



Università degli Studi di Salerno
Corso di Ingegneria del Software

Object Design Document
Quick Service Project

Versione 1.0



Data 19/05/2008

Prefazione:

Questo documento contiene i requisiti del Quick Service. E' destinato ai designers e ai clienti del progetto.

Target:

Clienti, Sviluppatori

MEMBRI DEL QUICK SERVICE:**Team Members:**

Vicidomini Vincenzo

Mercurio Antonio

Iacoletti Alessandro

Pacifico Marta

Pizza Luca Ernesto

INDICE DEI CONTENUTI

1. INTRODUZIONE.....	
1.1 OBJECT DESIGN TRADE-OFF.....	
1.2 LINEE GUIDA PER LA DOCUMENTAZIONE DELLE INTERFACCE.....	
1.3 DEFINIZIONI, ACRONIMI ED ABBREVIAZIONI.....	
2. PACKAGE	
3. INTERFACCIA DELLE CLASSI	
4. GLOSSARIO	

1. Introduzione

Il documento descrive gli obiettivi del design imposti dagli sviluppatori e le linee guida per lo sviluppo delle interfacce dei sottosistemi e la suddivisione dei sottosistemi in package e classi.

1.1. Object Design trade – off

Questo documento si propone di evidenziare gli obiettivi principali dello sviluppo dal punto di vista degli sviluppatori:

- L'applicazione non deve richiedere molte risorse al sistema in termini di memoria e tempo CPU, in questo modo è possibile usare anche macchine non recenti
- Il tempo di risposta non è fondamentale, ma non deve esserci una risposta troppo lenta del sistema, in particolare la parte che interagisce con l'utente.
- Il sistema deve sfruttare al meglio la rete senza sovraccaricarla inutilmente
 - Lo spazio occupato dai file dell'applicazione sul disco non è fondamentale visto il rapporto costo/spazio dei supporti

1.2. Linee guida per la documentazione delle Interfacce

Nel redigere il codice del programma verranno usate le seguenti convenzioni oltre alle convenzioni che sono normalmente adottate nella scrittura del codice Java:

- I nomi delle Classi devono essere al singolare ed avere l'iniziale maiuscola, in caso sia il nome di un frame preceduti dal prefisso “frm_”.
- I nomi dei pannelli sono preceduti dal prefisso “pnl”
- I nomi dei pulsanti sono preceduti dal prefisso “btn”
- I nomi delle label sono preceduti dal prefisso “lbl”
- I nomi delle tabelle sono preceduti dal prefisso “tbl”
- Le classi che implementano le interfacce hanno il suffisso “Imp”
- I metodi hanno per nome un verbo che indica l'azione che compie il metodo

1.3. Definizioni, acronimi e abbreviazioni

Di seguito sono riportati alcune abbreviazioni usate dagli sviluppatori durante la scrittura del codice:

- obj = object
- uid = User ID

- pwd = password
- hh = ora
- mm = minuti
- ss = secondi

2. Package

In questa sezione sono riportati tutti i package che costituiscono l'applicazione.

2.1. Package QsServer

In questo Package ci sono le classi e le interfacce relative all'interfaccia e alla classe della sessione semplice e altre classi necessarie al funzionamento del server, le classi delle altre sessioni si trovano nei sottopacchetti QsServer.DataLayer, QsServer.AddettoBar, QsServer.Cliente.

Interface Summary

<u>Sessione</u>	Titolo: Sessione Descrizione: Interfaccia per la sessione di default
------------------------	----------------------------------------------------------------------

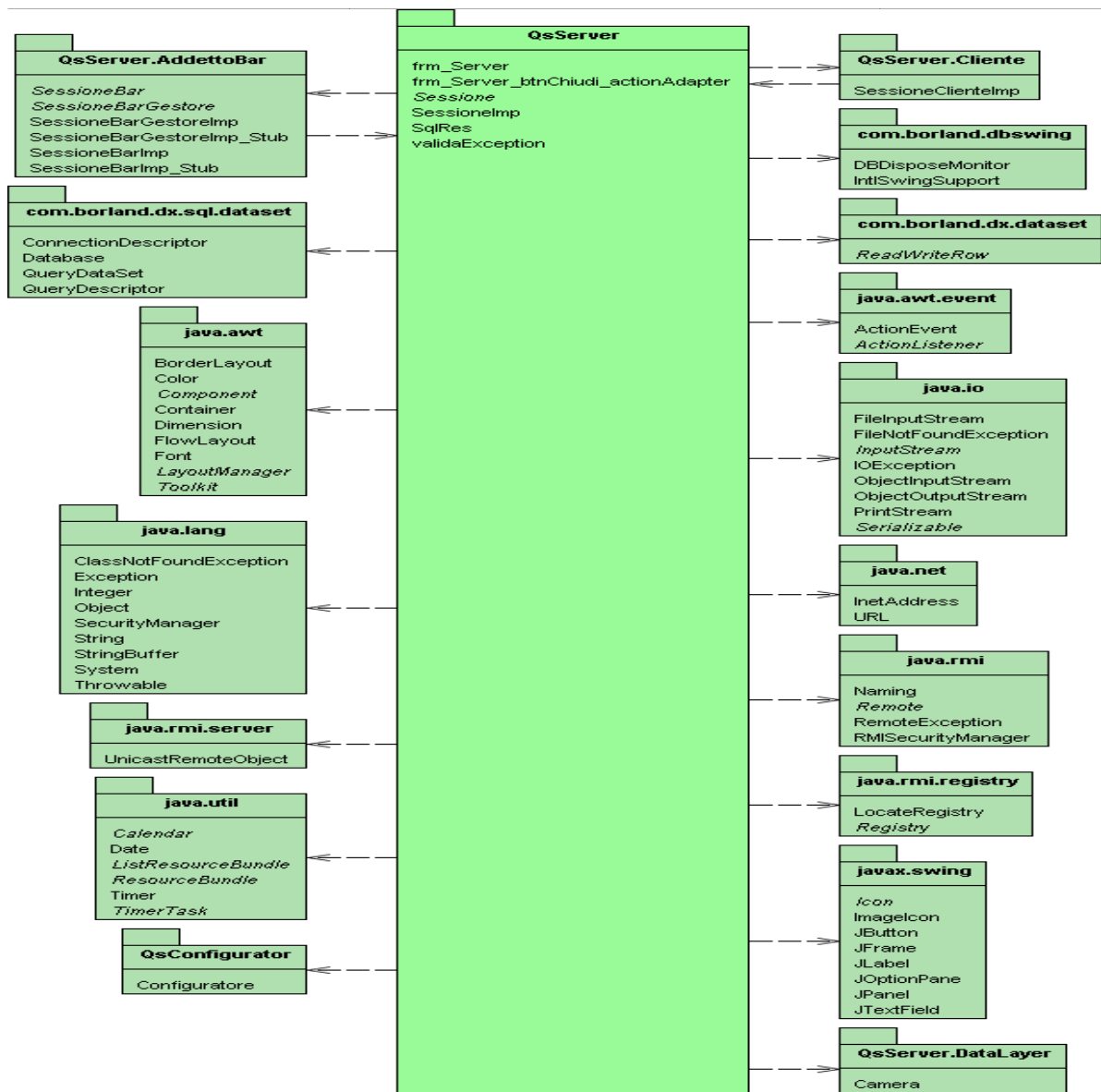
Class Summary

<u>frm_Log</u>	Titolo: frm_Log Descrizione: La classe permette di gestire i log registrati dal sistema
<u>frm_Log_btnChiudi_actionAdapter</u>	Titolo: frm_Log_btnChiudi_actionAdapter Descrizione: L'Adapter associa il metodo actionPerformed al pulsante btnChiudi
<u>frm_Log_btnClear_actionAdapter</u>	Titolo: frm_Log_btnClear_actionAdapter Descrizione: Permette di associare l'actionPerformed al pulsante btnClear

