**Università degli Studi di Salerno**

Simone Cristiano 0512105070

Mario Offertucci 0512105184

Valeria Santarpia 0512105232

Smart Restaurant  
Object Design Document

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versione** | **Descrizione** | **Autore** |
| 03/01/2020 | 1.0 | Prima stesura del documento | Valeria Santarpia |
| 4/01/2020 | 2.0 | Inseriti screen package | Valeria Santarpia |

# 1. Introduzione

**1*.*1 Object Design Trade-offs**

Sicurezza vs Efficienza   
La sicurezza rappresenta un elemento importante nel sistema. La piattaforma client di tipo Android ci faciliterà tale compito, essendo di per sé

Response Time vs Hardware  
Il tempo di risposta rappresenta un fattore importante nel sistema, in particolare su determinate funzionalità, come lettura delle comande e del menu. Tutto ciò dipenderà, tuttavia, anche dal tipo di rete utilizzata per lo scambio di informazioni tra client e server.

**1*.*2 Interface documentation guidelines**

Verranno seguite le seguenti linee guida per la scrittura del codice:

Classi Java:

* I nomi dovranno iniziare con la lettera maiuscola;
* Se il nome contiene più parole, ognuna di esse dovrà iniziare con una lettera maiuscola;
* I nomi dovranno corrispondere alle informazioni e/o funzionalità che offre quella classe.

Metodi:

* I nomi dovranno iniziare con la lettera minuscola;
* Se il nome contiene più parole, ognuna di esse dovrà iniziare con la lettera maiuscola (tranne la prima);
* I nomi dovranno corrispondere alle informazioni e/o funzionalità che offre quel metodo. Si utilizzeranno verbi o aggettivi;
* I nomi dei metodi per ottenere e settare attributi saranno chiamati ‘get[nomeattributo]’ e ‘set[nomeattributo]’.

Variabili:

* I nomi delle variabili dovranno iniziare con la lettera minuscola;
* Se il nome contiene più parole, ognuna di esse dovrà iniziare con la lettera maiuscola (tranne la prima).

Costanti:

* I nomi delle costanti dovranno essere in maiuscolo;
* Se il nome contiene più parole, esse saranno separate da un underscore.

**1*.*3 Referenze**

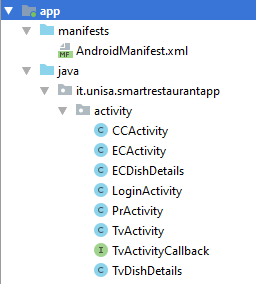
* **Requirement Analysis Document**
* **System Design Document**

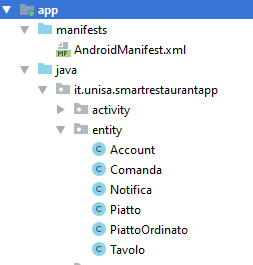
# 2. Package

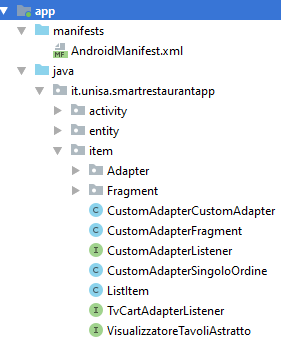
**2*.*1 Lato client**

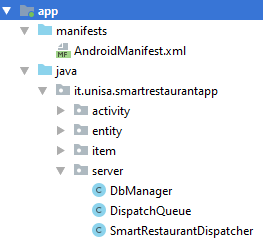
Il sistema è diviso nei seguenti package:

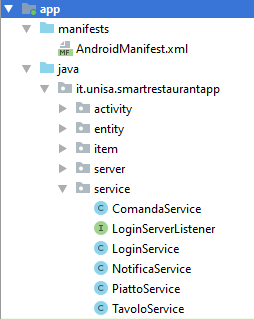
* activity;
* entity;
* item;
* server;
* service;
* androidTest;
* drawable;
* layout;
* mipmap;
* values.

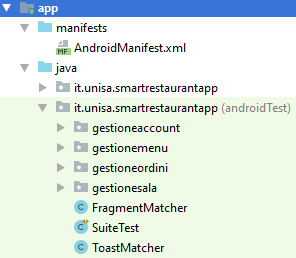


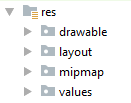








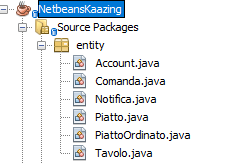


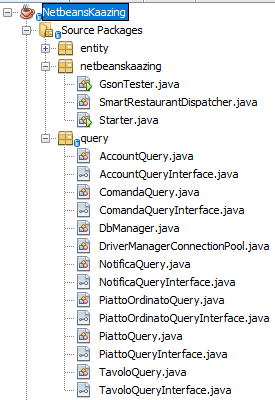


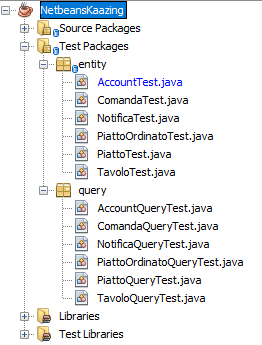
**2*.*2 Lato server**

Il sistema è diviso nei seguenti package:

* entity;
* netbeanskaazing;
* query.







.

# 3. Javadoc

Per le informazioni riguardanti le classi ed i loro contenuti consultare il javadoc di client e server.