**SPRINT 3**

Per l**’analisi dei requisiti** riferirsi al [documento](https://liveunibo-my.sharepoint.com/:w:/r/personal/eugenio_cerulo_studio_unibo_it/_layouts/15/Doc.aspx?sourcedoc=%7BAAB47CD8-EB32-4E88-A443-1AFA19B8DB71%7D&file=Analisi%20dei%20Requisiti.docx&action=default&mobileredirect=true).

**Problem analysis**

All’interno di questo Sprint si cerca di tener conto di tutti i requisiti, senza dover fare ulteriori semplificazioni.

In particolare, si vuole approfondire il **task di pulizia del tavolo rendendola un’azione interrompibile**.

*Problematica della pulizia dei tavoli interrompibile*

Nello [sprint 2](https://liveunibo-my.sharepoint.com/:w:/r/personal/eugenio_cerulo_studio_unibo_it/_layouts/15/Doc.aspx?sourcedoc=%7BEEB9F1E1-C128-49C3-B726-403316A36A8D%7D&file=Analisi%20del%20problema.Sprint2.docx&action=default&mobileredirect=true) si è supposto che il tavolo potesse essere in uno dei seguenti stati:

* **Pulito** (clean)
* **Occupato** (taken)
* **Sporco** (dirty)

Nella realtà sappiamo che l’operazione di pulizia nel periodo del covid-19 risulta essere più complessa e articolata, in particolare l’operazione di pulizia può essere suddivisa in due momenti:

* **Sparecchiare** (clear)
* **Igienizzare** (sanitize)

Di conseguenza viene naturale aggiungere nuovi stati alla condizione del tavolo, gli stati definitivi in cui un tavolo si può trovare sono quindi i seguenti:

* **Clean**
* **Taken**
* **Dirty**
* **Clear**

Per passare da Dirty a clear il tavolo deve essere sparecchiato e analogamente, per passare da clear a clean è necessario igienizzare il tavolo.

Tali modifiche possono essere implementate molto facilmente, grazie al fatto che si è deciso di utilizzare una base di conoscenza prolog all’interno dello sprint 2.

Infatti, è sufficiente estendere la base di conoscenza aggiungendo i nuovi stavi e le nuove regole di transizione.  
Per quanto riguarda gli attori, è necessario aggiungere gli stati al waiterbody per poter andare a sparecchiare e igienizzare il tavolo, e al tearoom per poter effettuare le transizioni sopra descritte.

*Modello*

Disponibile presso il link: [*https://github.com/virtualms/IssProject/tree/master/sprint3*](https://github.com/virtualms/IssProject/tree/master/sprint3)

*TEST PLAN*

In questo sprint, un possibile test plan consiste nel verificare che lo stato di un tavolo cambi, in ordine, i seguenti stati: clean -> dirty -> clear -> clean.

Il codice del test plan è disponibile presso il link:

<https://github.com/virtualms/IssProject/blob/master/sprint3/tests/WaiterTest_Sprint3.kt>

***PROGETTO***

Per implementare le modifiche richieste nell’analisi del problema di questo sprint è sufficiente arricchire il modello QAK dello sprint 2 con i nuovi stati per poter andare a sparecchiare e igienizzare il tavolo.

Di conseguenza vanno aggiunte anche le nuove richieste, la tearoom si occupa di fornire le informazioni sui tavoli da pulire (cioè da igienizzare/sparecchiare).

Il funzionamento del web server rimane invariato dal secondo sprint, l’unica modifica sarà nello stato mostrato nella pagina, che mostrerà anche quando il waiter sta sparecchiando e igienizzando uno dei tavoli.