<u>frm_Log_btnRefresh_actionAdapter</u>	Titolo: frm_Log_btnRefresh_actionAdapter Descrizione: L'Adapter associa il metodo actionPerformed al pulsante btnRefresh
<u>frm_Server</u>	Titolo: frm_Server Descrizione: Mostra il frame principale del server
<u>frm_Server_btnChiudi_actionAdapter</u>	Titolo: frm_Server_btnChiudi_actionAdapter Descrizione: Permette di associare l'actionPerformed al pulsante btnChiudi
<u>frm_Server_btnLog_actionAdapter</u>	Titolo: frm_Server_btnLog_actionAdapter Descrizione: Permette di associare l'actionPerformed al pulsante btnLog
<u>SessioneImp</u>	Titolo: SessioneImp Descrizione: Progetto QuickService
<u>SqlRes</u>	Titolo: SqlRes Descrizione: Raggruppa tutte le query in una matrice

Exception Summary

<u>validaException</u>	Titolo: validaException Descrizione: Eccezione per la funziona valida
-------------------------------	--------------------------------------------------------------------------



2.2. Package QsServer.AddettoBar

Contiene le classi e le interfacce relative alle sessioni dell'addettoBar.

Interface Summary

<u>SessioneBar</u>	<p>Titolo: SessioneBar</p> <p>Descrizione: Interfaccia per la sessione Bar, fornisce i metodi basilari per il bar</p>
<u>SessioneBarGestore</u>	<p>Titolo: SessioneBarGestore</p> <p>Descrizione: nterfaccia per la sessione Bar, fornisce i metodi basilari per la gestione del bar</p>

Class Summary

<u>SessioneBarGestoreImp</u>	Titolo: SessioneBarGestoreImp Descrizione: Implementa i metodi dell'interfaccia SessioneBarGestore
<u>SessioneBarImp</u>	Titolo: SessioneBarImp Descrizione: Implementa i metodi dell'interfaccia SessioneBar

Exception Summary

<u>ProdottoBarException</u>	Titolo: ProdottoBarException Descrizione: Eccezione che viene lanciata in caso di errori nella gestione dei prodotti bar
------------------------------------	-----------------------------------------------------------------------------------------------------------------------------



2.3. Package QsServer.Cliente

Contiene le classi e le interfacce relative alla sessione del Cliente

Interface Summary

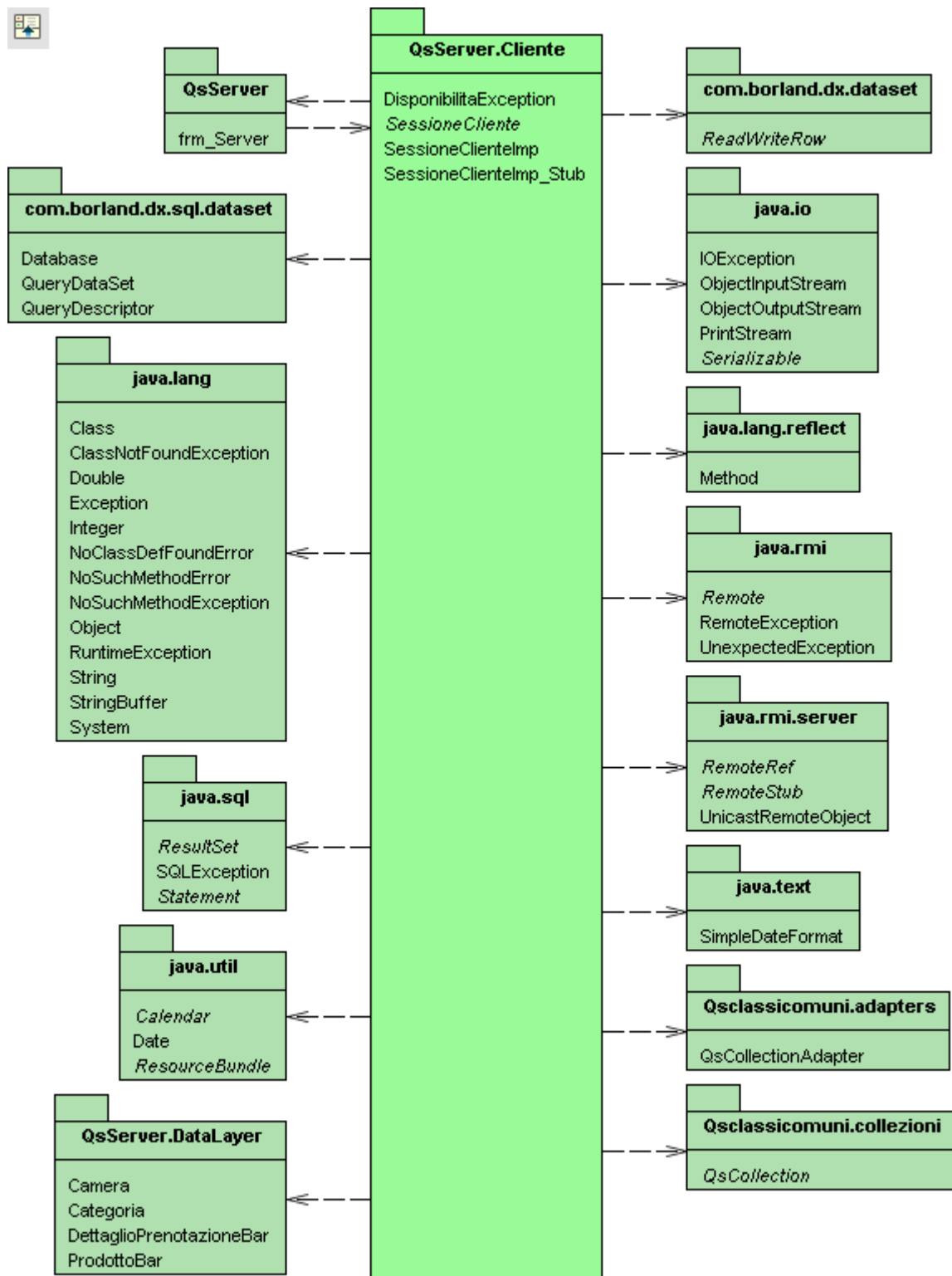
<u>SessioneCliente</u>	Titolo: SessioneCliente Descrizione: Interfaccia che fornisce i metodi necessari al client per effettuare/modificare/cancellare le ordinaizoni
-------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------

Class Summary

<u>SessioneClienteImp</u>	Titolo: SessioneClienteImp Descrizione: Progetto QuickService
----------------------------------	------------------------------------------------------------------

