



Manuale Utente

Versione: 0.0.1 12/04/2026

Redattori

Luca Parise

Verifica

Approvazione

Uso

Esterno

nextsoftpadova@gmail.com

Registro dei cambiamenti

Versione	Data	Autore	Descrizione	Verifica
0.0.1	19/03/2025	Luca Parise	Creazione documento, stesura introduzione e altri capitoli	
0.0.2	10/04/2025	Luca Parise	Stesura dei capitoli riguardanti le funzionalità del plug-in	
0.0.3	11/04/2025	Luca Parise	Modifica alle immagini e continuazione stesura dei vari capitoli	
0.0.4	12/04/2025	Luca Parise	Aggiunto capitolo per la disinstallazione del prodotto	

Indice

1	Introduzione	4
1.1	Scopo del documento	4
1.2	Scopo del prodotto	4
1.3	Glossario	4
2	Installazione	5
2.1	Requisiti	5
2.1.1	Come creare il file .vsix	5
2.1.2	Come installare il plug-in	6
3	Configurazione	8
3.1	Modello LLM	8
3.1.1	Prerequisiti	8
3.1.2	Come configurare il modello LLM ?	8
3.2	Soglia di accettazione	9
3.2.1	Come configurare la soglia dei requisiti ?	9
4	Istruzioni all'uso	10
4.1	Prerequisiti	10
4.2	Importazione dei requisiti	10
4.3	Analisi dell'implementazione	12
4.4	Analisi dei requisiti	13
4.5	Esportazione requisiti	15
4.6	Ricerca requisiti	16
4.7	Analisi di un singolo requisito	16
4.8	Analisi dell'implementazione di un singolo requisito	17
4.9	Filtraggio dei requisiti	17

5	Disinstallazione del plug-in	19
6	Elenco delle immagini	21

1 Introduzione

“**Requirement Tracker per Visual Studio Code**” è un plug-in progettato per l’omonimo editor, con l’obiettivo di supportare l’analisi dei requisiti software. Dato un insieme di requisiti, sia tracciati che non tracciati, il plug-in esegue un’analisi approfondita e fornisce una valutazione del loro grado di implementazione all’interno del codice sorgente.

1.1 Scopo del documento

Il seguente manuale fornisce una guida dettagliata all’installazione, configurazione e utilizzo del plug-in. L’obiettivo è consentire agli utenti di comprendere il funzionamento dello strumento e di sfruttarne appieno le funzionalità per il monitoraggio e la valutazione dei requisiti software all’interno del codice sorgente.

In particolare, il documento si propone di:

- Fornire le istruzioni passo-passo per l’installazione e la configurazione
- Descrivere le funzionalità principali del plug-in
- Spiegare come eseguire l’analisi dei requisiti e interpretare i risultati

1.2 Scopo del prodotto

Il plug-in è progettato per supportare gli sviluppatori e i team di progetto nel tracciamento e nella verifica dei requisiti software. Il plug-in analizza il codice sorgente per identificare il livello di implementazione dei requisiti, fornendo una valutazione chiara e strutturata.

Il prodotto sfrutta l’intelligenza artificiale basata su modelli di linguaggio di grandi dimensioni (LLM) per migliorare l’analisi del codice e il riconoscimento dei requisiti. Grazie a questa tecnologia, il plug-in è in grado di comprendere il contesto del codice sorgente, rilevare correlazioni con i requisiti specificati e suggerire miglioramenti o integrazioni.

L’obiettivo principale è migliorare la gestione dei requisiti all’interno del ciclo di sviluppo, garantendo maggiore coerenza tra specifiche e codice, riducendo il rischio di requisiti mancanti o non conformi.

1.3 Glossario

Per evitare ambiguità relative alle terminologie utilizzate è stato creato un documento denominato **Glossario**. Questo documento comprende tutti i termini tecnici scelti dai membri del gruppo e utilizzati nei vari documenti con le relative definizioni. Tutti i termini inclusi in questo glossario vengono segnalati all’interno del documento con l’apice ^G accanto alla parola.

