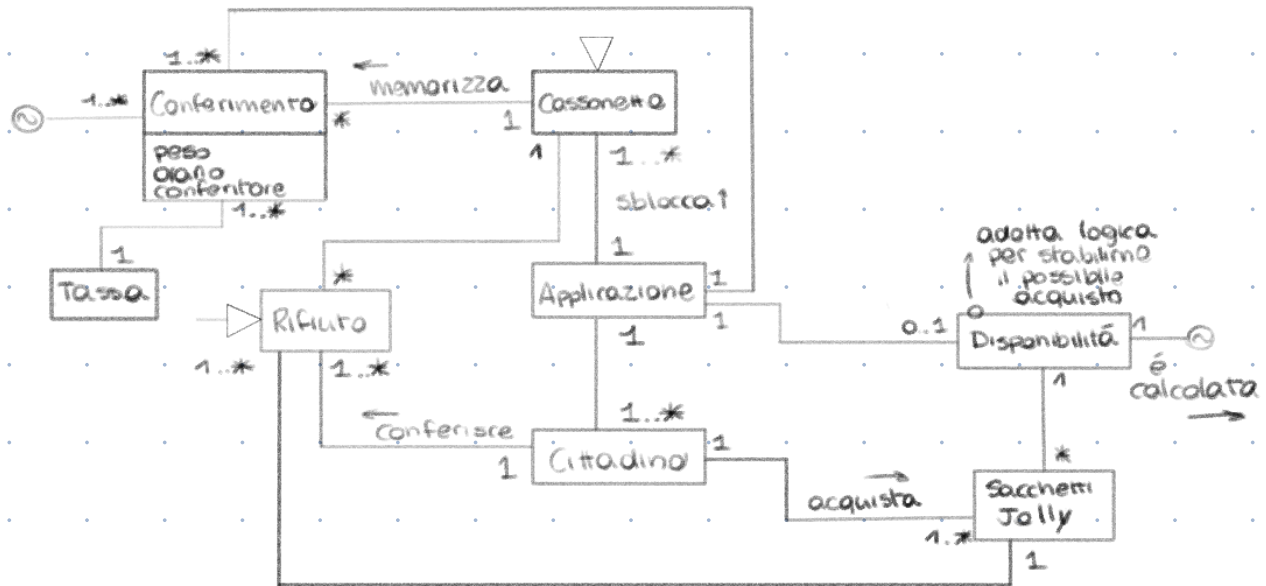
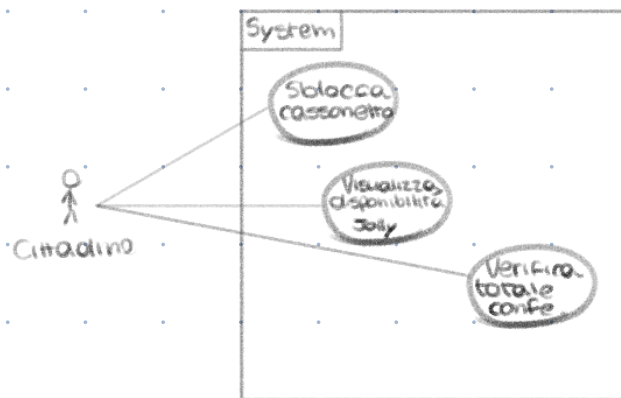