Exception Summary

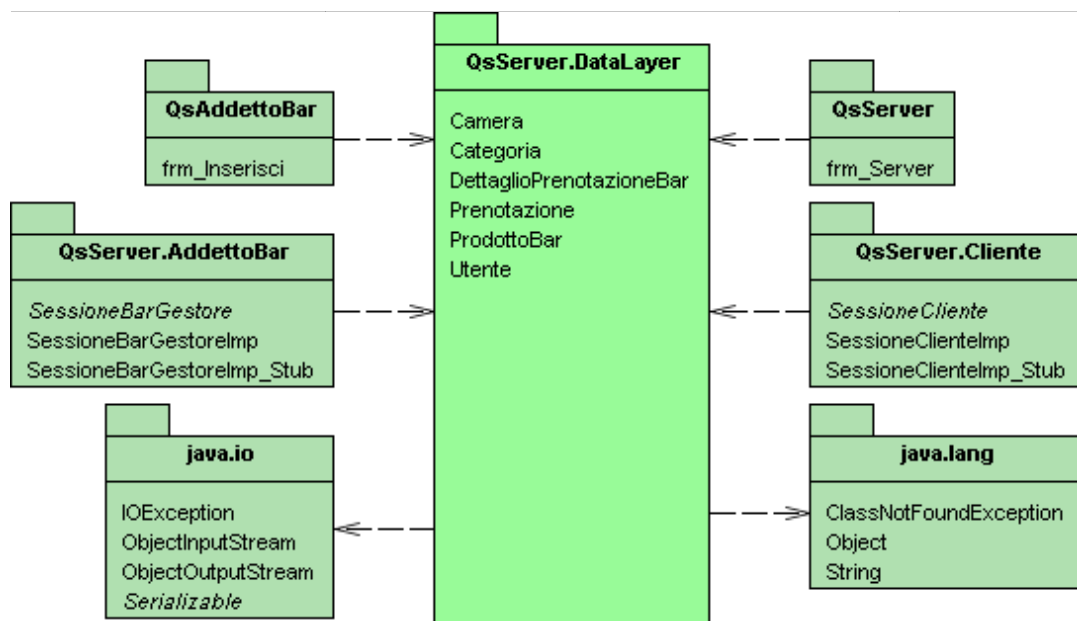
<u>DisponibilitaException</u>	Titolo: DisponibilitaException Descrizione: Eccezione che viene lanciata in caso la disponibilità non è sufficiente per soddisfare la richiesta di un determinato prodotto
--------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



2.4. Package QsServer.DataLayer

Contiene le classi del data layer che vengono usate per rappresentare le entità di cui tener traccia all'interno del sistema.

Class Summary	
<u>Camera</u>	Titolo: Camera Descrizione: Descrive una camera della struttura
<u>Categoria</u>	Titolo: Categoria Descrizione: Descrive una categoria di prodotti
<u>DettaglioPrenotazioneBar</u>	Titolo: DettaglioPrenotazioneBar Descrizione: Descrive elemento di una prenotazione
<u>Prenotazione</u>	Titolo: Prenotazione Descrizione: Descrive una prenotazione
<u>ProdottoBar</u>	Titolo: ProdottoBar Descrizione: Descrive un prodotto del bar
<u>Utente</u>	Titolo: Utente Descrizione: Descrive l'utente che si logga al sistema



2.5. Package QsServer.util

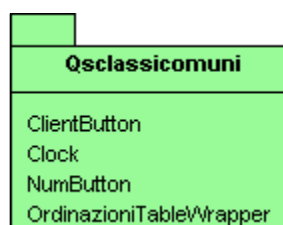
Contiene le classi per la gestione dei log di sistema

Class Summary	
<u>Log</u>	Titolo: Log Descrizione: Descrive un evento del sistema
<u>LogModel</u>	Titolo: LogModel Descrizione: Modello di tabella, questo modello incorpora una array list come struttura dati interna, la classe client continua ad usare la tabella a prescindere dalla struttura dati utilizzata per la collezione

2.6. Package Qsclassicomuni

Contiene Classi che vengono usate sia dai client che dal server, le classi adapter e le collezioni sono nei sottopacchetti Qsclassicomuni.adapters, Qsclassicomuni.collezione.

Class Summary	
<u>ClientButton</u>	Titolo: ClientButton Descrizione: Descrive i pulsanti per i prodotti del listino cliente
<u>Clock</u>	Titolo: Clock Descrizione: Descrive un orologio
<u>NumButton</u>	Titolo: NumButton Descrizione: Descrive i pulsanti del tastierino numerico
<u>OrdinazioniTableWrapper</u>	Titolo: OrdinazioniTableWrapper Descrizione: Modello di tabella, questo modello incorpora una array list come struttura dati interna, la classe client continua ad usare la tabella a prescindere dalla struttura dati utilizzata per la collezione



2.7. Package Qsclassicomuni.adapters

Contiene gli adapter per le classi collection

Class Summary

<u>QsCollectionAdapter</u>	Titolo: QsCollectionAdapter Descrizione: Questa Classe invece ha il ruolo di adapter cioè fa da tramite tra la classe target e la classe Adaptee, ovvero la classe che fornisce l'implementazione al target, che nel nostro caso è la classe ArrayList.
-----------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2.8. Package Qsclassicomuni.collezioni

Contiene le classi collection per recuperare più oggetti dello stesso tipo dal database.

Interface Summary

<u>QsCollection</u>	Titolo: QsCollection Descrizione: Questa classe rappresenta la classe Target a cui il client accede.
----------------------------	---------------------------------------------------------------------------------------------------------

Class Summary

<u>CollectionListino</u>	Titolo: CollectionListino Descrizione: Collezione di oggetti di tipo prodottoBar che compongono il listino
---------------------------------	---------------------------------------------------------------------------------------------------------------

2.9. Package QsAddettoBar

Contiene le classi per l'interfaccia grafica del client dell'addetto bar.

Class Summary

<u>frm AddettoBar</u>	Titolo: frm_AddettoBar Descrizione: La classe rappresenta il frame principale dove vengono istanziati i pulsanti che permettono l'interazione con l'utente.
<u>frm Inserisci</u>	Titolo: frm_Inserisci Descrizione: Rappresenta il frame che permette l'inserimento di un nuovo prodotto nell'archivio

<u>frm Inserisci btnAnnulla actionAdapter</u>	<p>Titolo: frm_Inserisci_btnAnnulla_actionAdapter</p> <p>Descrizione: La classe funge da Adapter per poter gestire l'actionPerformed del pulsante btnAnnulla</p>
<u>frm Inserisci btnChiudi actionAdapter</u>	<p>Titolo: frm_Inserisci_btnChiudi_actionAdapter</p> <p>Descrizione: La classe funge da Adapter per poter gestire l'actionPerformed del pulsante btnChiudi</p>
<u>frm Inserisci btnOk actionAdapter</u>	<p>Titolo: frm_Inserisci_btnOk_actionAdapter</p> <p>Descrizione: La classe funge da Adapter per poter gestire l'actionPerformed del pulsante btnOk</p>
<u>frm Inserisci cbxillimitato itemAdapter</u>	<p>Titolo: frm_Inserisci_cbxillimitato_itemAdapter</p> <p>Descrizione: Associa gli eventi legati al cambiamento di stato del check box cbxillimitato</p>
<u>frm Inserisci descrizione keyAdapter</u>	<p>Titolo: frm_Inserisci_descrizione_keyAdapter</p> <p>Descrizione: Associa gli eventi legati all'input da tastiera per l'inserimento del campo descrizione</p>
<u>frm Inserisci prezzo key Adapter</u>	<p>Titolo: frm_Inserisci_prezzo_keyAdapter</p> <p>Descrizione: Associa gli eventi legati all'input da tastiera per l'inserimento del campo prezzo</p>
<u>frm Login</u>	<p>Titolo: frm_Login</p> <p>Descrizione: Il frame permette all'utente di inserire nome e password e loggarsi all'interno del sistema)</p>
<u>frm Login btnAnnulla actionAdapter</u>	<p>Titolo: frm_Login_btnAnnulla_actionAdapter</p> <p>Descrizione: Progetto La classe funge da Adapter per poter gestire l'actionPerformed del pulsante btnAnnulla</p>
<u>frm Login btnOk actionAdapter</u>	<p>Titolo: frm_Login_btnOk_actionAdapter</p> <p>Descrizione: La classe funge da Adapter per poter gestire l'actionPerformed del pulsante btnOk</p>
<u>frm Visualizza</u>	<p>Titolo: frm_Visualizza</p> <p>Descrizione: Visualizza in una tabella il listino</p>
<u>frm Visualizza btnChiudi actionAdapter</u>	<p>Titolo: frm_Visualizza_btnChiudi_actionAdapter</p> <p>Descrizione: permette di associare il metodo btnChiudi_actionPerformed(e) al pulsante btnChiudi</p>
<u>frm Visualizza cbRicerca itemAdapter</u>	<p>Titolo: frm_Visualizza_cbRicerca_itemAdapter</p> <p>Descrizione: Permette di associare l'evento cbRicerca_itemStateChanged(e) all'oggetto</p>