2 Installazione

2.1 Requisiti

Sono necessarie le seguenti tecnologie installate:

- Node.js versione 20.0.0 o superiore
- npm
- TypeScript (installabile con `npm install -g typescript`)
- Visual Studio Code
- Ollama attivo sulla porta standard `localhost:11434`

2.1.1 Come creare il file .vsix

Per creare un file **.vsix** (che è il pacchetto installabile di una estensione per Visual Studio Code), devi usare lo strumento **vsce** (Visual Studio Code Extension Manager).

Di seguito i vari passaggi per creare tale pacchetto:

- **1)** Apri il terminale e installa **vsce** (Visual Studio Code Extension Manager), se ancora non lo hai installato, con il comando:

```
$ npm install -g vsce
```

- **2)** Naviga con il terminale fino alla cartella della tua estensione. Quindi vai nella cartella in cui si trova il file **package.json** della tua estensione. Puoi usare il seguente comando per navigare nella cartella:

```
$ cd /path/to/your/extension
```

- **3)** Costruisci il file **.vsix** con il seguente comando, il quale genera un file **.vsix**, ad esempio `nome-estensione-0.0.1.vsix`:

```
$ vsce package
```

- **4) (Facoltativo)** Ignora i file non necessari. Puoi usare un file **.vscodeignore** nella root della tua estensione per escludere i file o cartelle (come `node_modules`, `test/`, ecc.) dal pacchetto finale.

2.1.2 Come installare il plug-in

Una volta creato il pacchetto .vsix come delineato nel capitolo precedente è possibile installare il plug-in in Visual Studio Code. Per farlo, segui questi passaggi:

- **1)** Apri Visual Studio Code
- **2)** Apri un terminale e naviga nella cartella in cui hai salvato il file .vsix. Puoi usare il comando:

```
$ cd /path/to/your/extension
```

- **3)** Usa il seguente comando per installare il pacchetto .vsix:

```
$ code --install-extension nome-estensione-0.0.1.vsix
```

- **4)** Se non vedi la estensione installata prova a riavviare Visual Studio Code
- **5)** Verifica che l'estensione sia installata correttamente. Puoi farlo andando nella sezione delle estensione di Visual Studio Code e cercando il nome del plug-in oppure via terminale usando il seguente comando:

```
$ code --list-extensions
```

Ora l'estensione dovrebbe essere stata installata con successo. Per poterla usare avrete anche bisogno di Ollama attivo e configurato con un modello LLM. Per farlo leggete il prossimo capitolo.

2.1.2.1 Attivare Ollama

Per usare l'estensione è necessario che Ollama sia attivo e stia ascoltando sulla porta standard 11434. Per fare ciò, aprite un terminale e digitate il comando

```
$ ollama run nomemodello
```

dove nome modello indica il modello usato. Di base l'estensione usa il modello llama3.2:3b ma potete cambiarlo in qualsiasi momento andando su settings del plug-in. Per fare questo vi rimando al capitolo apposito per la configurazione del modello da usare.

2.1.2.2 Attivare il server API

Per fare questo dovete posizionarvi nella cartella API in cui troverete il codice in typescript che gestisce il server. Siccome viene usato node sarà necessario convertire il codice in javascript. Per fare ciò aprite un terminale, posizionatevi sulla cartella ed eseguite il comando

```
$ npm install
```

```
$ npx tsc
```

Questi comandi installeranno le dipendenze e convertiranno il codice typescript in codice javascript in una cartella chiamata dist. Quindi per avviare il server spostatevi dal terminale sulla cartella dist e cercate il file main.js. Quindi eseguite il comando

```
$ node main.js
```

Questo comando avvierà il server che rimarrà in ascolto delle richieste e le invierà ad Ollama. Se tutto è andato liscio, potete proseguire.

3 Configurazione

3.1 Modello LLM

Il modello LLM di Requirement Tracker per Visual Studio Code viene eseguito tramite Ollama, permettendo un'analisi locale dei requisiti software. Esamina la qualità testuale dei requisiti e verifica la loro implementazione nel codice sorgente, restituendo un punteggio da 0 a 100 per indicare il grado di conformità. Inoltre, aiuta a identificare le sezioni di codice correlate ai requisiti e consente la configurazione di modelli personalizzati per ottimizzare le prestazioni dell'analisi.

