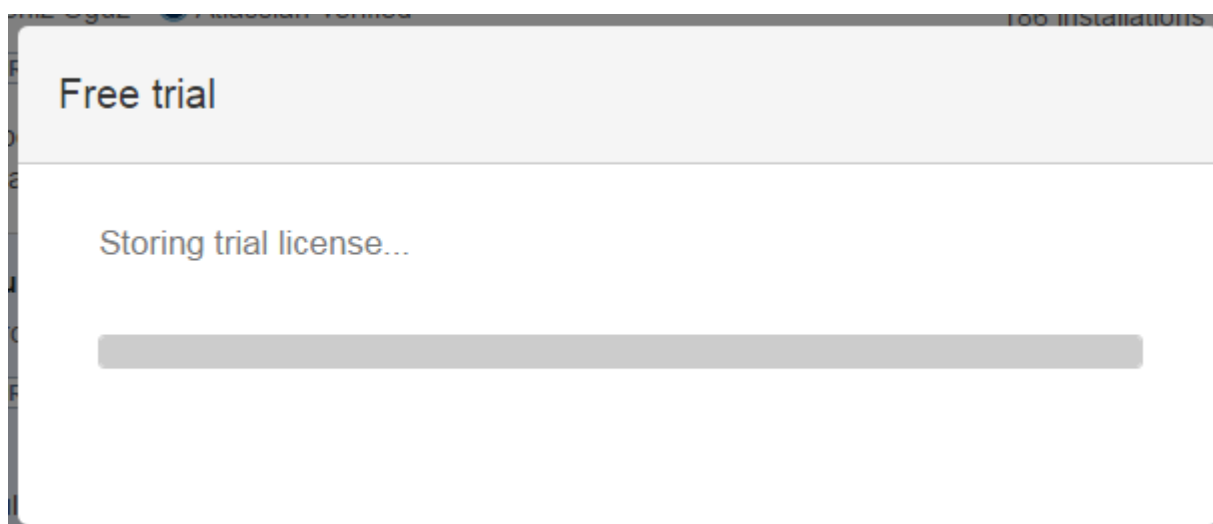
... facciamo eseguire il download e la relativa installazione ...



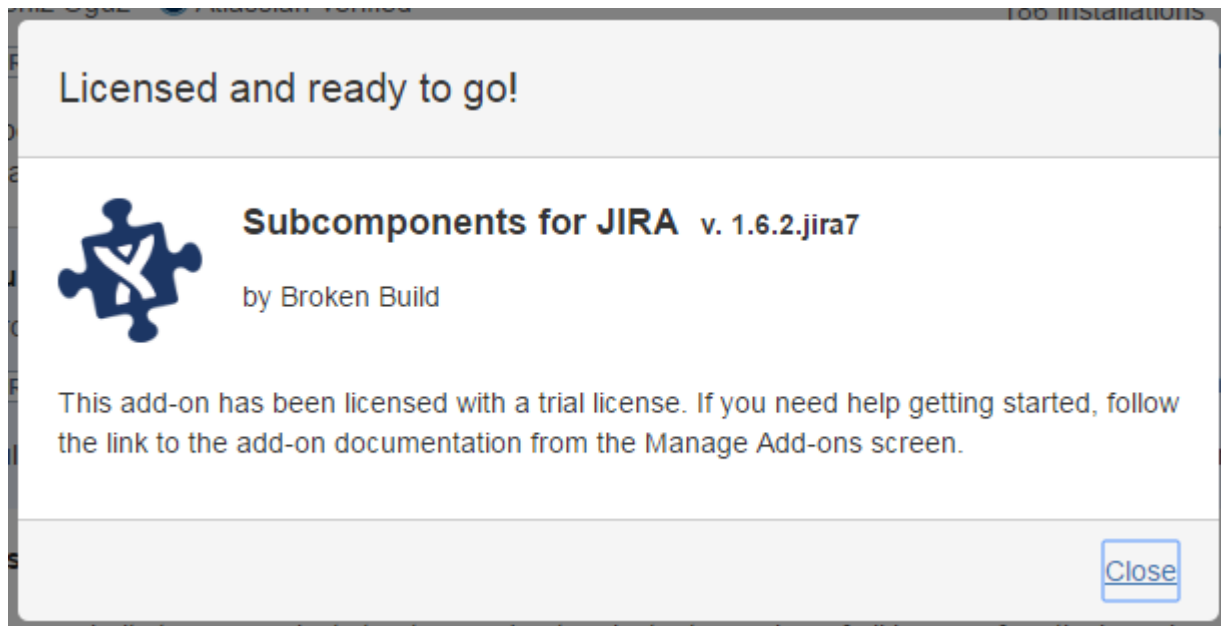
... inseriamo le credenziali per accedere al *my.atlassian.com* ...