<u>frm_Visualizza_QVisualizzaListino_calcFieldsAdapter</u>	Titolo: frm_Visualizza_QVisualizzaListino_calcFieldsAdapter Descrizione: classe Adapter per i campi calcolati
<u>frm_VisualizzaOrdinazioni</u>	Titolo: frm_VisualizzaOrdinazioni Descrizione: visualizza le ordinazioni effettuate dai clienti che sono state
<u>frmAddettoBar_btnCancella_actionAdapter</u>	Titolo: frmAddettoBar_btnCancella_actionAdapter Descrizione: La classe funge da Adapter per poter gestire l'actionPerformed del pulsante btncancella
<u>frmAddettoBar_btnExit_actionAdapter</u>	Titolo: frmAddettoBar_btnExit_actionAdapter Descrizione: La classe funge da Adapter per poter gestire l'actionPerformed del pulsante btnExit
<u>frmAddettoBar_btnInserisci_actionAdapter</u>	Titolo: frmAddettoBar_btnInserisci_actionAdapter Descrizione: La classe funge da Adapter per poter gestire l'actionPerformed del pulsante btnInserisci
<u>frmAddettoBar_btnLogout_actionAdapter</u>	Titolo: frmAddettoBar_btnLogout_actionAdapter Descrizione: a classe funge da Adapter per poter gestire l'actionPerformed del pulsante btnLogout
<u>frmAddettoBar_btnModifica_actionAdapter</u>	Titolo: frmAddettoBar_btnModifica_actionAdapter Descrizione: La classe funge da Adapter per poter gestire l'actionPerformed del pulsante btnModifica
<u>frmAddettoBar_btnVisualizza_actionAdapter</u>	Titolo: frmAddettoBar_btnVisualizza_actionAdapter Descrizione: La classe funge da Adapter per poter gestire l'actionPerformed del pulsante btnVisualizza
<u>frmAddettoBar_btnVisualizzaPrenotazioni_actionAdapter</u>	Titolo: frmAddettoBar_btnVisualizzaPrenotazioni_actionAdapter Descrizione: La classe funge da Adapter per poter gestire l'actionPerformed del pulsante btnVisualizzaPrenotazioni
<u>VisualizzaPrenotazioni_btnChiudi_actionAdapter</u>	Titolo: VisualizzaPrenotazioni_btnChiudi_actionAdapter Descrizione: Funge da adapter per associare il metodo btnChiudi_actionPerformed(e) al pulsante btnChiudi



2.10. Package QsCliente

Contiene le classi usate dai client per l'interfaccia grafica del client per i clienti

Class Summary	
<u>frm_Bar</u>	<p>Titolo: frm_bar</p> <p>Descrizione: La classe permette di navigare il listino ed effettuare delle prenotazioni</p>

<u>frm Bar btnCancella Prodotto actionAdapter</u>	<p>Titolo: frm_Bar_btnCancellaProdotto_actionAdapter</p> <p>Descrizione: L'Adapter permette di gestire l'actionPerformed del pulsante btnCancellaProdotto</p>
<u>frm Bar btnChiudi actionAdapter</u>	<p>Titolo: frm_Bar_btnChiudi_actionAdapter</p> <p>Descrizione: L'Adapter permette di gestire l'actionPerformed del pulsante btnChiudi</p>
<u>frm Bar btnConferma actionAdapter</u>	<p>Titolo: frm_Bar_btnConferma_actionAdapter</p> <p>Descrizione: L'Adapter permette di gestire l'actionPerformed del pulsante btnConferma</p>
<u>frm Bar btnNuovaOrdinazione actionAdapter</u>	<p>Titolo: frm_Bar_btnNuovaOrdinazione_actionAdapter</p> <p>Descrizione: L'Adapter permette di gestire l'actionPerformed del pulsante btnNuovaOrdinazione</p>
<u>frm Bar btnOpzioni actionAdapter</u>	<p>Titolo: frm_Bar_btnOpzioni_actionAdapter</p> <p>Descrizione: L'adapter permette di gestire l'actionPerformed del pulsante btnOpzioni</p>
<u>frm Cliente</u>	<p>Titolo: frm_Cliente</p> <p>Descrizione: Mostra il frame principale del cliente</p>
<u>frm Cliente btnBar actionAdapter</u>	<p>Titolo: frm_Cliente_btnBar_actionAdapter</p> <p>Descrizione: Permette di associare l'actionPerformed al pulsante</p>
<u>frm Note</u>	<p>Titolo: frm_Note</p> <p>Descrizione: Rappresenta il frame per l'inserimento delle opzioni sul prodotto selezionato</p>
<u>frm Note btnAnnulla actionAdapter</u>	<p>Titolo: frm_Note_btnAnnulla_actionAdapter</p> <p>Descrizione: l'adapter permette di associare l'actionPerformed al pulsante btnAnnulla</p>
<u>frm Note btnInserisci actionAdapter</u>	<p>Titolo: frm_Note_btnInserisci_actionAdapter</p> <p>Descrizione: l'adapter permette di associare l'actionPerformed al pulsante btnInserisci</p>



3. Class Interface

Verranno presentate solo le classi principali del progetto per le altre riferirsi alla documentazione javadoc.