3.1.1 Prerequisiti

I modelli di Ollama utilizzati sono configurabili tramite un file dedicato: config.json. Modello di default: llama3.2:3b

- Assicurati che Ollama sia attivo sulla porta 11434 prima di utilizzare l'estensione e che i relativi modelli siano disponibili.
- La porta diversa da quella di default si può specificare nel file: .env insieme alla porta dell'API.
- Nel file config.json del plugin si può modificare l'url del server node.

3.1.2 Come configurare il modello LLM ?

Clicca qui > Clicca la > E Uala

3.2 Soglia di accettazione

Quando un requisito viene sottoposto ad analisi, il modello LLM ne valuta l'implementazione nel codice sorgente e assegna un punteggio numerico compreso tra 0 e 100. Questo valore indica il grado di conformità del codice rispetto al requisito, fornendo una misura quantitativa della correttezza e completezza dell'implementazione.

3.2.1 Come configurare la soglia dei requisiti ?

4 Istruzioni all'uso

4.1 Prerequisiti

Se siete arrivati qui, allora avete installato il plug-in e siete pronti ad usarlo.

Quindi, cliccate sull'icona del plug-in presente nella colonna a sinistra della finestra di visual studio code per aprire la finestra di analisi dei requisiti. L'icona è mostrata nella figura sottostante.

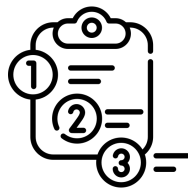


Figure 1: Icona del plug-in

4.2 Importazione dei requisiti

Per poter usare il plug-in è necessario importare i requisiti da analizzare. Per fare ciò, cliccate sull'icona denominata **"Load Requirements"** mostrata in figura.

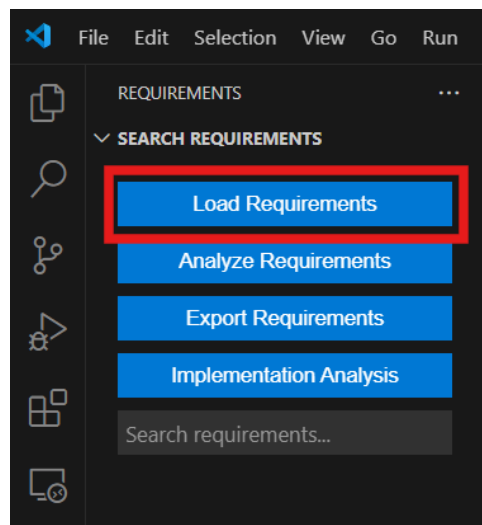


Figure 2: Icona per l'importazione dei requisiti

Una volta cliccato sull'icona verrà aperto il file system di sistema da cui selezionare il **file CSV** contenente i requisiti. In caso venga selezionato un file di formato diverso o non valido, il plug-in restituirà un messaggio di errore.

E' consigliato, anche se non necessario, aver aperto in Visual Studio Code il progetto contenente il codice sorgente.

Una volta importati, i requisiti verranno presentati in una lista nella sezione Requirements, come mostrato nella seguente figura:

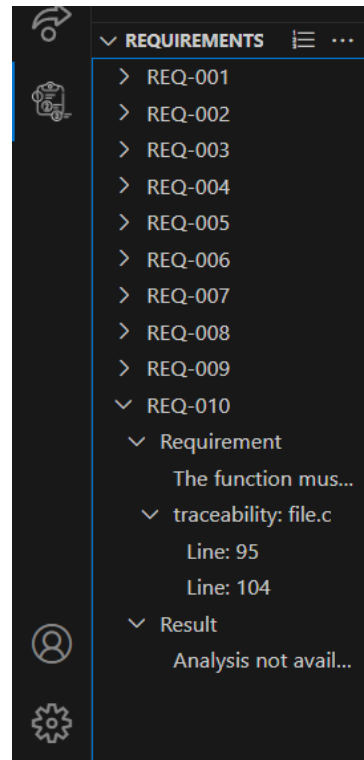


Figure 3: Requisiti importati

La lista, una volta selezionato un file CSV valido, elencherà per ogni requisito all'interno del documento:

- ID
- Descrizione
- Tracciabilità nel codice sorgente (solo se presente nel file CSV precedentemente caricato)

In caso di requisiti molto lunghi o di più righe sarà sufficiente posizionare il cursore sopra il requisito per visualizzare il testo completo.

