04/07/2020

**PIAZZA\_SPRINT 3**

## Sprint Goal

Essere in grado di gestire più clienti, seppur in maniera "non ottimizzata".

## Requisiti

-I requisiti sono gli stessi elencati nel file [TFBO20ISS.pdf](../../../../DOCUMENTS/TFBO20ISS.pdf) con **alcune assunzioni semplificative**.

-Assunzioni:

1) Si rilassa il vincolo de "un solo cliente in sala". Ora possono arrivare tutte le richieste di ingresso che si vuole. Come da requisiti, la stanza potrà accogliere fino a N=2 clienti contemporaneamente.

2) Al barman, di conseguenza, potrà essere richiesto di lavorare alla preparazione di più ordini, anche in parallelo. Potrà lavorare al massimo su N=2 preparazioni in parallelo.

3) Il tempo di preparazione di un ordine è sempre lo stesso, a prescindere da cosa è stato ordinato.

4) I task del waiter **non** sono interrompibili.

## Analisi dei Requisiti

COSA deve essere il sistema? Mi aspetto che al termine di questo Sprint il sistema si comporti in questo modo:

### Bozze di Test Plan

## Analisi del Problema

### Problematiche riprese dall’overview iniziale e da affrontare in questo sprint

Stima del maxWaitingTime: cosa fare se arriva un cliente e tutti i due tavoli sono dirty (free but not clean)? In questo caso si può pensare ad una stima diversa da quella pensata nell’overview e che sia esattamente uguale al tempo di pulizia di un tavolo. Se uno è busy e uno è dirty? In questo caso maxWaitingTime = min(clean\_time, maxStayTime+ LongestPreparationTime)

* Che fare se un cliente dice “Sono pronto a ordinare”, oppure “sono pronto a pagare” ma il waiter sta servendo un altro cliente? Potrebbe scadere il maxStayTime mentre il waiter sta finendo di servire l’altro cliente. Il cliente non ha colpa…tuttavia non sarebbe giusto secondo me dare a lui la responsabilità di informare direttamente il maxstaytimeobserver.

Al massimo devo introdurre un nuovo attore che faccia le veci del waiter per tenere sotto controllo solo la situazione maxstaytime.

* Simulare più clienti concorrenti.
* Il barman deve gestire più ordini contemporaneamente.

Nell’ottica di avere un feedback più chiaro ed efficiente con il committente, il progettista valuti l’opportunità di fare un’interfaccia grafica con alcuni pulsanti che possano permettere di “suonare il campanello” tutte le volte che voglio, e ogni volta viene mandata una ring.

Volendo si possono mettere anche dei pulsanti che permettano al cliente di dire: sono pronto ad ordinare, sono pronto a pagare. In questo modo non devo simulare i tempi e impazzire a sincronizzare. Decido io con un pulsante quando il cliente prende iniziativa.

Poi si può fare un’interfaccia dove faccio vedere lo stato della stanza.

### Architettura Logica

### Modello eseguibile

## Test Plan

## Progetto

**SPRINT 2 – REVIEW**