QsServer

Class SessioneImp

```
java.lang.Object
  java.rmi.server.RemoteObject
    java.rmi.server.RemoteServer
      java.rmi.server.UnicastRemoteObject
        QsServer.SessioneImp
```

All Implemented Interfaces:

java.rmi.Remote, java.io.Serializable, Sessione

Direct Known Subclasses:

SessioneBarImp

```
public class SessioneImp
  extends java.rmi.server.UnicastRemoteObject
  implements Sessione
```

Title: SessioneImp

Description: Implementazione della sessione di default

Version:

1.0

Author:

Mercurio Antonio, Vicidomini Vincenzo

Field Summary

Fields inherited from class java.rmi.server.RemoteObject

Ref

Constructor Summary

SessioneImp ()

Costruttore di default

Method Summary

boolean	<u>controllaOra</u> (int[] d, int ahh, int amm, int ass, int mills) controlla l'ora del server con l'ora relativa alla prenotazione
java.util.Date	<u>getData</u> () Restituisce la data del server

Methods inherited from class java.rmi.server.UnicastRemoteObject

clone, exportObject, exportObject, exportObject, unexportObject

Methods inherited from class java.rmi.server.RemoteServer

getClientHost, getLog, setLog

Methods inherited from class java.rmi.server.RemoteObject

equals, getRef, hashCode, toString, toStub

Methods inherited from class java.lang.Object

finalize, getClass, notify, notifyAll, wait, wait, wait

Constructor Detail

SessioneImp

```
public SessioneImp()  
    throws java.rmi.RemoteException
```

Costruttore di default

Throws:

java.rmi.RemoteException - lancia l'eccezione per problemi con RMI

Method Detail

getData

```
public java.util.Date getData()
```

Restituisce la data del server

Specified by:

getData in interface Sessione

Returns:

Date data del server

controllaOra

```
public boolean controllaOra(int[] d,  
                             int ahh,  
                             int amm,  
                             int ass,  
                             int mills)
```

controlla l'ora del server con l'ora relativa alla prenotazione

Specified by:

controllaOra in interface Sessione

Parameters:

d - int[] array che contiene la data della prenotazione

ahh - int ora della prenotazione

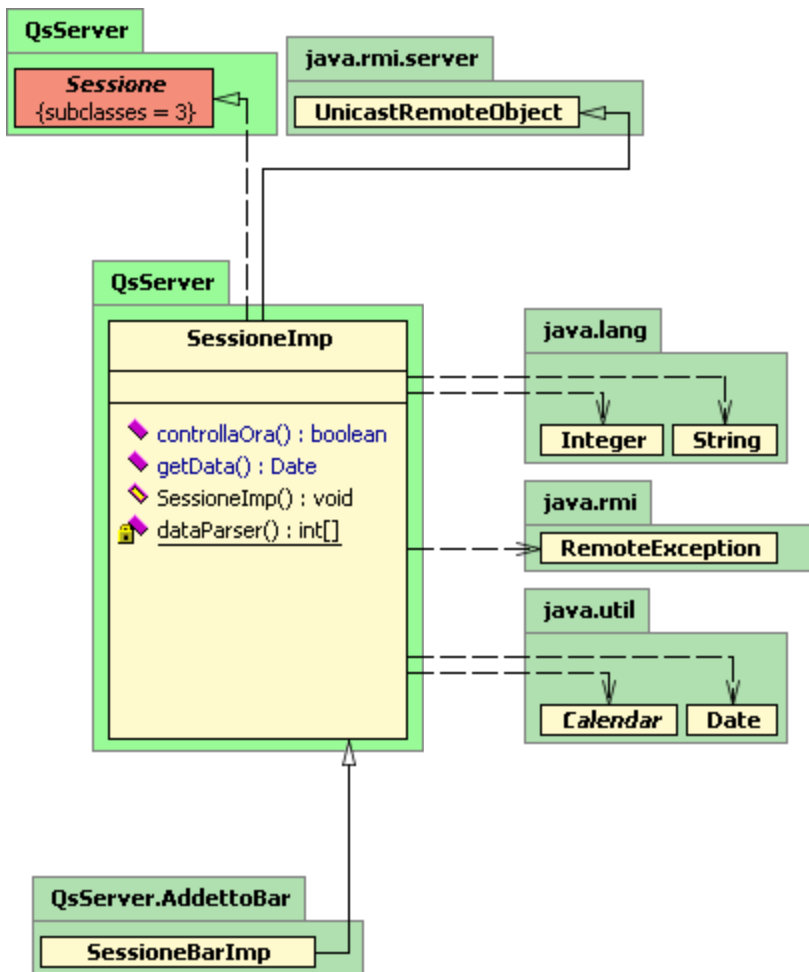
amm - int minuti della prenotazione

ass - int secondi della prenotazione

mills - int millisecondi tempo che deve trascorrere per confermare la prenotazione

Returns:

boolean ritorna true se il tempo trascorso dal momento della prenotazione è maggiore di
mills false altrimenti



QsServer.AddettoBar

Class SessioneBarImp

```

java.lang.Object
  java.rmi.server.RemoteObject
    java.rmi.server.RemoteServer
      java.rmi.server.UnicastRemoteObject
        QsServer.SessioneImp
          QsServer.AddettoBar.SessioneBarImp
    
```

All Implemented Interfaces:

java.rmi.Remote, java.io.Serializable, Sessione, SessioneBar

```

public class SessioneBarImp
  extends SessioneImp
  implements SessioneBar
    
```

Title: SessioneBarImp

Description: Implementa i metodi dell'interfaccia SessioneBar

Version:

1.0

Author:

Alessandro Iacoletti, Pacifico Marta

Field Summary	
(package private) com.borland.dx.sql.dataset.Database	<u>database</u>
(package private) com.borland.dx.sql.dataset.QueryDataSet	<u>qds</u>
(package private) java.util.ResourceBundle	<u>sqlRes</u>

Fields inherited from class java.rmi.server.RemoteObject
Ref

Constructor Summary	
<u>SessioneBarImp</u> (com.borland.dx.sql.dataset.Database database, java.util.ResourceBundle sqlRes) Costruttore pubblico Costruttore pubblico, imposta l'oggetto remoto	

Method Summary	
com.borland.dx.dataset.DataSetData	<u>visualizzaListino</u> () Restituisce tutti i prodotti del bar
com.borland.dx.dataset.DataSetData []	<u>visualizzaOrdinazione</u> () visualizza le ordinazioni che sono state confermate

Methods inherited from class QsServer. <u>SessioneImp</u>
<u>controllaOra</u> , <u>getData</u>

Methods inherited from class java.rmi.server.UnicastRemoteObject
clone, exportObject, exportObject, exportObject, unexportObject

Methods inherited from class java.rmi.server.RemoteServer
getClientHost, getLog, setLog

Methods inherited from class java.rmi.server.RemoteObject

`equals, getRef, hashCode, toString, toStub`

Methods inherited from class `java.lang.Object`

`finalize, getClass, notify, notifyAll, wait, wait, wait`

Methods inherited from interface `QsServer.Sessione`

`controllaOra, getData`

Field Detail

database

`com.borland.dx.sql.dataset.Database database`

sqlRes

`java.util.ResourceBundle sqlRes`

qds

`com.borland.dx.sql.dataset.QueryDataSet qds`

Constructor Detail

SessioneBarImp

```
public SessioneBarImp(com.borland.dx.sql.dataset.Database database,
                      java.util.ResourceBundle sqlRes)
    throws java.rmi.RemoteException
```

Costruttore pubblico Costruttore pubblico, imposta l'oggetto remoto

Parameters:

`database` - Database connessione al database

`sqlRes` - ResourceBundle matrice con le query SQL

Throws:

`java.rmi.RemoteException` - lancia l'eccezione in caso di errore RMI

Method Detail

visualizzaListino

```
public com.borland.dx.dataset.DataSetData visualizzaListino()
```

throws
java.rmi.RemoteException,
com.borland.dx.dataset.DataSetException

Restituisce tutti i prodotti del bar

Specified by:

visualizzaListino in interface SessioneBar

Returns:

DataSetData contiene tutti i prodotti del listino bar

Throws:

java.rmi.RemoteException - lancia l'eccezione in caso di errore RMI

com.borland.dx.dataset.DataSetException - l'eccezione se la query non ha successo

visualizzaOrdinazione

```
public com.borland.dx.dataset.DataSetData[] visualizzaOrdinazione()  
throws  
java.rmi.RemoteException,  
com.borland.dx.dataset.DataSetException
```

visualizza le ordinazioni che sono state confermate

Specified by:

visualizzaOrdinazione in interface SessioneBar

Returns:

DataSetData[] contiene le informazioni di tutte le ordinazioni confermate

Throws:

java.rmi.RemoteException - lancia l'eccezione in caso di errore RMI

com.borland.dx.dataset.DataSetException - lancia l'eccezione se la query non ha successo

QsServer.AddettoBar

Interface SessioneBarGestore

All Superinterfaces:

java.rmi.Remote

All Known Implementing Classes:

SessioneBarGestoreImp

```
public interface SessioneBarGestore  
extends java.rmi.Remote
```

Title: SessioneBarGestore

Description: nterfaccia per la sessione Bar, fornisce i metodi basilari per la gestione del bar

Version:

1.0

Author:

Pizza Luca Ernesto, Vicidomini Vincenzo

Method Summary

Qsclassicomuni.collezioni.QsCollection	<u>caricaCategorie</u> () Restituisce una collezione con le categorie presenti nel database;
void	<u>controllaProdotto</u> (<u>ProdottoBar</u> p) Controlla se i dati riguardanti il prodotto sono corretti
<u>Utente</u>	<u>getDatiUtente</u> () Restituisce i dati dell'utente loggato
void	<u>inserisciProdotto</u> (<u>ProdottoBar</u> p) Inserisce il prodotto contenuto in p
boolean	<u>valida</u> (java.lang.String UID, java.lang.String PWD) Effettua il login dell'utente

Method Detail

valida

```
public boolean valida(java.lang.String UID,  
                      java.lang.String PWD)  
    throws java.rmi.RemoteException,  
           validaException
```

Effettua il login dell'utente

Parameters:

UID - String (User id Utente)

PWD - String (Pasword Utente)

Returns:

boolean Ritorna True se la User Id e la Password Sono corrette

Throws:

java.rmi.RemoteException - lancia l'eccezione in caso di errore RMI
validaException

caricaCategorie

```
public Qsclassicomuni.collezioni.QsCollection caricaCategorie()  
                                                    throws  
java.rmi.RemoteException
```

Restituisce una collezione con le categorie presenti nel database;

Returns:

DataSetData contiene tutte le categorie dei prodotti del bar

Throws:

java.rmi.RemoteException - lancia l'eccezione in caso di errore RMI

controllaProdotto

```
public void controllaProdotto(ProdottoBar p)  
                               throws java.rmi.RemoteException,  
                                     ProdottoBarException
```

Controlla se i dati riguardanti il prodotto sono corretti

Parameters:

p - ProdottoBar è il prodotto da controllare

Throws:

java.rmi.RemoteException - lancia l'eccezione in caso di errore RMI

ProdottoBarException - lancia l'eccezione se uno o più dati all'interno della classe sono errati

inserisciProdotto

```
public void inserisciProdotto(ProdottoBar p)  
                               throws java.rmi.RemoteException
```

Inserisce il prodotto contenuto in p

Parameters:

p - Prodotto riferimento ad un oggetto che contiene i dati relativi ad un prodotto

Throws:

java.rmi.RemoteException - lancia l'eccezione in caso di errore RMI

getDatiUtente

```
public Utente getDatiUtente()  
                               throws java.rmi.RemoteException
```

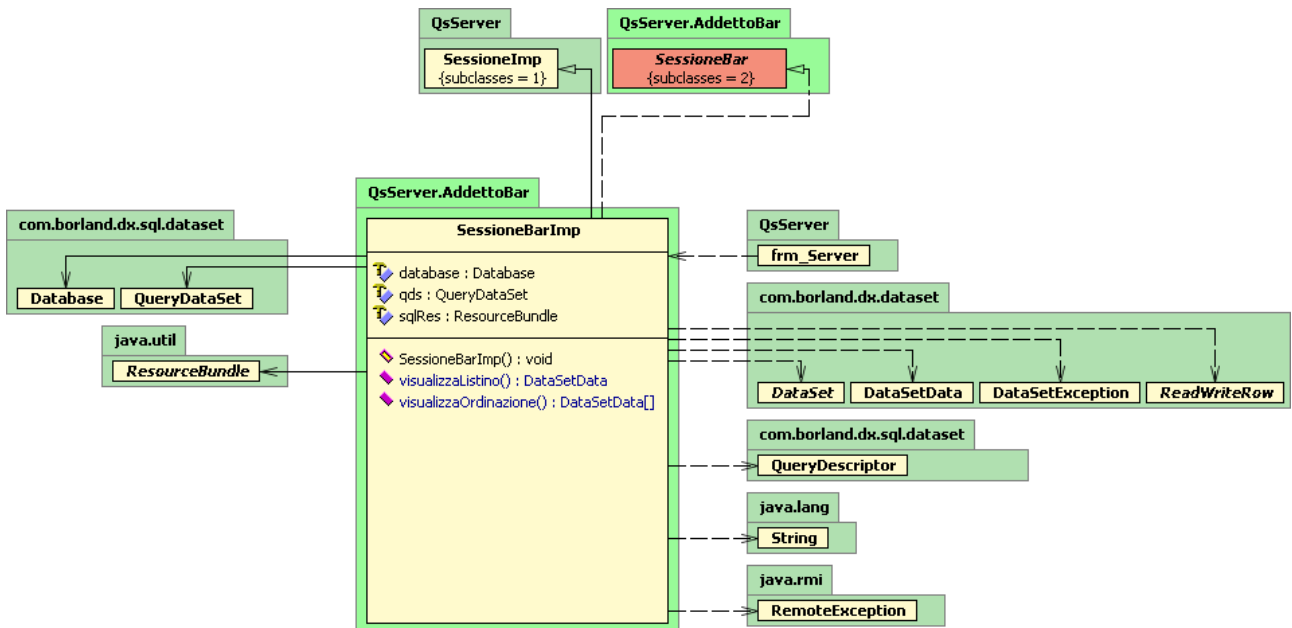
Restituisce i dati dell'utente loggato

Returns:

Utente istanza con i dati dell'utente loggato

Throws:

java.rmi.RemoteException - lancia l'eccezione in caso di errore RMI



QsServer.Ciente

Class SessionClienteImp

java.lang.Object
 java.rmi.server.RemoteObject
 java.rmi.server.RemoteServer

java.rmi.server.UnicastRemoteObject
QsServer.Ciente.SessionClienteImp

All Implemented Interfaces:

java.rmi.Remote, java.io.Serializable, SessionCliente

public class **SessionClienteImp**
 extends java.rmi.server.UnicastRemoteObject
 implements SessionCliente

Title: SessionClienteImp

Description: Progetto QuickService (Gruppo De Sio)

Version:

1.0

Author:

Alessandro Iacoletti, Pacifico Marta

Field Summary

Fields inherited from class java.rmi.server.RemoteObject

ref

Constructor Summary

SessioneClienteImp(com.borland.dx.sql.dataset.Database database, java.util.ResourceBundle sqlRes, Camera camera)
Costruttore pubblico