4.3 Analisi dell'implementazione

Una volta importati i requisiti è possibile procedere con l'analisi dell'implementazione. Questa funzionalità serve a tracciare i requisiti nel codice sorgente qualora questi non fossero già tracciati (funziona anche nel caso lo fossero).

Per farlo, cliccate sull'icona **"Implementation Analysis"** mostrata in figura.

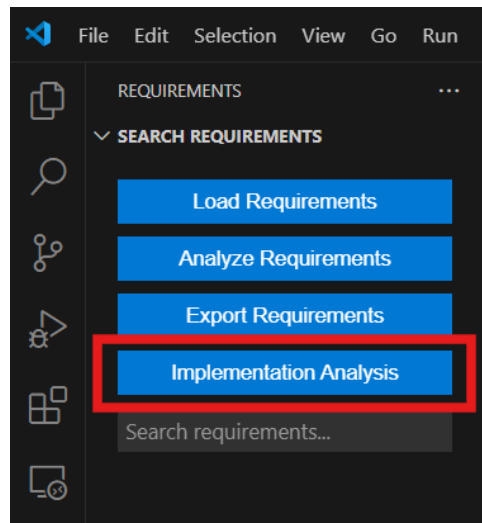


Figure 4: Bottone per l'analisi dell'implementazione

Una volta eseguita e completata l'analisi, ad ogni requisito verranno associate delle linee di codice che indicano il tracciamento del requisito all'interno del codice sorgente.

In caso di requisiti non tracciati, il plug-in restituirà un messaggio di errore. La figura seguente mostra il tracciamento nella sezione "Requirements":

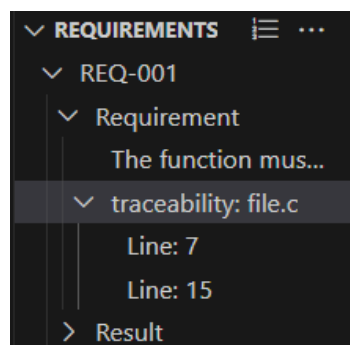


Figure 5: Tracciabilità di un requisito

4.4 Analisi dei requisiti

Questa funzionalità serve a valutare la qualità dei requisiti e il loro grado di implementazione nel codice sorgente. Una volta importati i requisiti, cliccate sull'icona "Analyze Requirements" mostrata in figura.

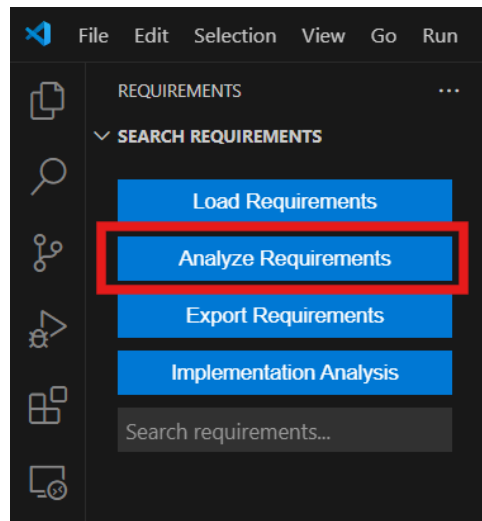


Figure 6: Bottone per l'analisi dei requisiti

In questo modo il plug-in eseguirà un'analisi approfondita di tutti i requisiti presenti nella lista. Potrete vedere in basso a destra della finestra un'icona di avanzamento che indica il progresso dell'analisi.

