



# Error\_\_418

[GitHub/Error-418-SWE](#)

[error418swe@gmail.com](mailto:error418swe@gmail.com)

---

## Verbale interno 07/12/23

Discussione conseguente ad incontro con il professor Cardin

---

### Informazioni

---

<b>Versione</b>	1.0.0
<b>Uso</b>	Interno
<b>Stato</b>	Approvato
<b>Responsabile</b>	Carraro Riccardo
<b>Redattore</b>	Nardo Silvio
<b>Verificatore</b>	Todesco Mattia
<b>Destinatari</b>	Gruppo Error__418 Vardanega Tullio Cardin Riccardo

## 1 Informazioni generali

- Luogo: Discord
- Data e ora: 07/12/23 @ 16:00 ~ 16:40
- Partecipanti (7):
  - Banzato Alessio
  - Carraro Riccardo
  - Gardin Giovanni
  - Nardo Silvio
  - Oseliero Antonio
  - Todesco Mattia
  - Zaccone Rosario

## 2 Ordine del giorno

A seguito di un dialogo che il gruppo ha avuto con il professor Cardin in data 07/12/23, sono stati discussi i seguenti argomenti:

- requisiti non funzionali;
- raggruppamento degli Use Case;
- quali sono i criteri che definiscono un'Analisi dei Requisiti ben fatta.

### 2.1 Requisiti non funzionali

#### 2.1.1 Specificare le versioni

È stata sottolineata l'importanza di specificare, nei requisiti non funzionali, i numeri di versione specifici dei componenti tecnologici con cui l'applicativo dovrà interfacciarsi, in particolare:

- le versioni dei sistemi operativi che supportano l'applicativo;
- le versioni dei software di gestione del database con il quale verrà interfacciato l'applicativo;
- le versioni dei browser che supportano l'applicativo.

#### 2.1.2 Requisiti di qualità

Altri requisiti non funzionali da includere sono relativi alla qualità e a come l'applicativo debba rispettare determinate norme prestabilite.

#### 2.1.3 Requisiti prestazionali

È necessario analizzare e definire con particolare attenzione i requisiti non funzionali relativi alle prestazioni, in quanto serve descrivere minuziosamente il contesto che permetta di replicare il requisito (velocità della rete, cardinalità delle operazioni, ecc) per poter poi imporre dei vincoli prestazionali su tale contesto.

#### 2.1.4 Requisito relativo all'utilizzo di alfabeti non latini

Diversamente da quanto ipotizzato inizialmente, il requisito riguardante la possibilità di utilizzare alfabeti non latini è un requisito funzionale in quanto rappresenta una vera e propria feature che determina il comportamento del programma.

## 2.2 Raggruppamento degli Use Case

La suddivisione degli Use Case è da svolgere in funzione alla loro leggibilità e utilità, perciò si ritiene più ragionevole suddividere gli Use Case relativi alla modifica dell'ambiente di lavoro in:

- modifica scaffale;
- modifica ambiente;
- modifica bin.

E successivamente, per ciascuno di questi, si può, in base alle esigenze, estenderli con i relativi Use Case legati a:

- creazione;
- ridimensionamento;
- spostamento;
- eliminazione.

## 2.3 Criteri che definiscono un'Analisi dei Requisiti ben fatta

Le caratteristiche di un'Analisi dei Requisiti ottimale sono:

- sintassi dei diagrammi UML corretta rispetto allo standard;
- requisiti atomici;
- requisiti deterministici;
- requisiti non ambigui;
- requisiti non troppo generici;
- chiaro tracciamento del rapporto tra Use Case e requisiti e viceversa.

## 3 To do

In luce di quanto concordato durante il meeting, le prossime attività da svolgere sono:

- dividere gli Use Case troppo grandi, in particolare quello relativo alla modifica dell'ambiente di lavoro;
- modificare i requisiti non funzionali sviluppandoli ulteriormente;
- aggiornare i diagrammi UML in funzione delle nuove modifiche agli Use Case.