Method Summary

<u>Camera</u>	<u>getDatiCamera</u> () restituisce i dati della camera
void	<u>ordinazioneBar</u> (int room, Qsclassicomuni.collezioni.QsCollection collect, double tot) invia l'ordinazione al server il metodo è sincronizzato in modo da evitare problemi di concorrenza nella chiamata contemporanea di questo metodo da parte dei client
Qsclassicomuni.collezioni.QsCollection	<u>visualizzaListino</u> (int codCat) Mostra i prodotti disponibili in base alla categoria scelta

Methods inherited from class java.rmi.server.UnicastRemoteObject

clone, exportObject, exportObject, exportObject, unexportObject

Methods inherited from class java.rmi.server.RemoteServer

getClientHost, getLog, setLog

Methods inherited from class java.rmi.server.RemoteObject

equals, getRef, hashCode, toString, toStub

Methods inherited from class java.lang.Object

finalize, getClass, notify, notifyAll, wait, wait, wait

Constructor Detail

SessioneClienteImp

```
public SessioneClienteImp(com.borland.dx.sql.dataset.Database database,  
                           java.util.ResourceBundle sqlRes,  
                           Camera camera)  
    throws java.rmi.RemoteException
```

Costruttore pubblico

Parameters:

camera - Camera camera del cliente che apre la sessione

Throws:

java.rmi.RemoteException - lancia l'eccezione in caso di errore RMI

Method Detail

ordinazioneBar

```
public void ordinazioneBar(int room,  
                           Qsclassicomuni.collezioni.QsCollection collect,  
                           double tot)  
    throws DisponibilitaException
```

invia l'ordinazione al server il metodo è sincronizzato in modo da evitare problemi di concorrenza nella chiamata contemporanea di questo metodo da parte dei client

Specified by:

ordinazioneBar in interface SessioneCliente

Parameters:

room - int numero della camera

collect - QsCollection prodotti dell'ordinazione

tot - double totale dell'ordinazione

Throws:

DisponibilitaException - lancia l'eccezione in caso sia impossibile invare l'ordinazione

visualizzaListino

```
public Qsclassicomuni.collezioni.QsCollection visualizzaListino(int codCat)  
    throws  
java.rmi.RemoteException
```

Mostra i prodotti disponibili in base alla categoria scelta

Specified by:

visualizzaListino in interface SessioneCliente

Parameters:

codCat - int codice categoria

Returns:

QsCollection ritorna un insieme di prodotti

Throws:

java.rmi.RemoteException - lancia l'eccezione in caso di errore RMI

getDatiCamera

```
public Camera getDatiCamera()
```

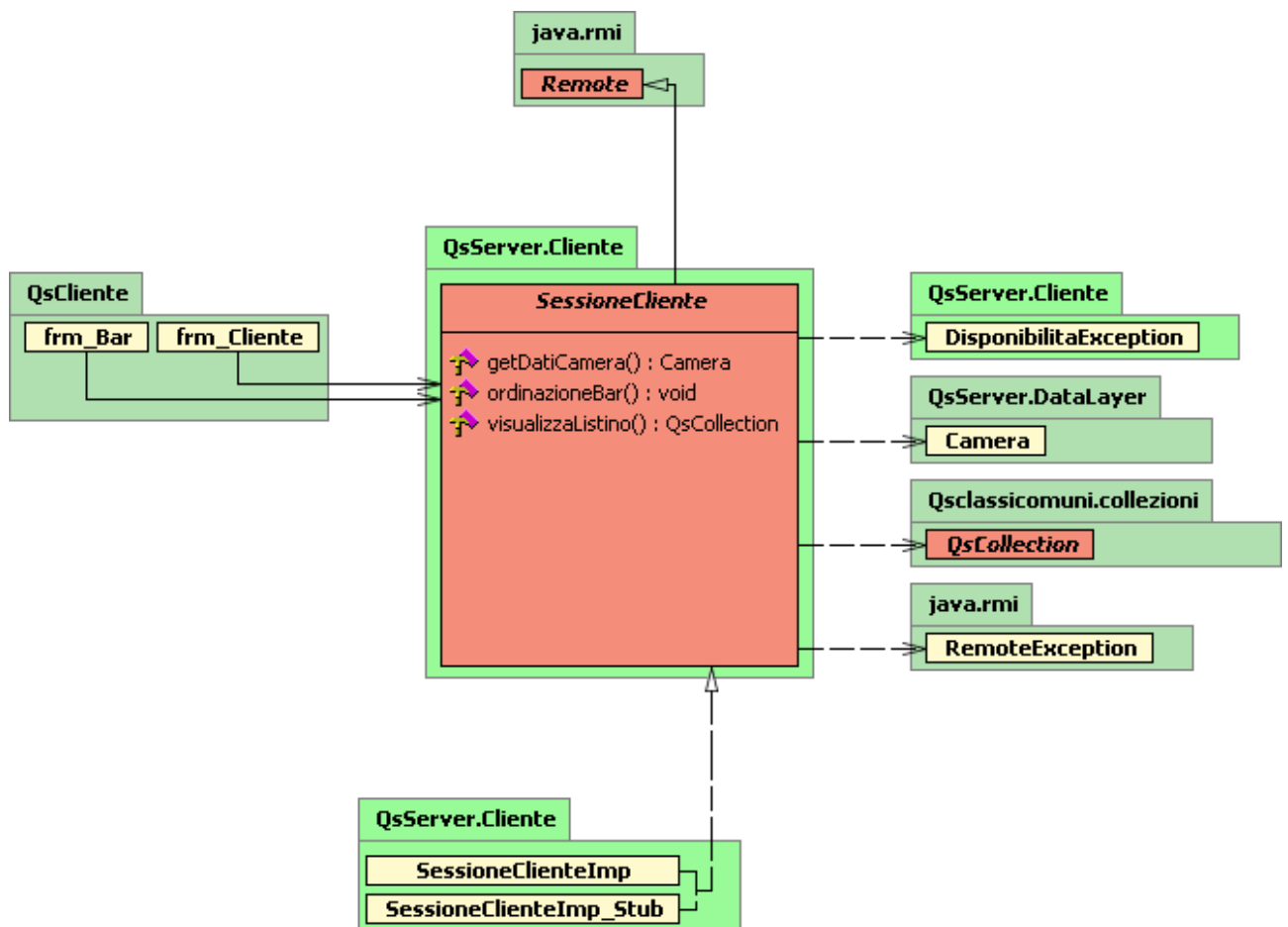
restituisce i dati della camera

Specified by:

getDatiCamera in interface SessioneCliente

Returns:

Camera dati della camera



`QsServer.DataLayer`

Class Camera

`java.lang.Object`

↳ `QsServer.DataLayer.Camera`

All Implemented Interfaces:

`java.io.Serializable`

```
public class Camera
```

```
extends java.lang.Object
```

```
implements java.io.Serializable
```

Title: Camera

Description: Descrive una camera della struttura

Version:

1.0

Author:

Mercurio Antonio

Constructor Summary

Camera()

Costruttore di default

Camera(int numero, double saldo, boolean occupata, boolean pensioneCompleta, int nadulti, int nbambini)

Costruttore pubblico

Method Summary

int **getNadulti**()

Ritorna il numero di posti per adulti disponibili nella camera

int **getNbambini**()

Ritorna il numero di posti per bambini nella camera

int **getNumero**()

recupera il numero della camera

double **getSaldo**()

ritorna il saldo

boolean **isOccupata**()