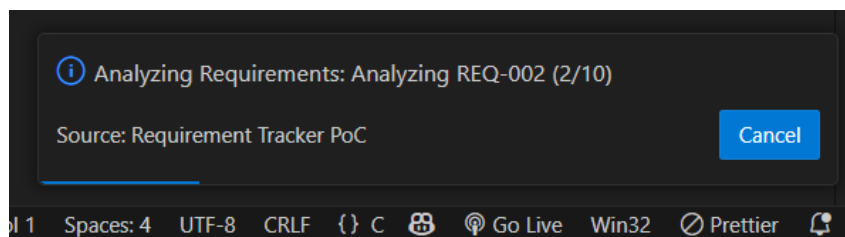


Figure 7: Stato di avanzamento analisi dei requisiti

Alla fine dell'analisi ogni requisito nella lista verrà aggiornato con i seguenti campi:

- Result
 - Result: passed/not passed
 - Code Compliance: 0-100
 - Issues:
 - Issue 1
 - Issue ...

- Suggestions:
 - Suggestion 1
 - Suggestion ...

Inoltre, ad ogni requisito sarà associato un'icona che indica se lo stesso è passato o meno all'analisi. Di seguito una figura che mostra una generica lista a seguito dell'analisi.

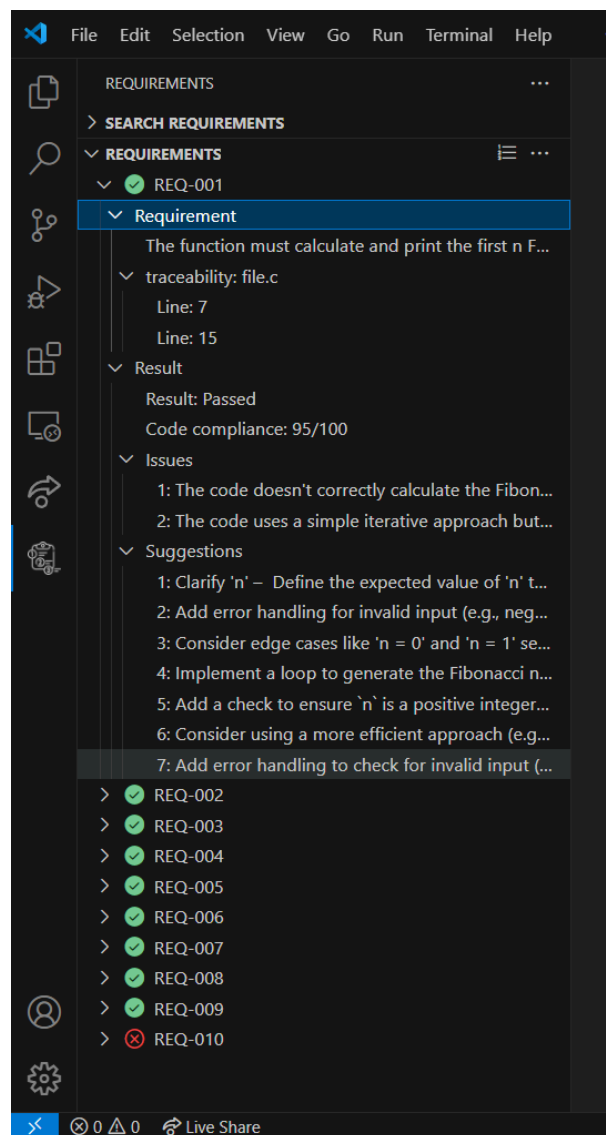


Figure 8: Risultato dell'analisi

4.5 Esportazione requisiti

Il plug-in presenta anche la funzione di esportazione, in formato CSV, dei risultati dell'analisi, qualora volesse salvarli. Per fare ciò premete sul pulsante **“Export Requirements”** mostrato nella seguente figura.