A screenshot of the "Atlassian account login" window. The title bar reads "Atlassian account login". Below the title bar, the text "Log in with your Atlassian account to start your trial." is displayed. There are two input fields: "Email*" and "Password*", both with asterisks indicating they are required. To the right of the "Password*" field is a link that says "Lost password". At the bottom right of the window, there are two buttons: "Log in" (in blue) and "Cancel" (in grey).

... lasciamo che installi la licenza ...



... 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: Tas

☒ Task

☐ Sub-t



Workfl

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](#)



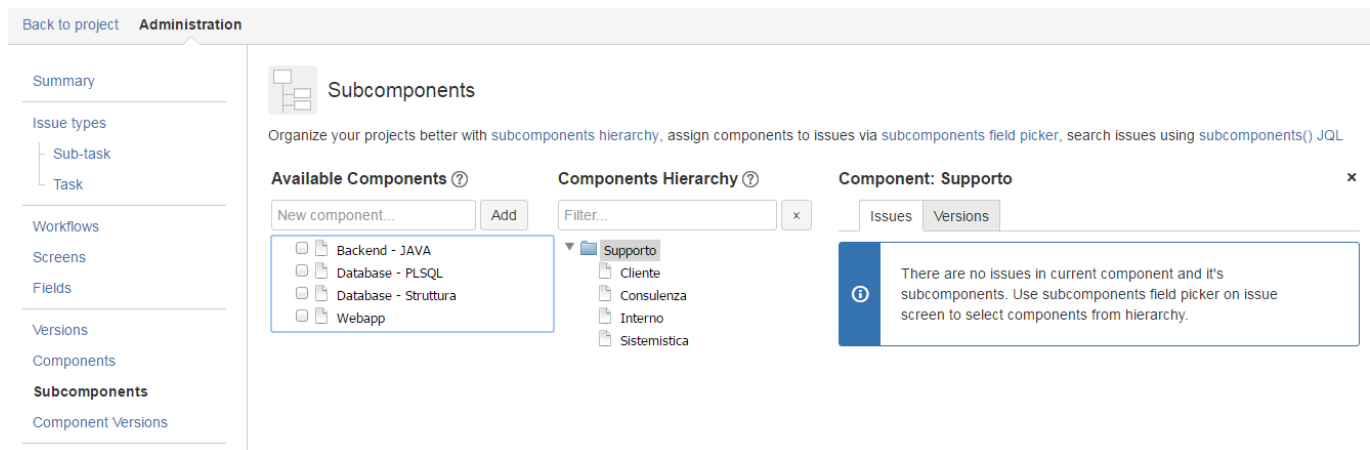
Components

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

	Name	Description	Cor
			
	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.



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.

				Project Default (Unassigne ▾)	Add
	Backend - JAVA	Definizione del backend Java		Project Default	Delete
	Cliente			Project Default	Delete
	Consulenza			Project Default	Delete
	Database - PLSQL	Definizione delle Stored procedure		Project Default	Delete
	Database - Struttura	Definizione della struttura del DB		Project Default	Delete
	Interno			Project Default	Delete
	Sistemistica			Project Default	Delete
	Supporto	richieste di supporto / manutenzione		Project Default	Delete
	Webapp	Front end web		Project Default	Delete

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 1

 Esempio / ES-7

Questo è un esempio

Edit

Comment

Assign

More ▾

Done

Admin ▾

Details

Type: ☒ Task

Status: **TO DO** (View Workflow)

Priority: ↑ Medium

Resolution: Unresolved

Component/s:

Labels: None

✓

✕

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: ✕

▼

☐ Supporto

☐ Cliente

☐ Consulenza

☐ Interno

☐ Sistemistica

Use context menu for smart selection

Apply

Cancel

Se lo selezioniamo possiamo andare a selezionare le nostre componenti.