Domain model



Use case



UC: Sblocca cassonetto

Attore: Cittadino

Pre: Cittadino autenticato dal sistema

1. Cittadino seleziona apertura cassonetto

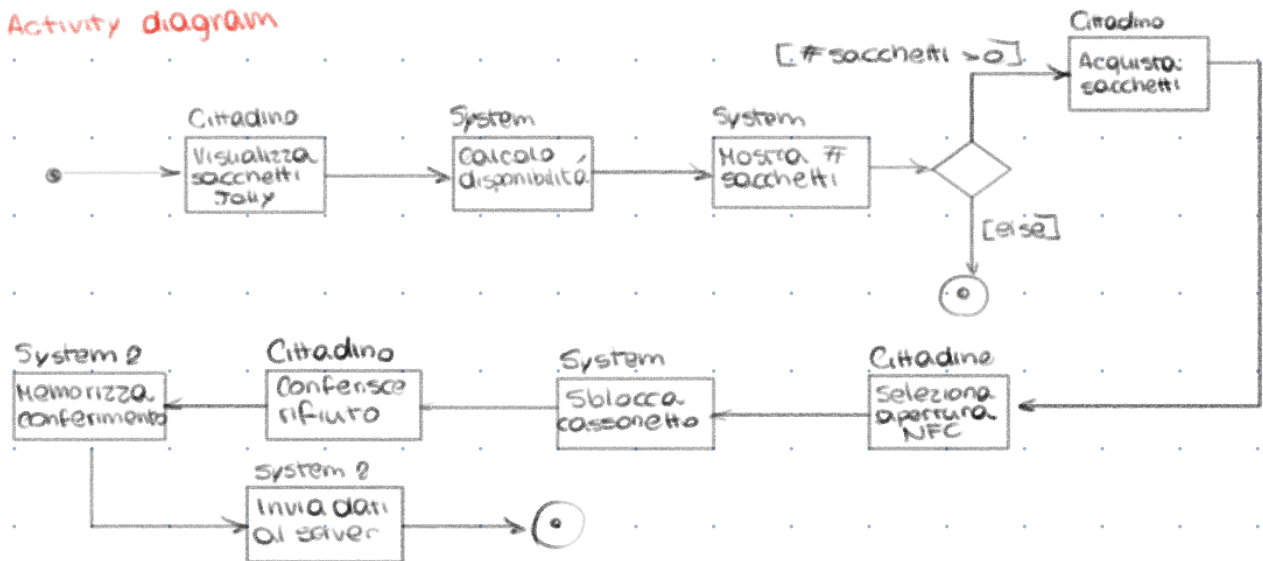
2. Sistema attiva NFC

3. Cliente pone maschera NFC in prossimità del lettore

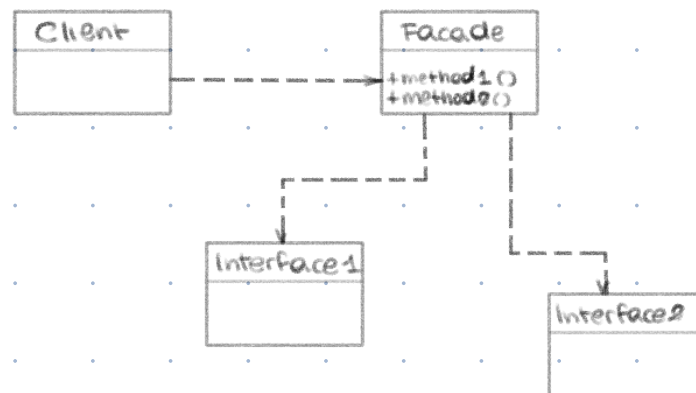
4. Sistema conferma avvenuta apertura

Post: Cittadino conferisce sacchetto nel cassonetto

Activity diagram



- ② Facade è un pattern strutturale del catalogo GoF, attuato affinché l'architettura interna del sistema sia manipolabile e dinamica, dinanzi alla necessità di introdurre nuove modifiche.
- L'intento del pattern consiste nel favorire totale isolamento alle classi di basso livello rispetto ad entità del dominio.
- Spesso soluzioni software si compongono di differenti sottosistemi, dove ognuno dei quali adotta propri layer di astrazione; perciò per classi di alto livello potrebbe divenire difficile mantenere ogni riferimento.
- Pertanto interviene Facade in grado di sviluppare un unico punto di visualizzazione che contenga ogni metodo realmente attuabile; rappresenta un aggregatore di interfacce, stabilendo in questo modo, un univoco punto di collegamento tra i due domini.
- Tuttavia, per far sì che si possa adeguare, l'entità introdotta deve consistere in un'interfaccia di alto livello, priva da logica algoritmica, definita empty class.
- La volontà di un approccio simile è causato dalla violazione di single responsibility principle, poiché l'interfaccia mantiene metodi di ogni natura, non sottostando alla definizione di asse di cambiamento; pertanto, l'assenza di logica algoritmica, permette che essa non provochi alcuna erronea gestione di dipendenze.
- È scontato definire che, mediante lo sviluppo di un'ulteriore entità astratta, sia pienamente soddisfatto dependency inversion principle.



③ Backlog rappresenta un arrefatto della metodologia agile Scrum; si ricorda che lo scopo delle metodologie agile consiste nell'impiego del maggior tempo possibile su attività di sviluppo e implementazione della soluzione software, rispetto a pratiche burocratiche, come la stesura di una documentazione esaustiva.

Rispetto alla struttura ideata, si evidenziano due backlog, definiti product backlog e sprint backlog, entrambi fondamentali per il lifecycle che caratterizza Scrum.

Product backlog contiene una lista di user stories, pertanto di obiettivi valorizzabili, a cui è associato un grado di importanza, stabilito rispetto alla natura tecnica, in termini di complessità, e in base alle priorità definite dal product owner.

Proseguendo, raffigura un insieme di intenti che dovranno essere terminati e sviluppati entro un certo numero limite di sprint; ossia di iterazioni implementative.