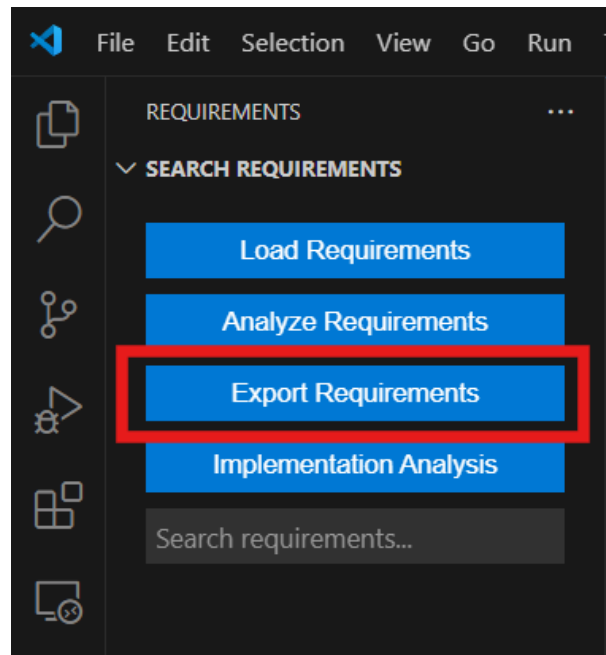


Figure 9: Bottone per l'esportazione dei requisiti

Quindi, una volta cliccato sul pulsante, verrà aperto il file system di sistema da cui selezionare la cartella in cui salvare il file CSV.

4.6 Ricerca requisiti

Il plug-in presenta anche la funzione di ricerca dei requisiti attraverso una barra di ricerca in cui inserire il codice del requisito ricercato. Per farlo, basta cliccare sull'icona “**Search Requirements**” mostrata in figura.

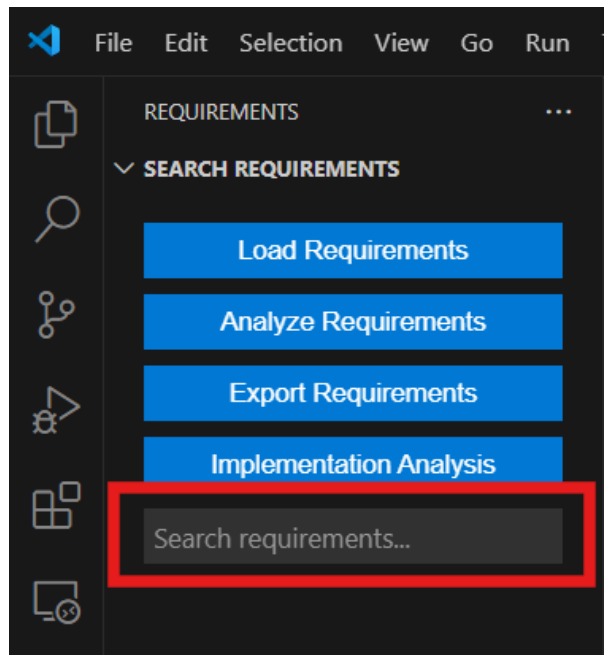


Figure 10: Campo di input per la ricerca dei requisiti

La lista **Requirements** mostrerà solo i requisiti che contengono il codice (o parte di esso) inserito nella barra di ricerca.

4.7 Analisi di un singolo requisito

Il plug-in offre la possibilità di analizzare un singolo requisito. Per farlo, basta cliccare sull'icona presente vicino al requisito che si desidera analizzare come mostrato in figura.

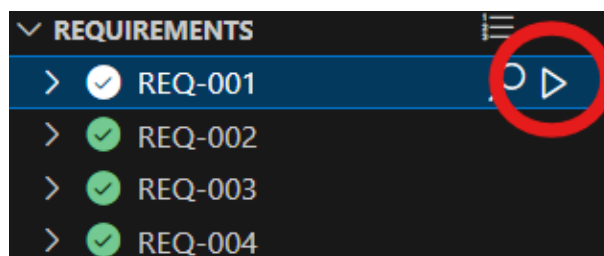


Figure 11: Bottone per l'analisi di un singolo requisito

4.8 Analisi dell'implementazione di un singolo requisito

Il plug-in offre la possibilità di analizzare l'implementazione di un singolo requisito. Per farlo, basta cliccare sull'icona presente vicino al requisito che si desidera analizzare come mostrato in figura.