Subcomponents:

Filter...

x

☒

Supporto

☐

Cliente

☒

Consulenza

☐

Interno

☐

Sistemistica

Use context menu for smart selection

Apply

Cancel

Se andiamo a confermare, il risultato è il seguente:

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 / ES-7

Questo è un esempio

Edit

Comment

Assign

More

Done

Admin

Details

Type:

☒ 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 editazione 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.

Esempio / ES-7
Questo è un esempio

Edit Comment Assign More Done Admin Export

Priority: Medium Resolution: Unresolved
Component/s: Consulenza, Database - Struttura, Sistemistica, Supporto ...
Labels: None

Reporter: JIRA Administrator
Votes: 0
Watchers: 1 Stop watching this issue

Description
Task di esempio

Attachments

Structure

Activity

All Comments Work Log History Activity

There are no comments yet on this issue.

Comment

Dates
Created: 36 minutes ago
Updated: Just now

HipChat discussions
Do you want to discuss this issue? Connect to HipChat.
Connect Dismiss

Subcomponents
Supporto
Consulenza
Sistemistica

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.

Filter it

Dig into the log by keyword or by time

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

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 antepriama) 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.

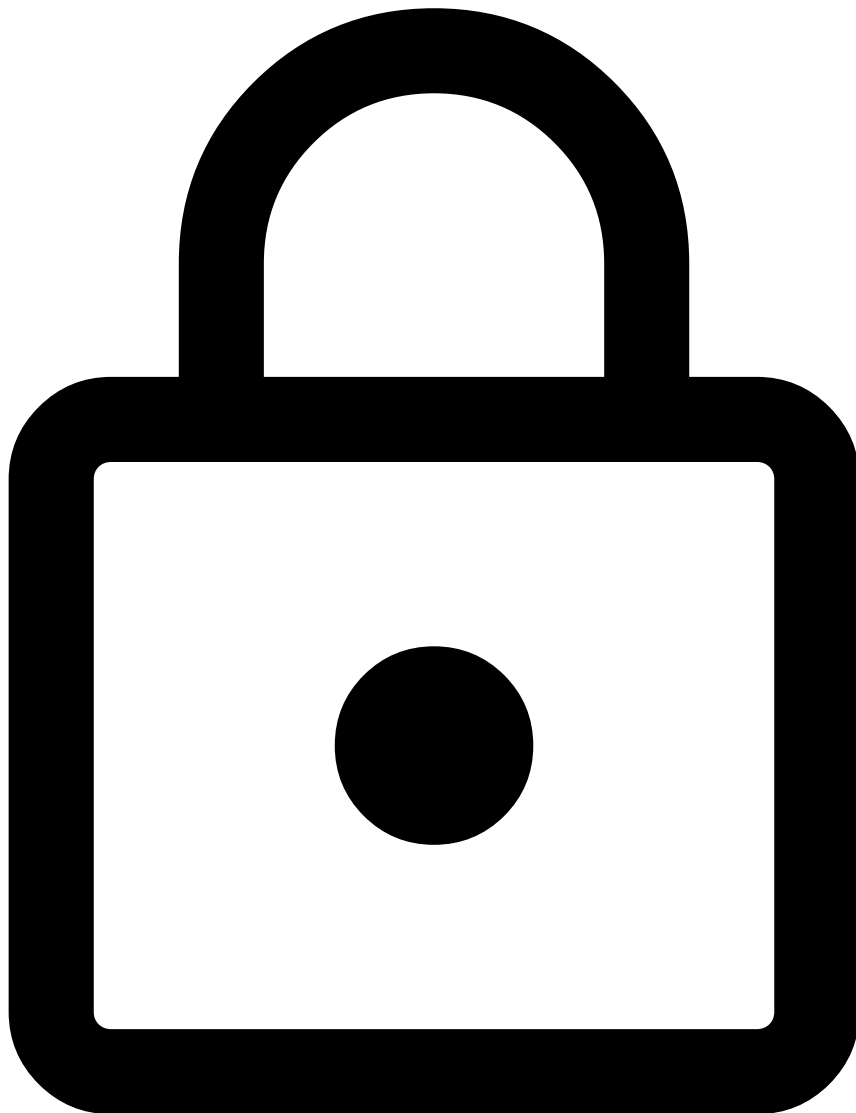
Reference

Maggiori informazioni disponibili [qui](#).

19 Maggio 2016 – Evento

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](#).

Una piccola review di addon per JIRA

Una piccola review

In questo post andremo ad esaminare alcuni addons per JIRA.

Iniziamo...



Partiamo con il primo addon: [Enlarge Your Excel Export](#), un addon che ci aiuta a superare un limite che ci tocca quando andiamo a eseguire una esportazione su foglio elettronico.

Nelle grandi istanze di JIRA, gli amministratori possono limitare la visualizzazione delle issue. Per esperienza posso affermare che molto spesso queste limitazioni aiutano gli amministratori ma non aiutano affatto gli utenti che (ebbene si lo ammetto), vogliono sempre estrarre tutti i dati per fare poi loro le elucubrazioni ☐



molto ci aiuta.

Proseguiamo con il secondo addon [New Status Colors](#), che permette di creare un effetto grafico interessante. Questo può essere di aiuto quando abbiamo bisogno di un colpo d'occhio sulle nostre Issue. Questo addon permette di aggiungere una nota di colore che



basta semplicemente creare degli stati con un nome specifico.
Ad esempio:

- **[red]** per ***Cancel*** oppure ***Cancelled***, ***Suspend*** oppure ***Suspended***, ***Reject*** oppure ***Rejected***
- **[orange]** per ***Wait*** oppure ***Waiting***, ***Hold***, ***On Hold***, ***Pause*** o ***Paused***



Proseguiamo con [Standalone Project Template](#), un addon che ci può sicuramente aiutare nella creazione di nuovi progetti. Consente, infatti, di poter generare dei progetti basati su determinati permission scheme template.



ATLASSIAN

La creazione del progetto NON cambia rispetto allo standard, ma si innesta perfettamente nel flusso di lavoro di JIRA.



ATLASSIAN



Completiamo il nostro giro con [Issue Card Printer for JIRA](#), che permette di poter stampare delle card dalla nostra AGILE BOARD



Conclusioni

Abbiamo fatto un piccolo tour su degli addon, per installazioni server, che aggiungono delle nuove funzionalità per JIRA. Si tratta di funzionalità che aiutano tantissimo nell'uso di tutti i giorni. Ve li consiglio ☐