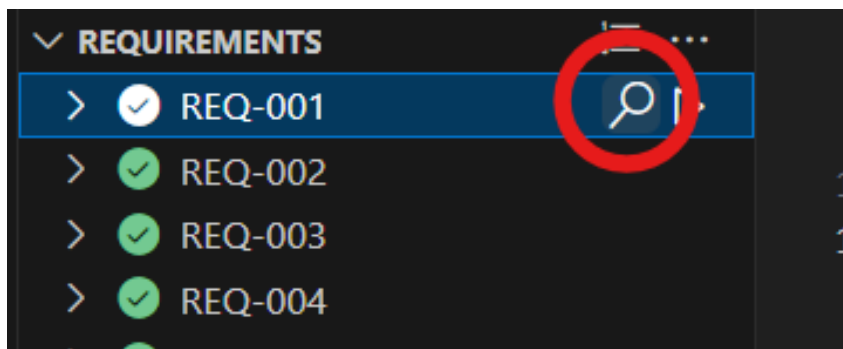


Figure 12: Bottone per l'analisi dell'implementazione di un singolo requisito

4.9 Filtraggio dei requisiti

Oltre a poter cercare i requisiti a seconda del loro codice, è possibile filtrare i requisiti in base al loro codice o al loro stato di analisi.

Ci sono quattro diversi modi in cui possono essere filtrati i requisiti:

- In ordine **crescente** (a seconda del loro codice), premendo il pulsante **“Default Order”** mostrato in figura

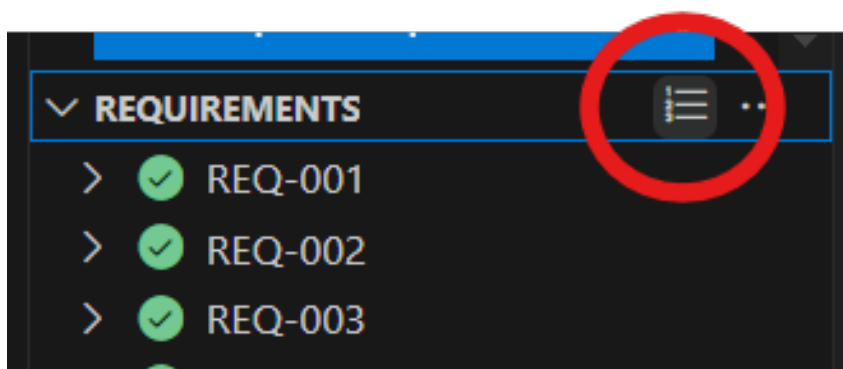


Figure 13: Bottone per organizzare i requisiti in ordine di default

- In ordine “**decrescente**”, premendo il pulsante “**Sort by ID (descending)**”. Mostrato in figura

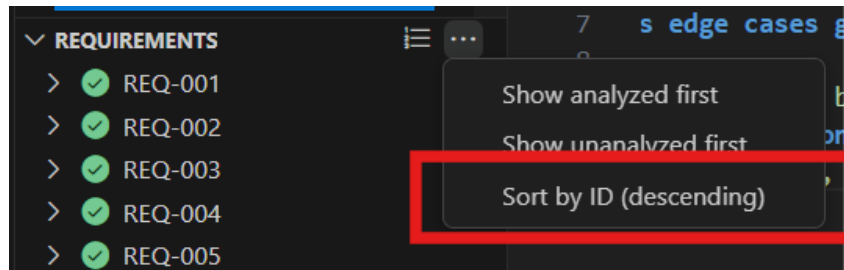


Figure 14: Bottone per organizzare i requisiti in ordine decrescente

- A seconda del loro **stato di analisi**:
 - **Analizzati**, usando il bottone “**Show analyzed first**” mostrato in figura