Issue Organizer – Prova su strada 2

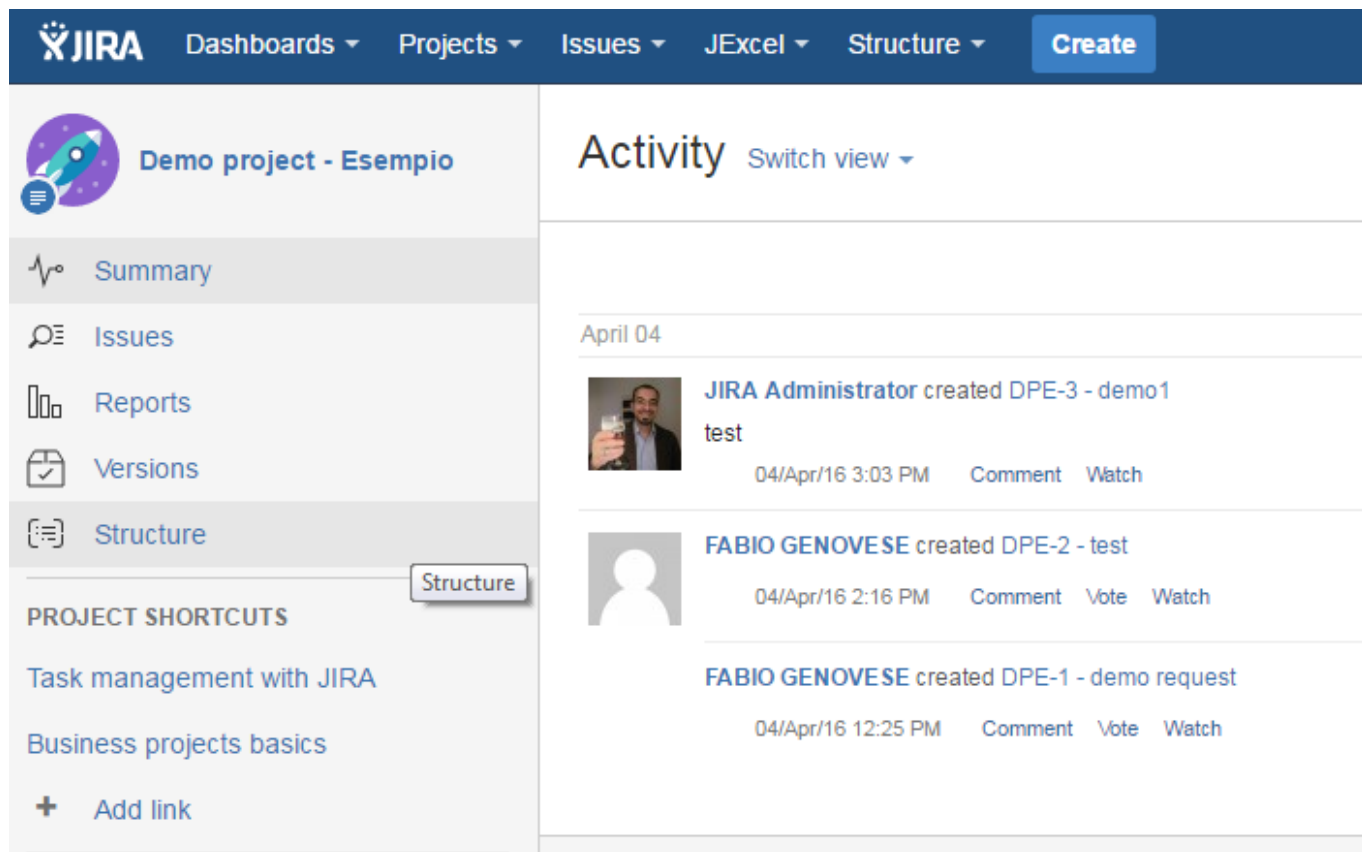
Prosegue la prova

In questo post proseguiamo il test dell'addon [Issue Organizer](#), andando a completare quanto iniziato nel precedente post.



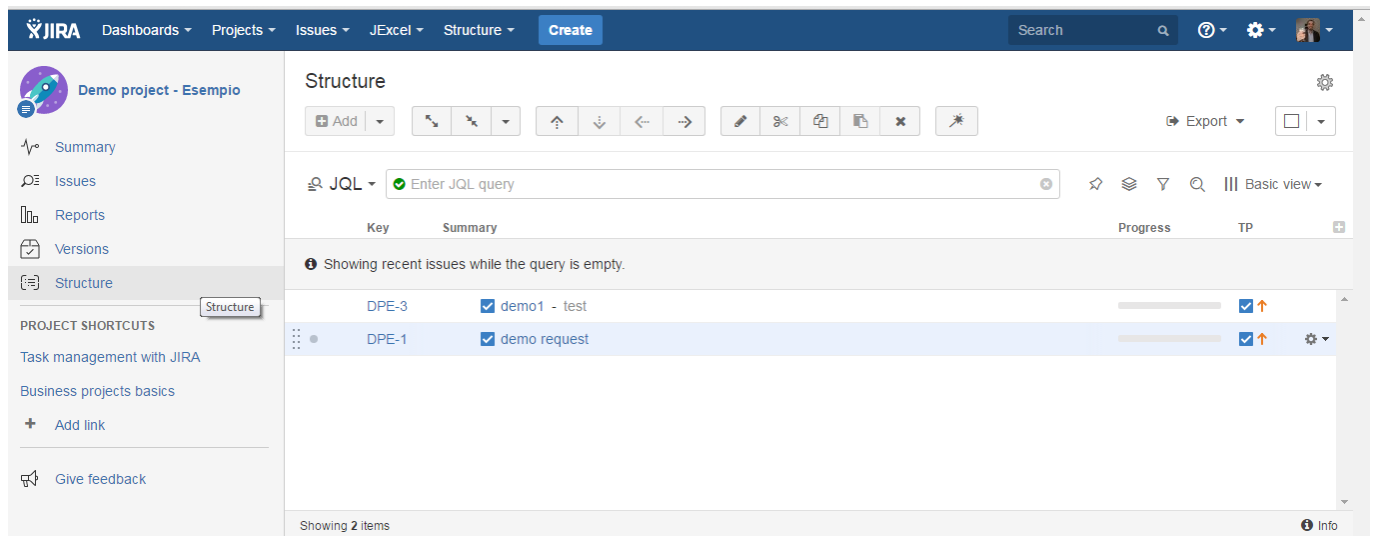
Prova

Dopo aver visionato la configurazione dell'addon, proseguiamo con le operazioni di test vero e proprio. La prima cosa che notiamo è che nella toolbar laterale della schermata del progetto si presenta una nuova opzione: **Structure**.