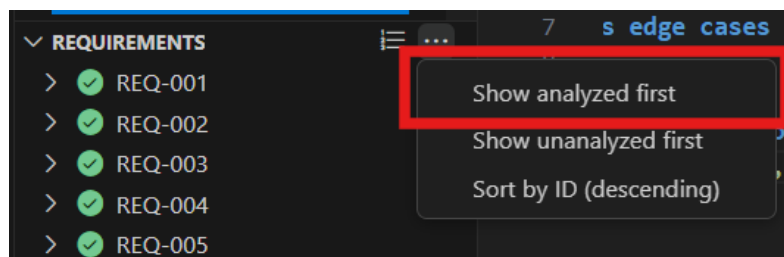


Figure 15: Bottone per organizzare i requisiti in ordine decrescente

- **Non Analizzati**, usando il bottone “**Show unanalyzed first**” mostrato in figura

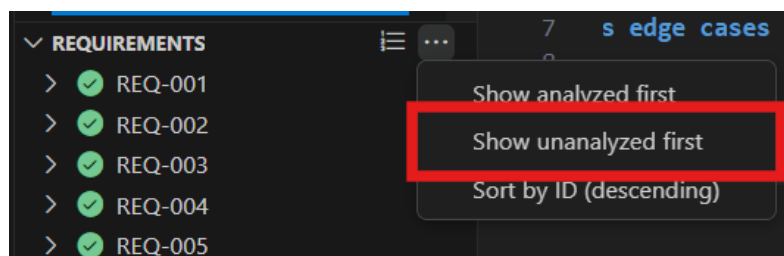


Figure 16: Bottone per organizzare i requisiti in ordine decrescente

5 Disinstallazione del plug-in

La disinstallazione del plug-in è molto semplice. Per farlo basta usare l'interfaccia di Visual Studio Code. Andate in basso a sinistra e cliccate sull'icona delle impostazioni a forma di ingranaggio e selezionate **"Extensions"**. Fatto ciò, vi comparirà la lista delle estensioni installate e raccomandate come in figura :

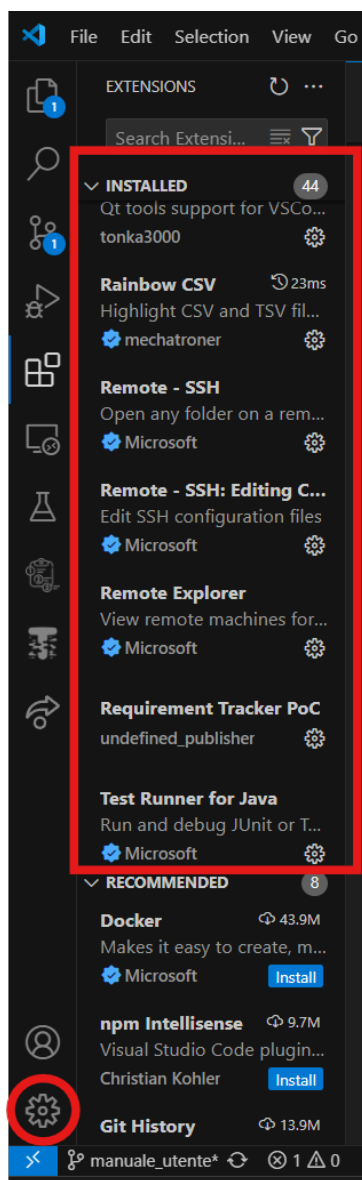


Figure 17: Icona delle estensioni

Quindi, cercate l'estensione **Requirement Tracker** e cliccate sull'icona a forma di ingranaggio vicino ad essa. Vi comparirà un menù a tendina in cui dovrete selezionare **"Uninstall"** come mostrato in figura:

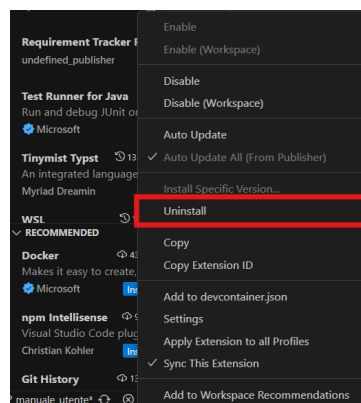


Figure 18: Icona del plug-in

Una volta cliccato Uninstall l'estensione verrà disinstallata e non sarà più visibile nella lista delle estensioni installate.

6 Elenco delle immagini