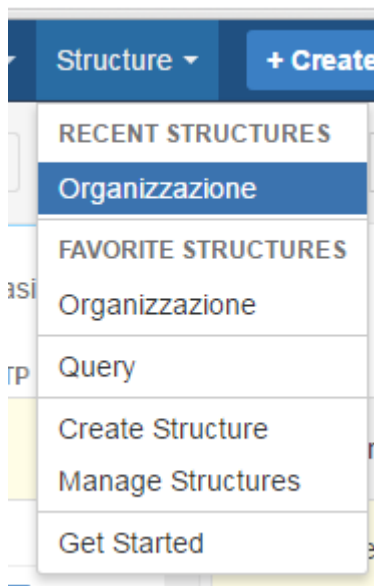


The screenshot displays the JIRA project interface for 'Demo project - Esempio'. The top navigation bar includes 'Dashboards', 'Projects', 'Issues', 'JExcel', 'Structure', and a 'Create' button. The left sidebar contains a menu with 'Summary', 'Issues', 'Reports', 'Versions', and 'Structure'. The 'Structure' option is highlighted, and a tooltip labeled 'Structure' is visible. Below the menu is a 'PROJECT SHORTCUTS' section with links for 'Task management with JIRA' and 'Business projects basics', and an 'Add link' button. The main content area shows the 'Activity' feed for the project, with a date separator for 'April 04'. The activity feed lists three items: 'JIRA Administrator created DPE-3 - demo1 test' (dated 04/Apr/16 3:03 PM), 'FABIO GENOVESE created DPE-2 - test' (dated 04/Apr/16 2:16 PM), and 'FABIO GENOVESE created DPE-1 - demo request' (dated 04/Apr/16 12:25 PM). Each item includes a profile picture, the user name, the action, the issue key, and options to 'Comment', 'Vote', or 'Watch'.

Se la andiamo a selezionare questo è quello che andiamo a scoprire ☐



ovvero entriamo nella gestione delle issues. Attraverso questa schermata è possibile selezionare le issue, utilizzando JQL per la ricerca, per comporre la nostra gerarchia. Adesso proviamo ad usarla. Come prima cosa andiamo a selezionare l'opzione **Create Structure** del nuovo menù **Structure**.



a questo punto possiamo iniziare a gestire la nostra struttura gerarchica:

The screenshot shows the JIRA Structure view. The left panel displays a hierarchy of issues under the 'Organizzazione' view. The right panel shows a JQL query result with 2 items selected.

Key	Summary	Progress	TP
ES-2	Esempio 2 - e		
ES-4	Esempio 4 - e		
DPE-1	demo req		
DPE-4	demo		
ES-6	esempio 6 - e		
ES-2	Esempio 2		
ES-5	esempio 5 - esemp		

Key	Summary	TP
ES-4	Esempio 4 - esempio4 - de	
ES-1	Esempio 1 - esempio1 - de	
ES-6	esempio 6 - esempio6 - de	
ES-5	esempio 5 - esempio5 - de	
ES-3	Esempio 3 - esempio3 - de	
ES-2	Esempio 2 - esempio2 - de	
DPE-3	demo1 - test	
DPE-1	demo request	

dalla immagine precedente vediamo che possiamo, con un semplice drag & drop, andare a comporre la nostra struttura gerarchica, selezionando le varie issue e costruendo la nostra struttura.

Al termine, sempre dalla sezione **Structure** della pagina principale di gestione progetti, possiamo andare a gestire il tutto.

The screenshot shows the JIRA Structure view for a project named 'Demo project - Esempio'. The left sidebar shows the project navigation menu. The main area displays the 'Structure' view with a hierarchy of issues under the 'Organizzazione' view.

Key	Summary	Progress	TP
ES-2	Esempio 2 - esempio2 - descrizione di test		
ES-4	Esempio 4 - esempio4 - descrizione di test		
DPE-1	demo request		
DPE-4	demo		
ES-6	esempio 6 - esempio6 - descrizione di test		

Conclusioni

Abbiamo visto da vicino questo addon. Possiamo creare in maniera molto semplice una struttura gerarchica che, come struttura di supporto, ci permette di superare la limitazione di task/subtask andando a comporre quanto ci serve. Possiamo anche utilizzare issue di progetti differenti, componendo anche situazioni che richiedono la sincronizzazione di più progetti.