Verifica se la camera è occupata

boolean **isPensioneCompleta**()

verifica il tipo di pensione

void **load**(int numero, double saldo, boolean occupata, boolean pensioneCompleta, int nadulti, int nbambini)

carica i dati relativi alla camera

void **setNadulti**(int nadulti)

Imposta il numero di posti adulti della camera

void **setNbambini**(int nbambini)

Imposta il numero di posti per bambini della camera

void **setNumero**(int numero)

Imposta il numero della camera

void **setOccupata**(boolean occupata)

Imposta lo stato della camera

void **setPensioneCompleta**(boolean pensioneCompleta)

imposta il tipo di pensione

void **setSaldo**(double saldo)

imposta il saldo

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

Camera

```
public Camera()  
    Costruttore di default
```

Camera

```
public Camera(int numero,  
              double saldo,  
              boolean occupata,  
              boolean pensioneCompleta,  
              int nadulti,  
              int nbambini)  
    Costruttore pubblico
```

Parameters:

numero - int numero camera
saldo - double saldo camera
occupata - boolean stato della camera true se occupata false altrimenti
pensioneCompleta - boolean tipo di pensione true se completa falsa altrimenti
nadulti - int numero di posti adulti della camera
nbambini - int numero di posti bambini della camera

Method Detail

load

```
public void load(int numero,  
                double saldo,  
                boolean occupata,  
                boolean pensioneCompleta,  
                int nadulti,  
                int nbambini)  
    carica i dati relativi alla camera
```

Parameters:

numero - int numero camera
saldo - double saldo camera
occupata - boolean stato della camera true se occupata false altrimenti
pensioneCompleta - boolean tipo di pensione true se completa falsa altrimenti
nadulti - int numero di posti adulti della camera
nbambini - int numero di posti bambini della camera

getNadulti

```
public int getNadulti()
```

Ritorna il numero di posti per adulti disponibili nella camera

Returns:

int posti adulti

setNadulti

public void **setNadulti**(int nadulti)

Imposta il numero di posti adulti della camera

Parameters:

nadulti - int posti adulti

getNbambini

public int **getNbambini**()

Ritorna il numero di posti per bambini nella camera

Returns:

int numero bambini

setNbambini

public void **setNbambini**(int nbambini)

Imposta il numero di posti per bambini della camera

Parameters:

nbambini - int numero bambini

getNumero

public int **getNumero**()

recupera il numero della camera

Returns:

int numero camera

setNumero

public void **setNumero**(int numero)

Imposta il numero della camera

Parameters:

numero - int numero camera

isOccupata

public boolean **isOccupata**()

Verifica se la camera è occupata

Returns:

boolean true se la camera è occupata false altrimenti

setOccupata

public void **setOccupata**(boolean occupata)

Imposta lo stato della camera

Parameters:

occupata - boolean true per se occupata false altrimenti

isPensioneCompleta

```
public boolean isPensioneCompleta()
```

verifica il tipo di pensione

Returns:

boolean se completa true, false altrimenti

setPensioneCompleta

```
public void setPensioneCompleta(boolean pensioneCompleta)
```

imposta il tipo di pensione

Parameters:

pensioneCompleta - boolean true se completa false altrimenti

getSaldo

```
public double getSaldo()
```

ritorna il saldo

Returns:

double totale saldo

setSaldo

```
public void setSaldo(double saldo)
```

imposta il saldo

Parameters:

saldo - double totale saldo

QsServer.DataLayer

Class Categoria

java.lang.Object

└─ QsServer.DataLayer.Categoria

All Implemented Interfaces:

java.io.Serializable

```
public class Categoria
```

```
extends java.lang.Object
```

```
implements java.io.Serializable
```

Title: Categoria

Description: Descrive una categoria di prodotti

Version:

1.0

Author:

Pizza Luca Ernesto, Vicidomini Vincenzo

Field Summary

(package private) java.lang.String	<u>descrizione</u>
(package private) int	<u>idCategoria</u>

Constructor Summary

<u>Categoria</u> () Costruttore di default	
<u>Categoria</u> (java.lang.String descrizione, int id) Costruttore pubblico	

Method Summary

java.lang.String	<u>getDescrizione</u> () Restituisce la descrizione della categoria
int	<u>getIdCategoria</u> () Restituisce l'id della categoria
void	<u>load</u> (int idCategoria, java.lang.String descrizione) Imposta le variabili istanza
void	<u>setDescrizione</u> (java.lang.String descrizione) Setta la descrizione della Categoria
void	<u>setIdCategoria</u> (int idCategoria) Setta l'id della categoria

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Field Detail

descrizione

java.lang.String **descrizione**

idCategoria

int **idCategoria**

Constructor Detail

Categoria

public **Categoria**()

Costruttore di default

Categoria

public **Categoria**(java.lang.String descrizione,
int id)

Costruttore pubblico

Parameters:

descrizione - String descrizione della categoria

id - int identificatore della categoria

Method Detail

load

public void **load**(int idCategoria,
java.lang.String descrizione)

Imposta le variabili istanza

Parameters:

idCategoria - int identificatore della categoria

descrizione - String descrizione della categoria

getDescrizione

public java.lang.String **getDescrizione**()

Restituisce la descrizione della categoria

Returns:

String descrizione categoria

getIdCategoria

public int **getIdCategoria**()

Restituisce l'id della categoria

Returns:

int identificatore categoria

setDescrizione

public void **setDescrizione**(java.lang.String descrizione)

Setta la descrizione della Categoria

Parameters:

descrizione - String descrizione categoria

setIdCategoria

public void **setIdCategoria**(int idCategoria)

Setta l'id della categoria

Parameters:

idCategoria - int identificatore categoria

QsServer.DataLayer

Class DettaglioPrenotazioneBar

java.lang.Object

└─ QsServer.DataLayer.DettaglioPrenotazioneBar

All Implemented Interfaces:

java.io.Serializable

public class **DettaglioPrenotazioneBar**

extends java.lang.Object

implements java.io.Serializable

Title: DettaglioPrenotazioneBar

Description: Descrive elemento di una prenotazione

Version:

1.0

Author:

Vicidomini Vincenzo

Field Summary

(package private) java.lang.String	<u>descrizione</u>
(package private) int	<u>idProdotto</u>
(package private) java.lang.String	<u>note</u>
(package private) double	<u>prezzo</u>
(package private) int	<u>quantita</u>

Constructor Summary

<u>DettaglioPrenotazioneBar</u> () Costruttore di default

<u>DettaglioPrenotazioneBar</u> (int idProdotto, java.lang.String descrizione, int quantita, double prezzo, java.lang.String note) Costruttore pubblico

Method Summary

java.lang.String	<u>getDescrizione</u> () Restituisce la descrizione del prodotto ordinato
int	<u>getIdProdotto</u> () Restituisce l'id del prodotto ordinato
java.lang.String	<u>getNote</u> () Restituisce le note del prodotto ordinato
double	<u>getPrezzo</u> () Restituisce il prezzo del prodotto ordinato
int	<u>getQuantita</u> () Restituisce la quantita del prodotto ordinato
void	<u>load</u> (int idProdotto, java.lang.String descrizione, int quantita, double prezzo, java.lang.String note) Carica i dati
void	<u>setDescrizione</u> (java.lang.String descrizione) Setta la descrizione del prodotto ordinato
void	<u>setIdProdotto</u> (int idProdotto) Setta l'id del prodotto ordinato
void	<u>setNote</u> (java.lang.String note) Setta le note del prodotto ordinato
void	<u>setPrezzo</u> (double prezzo) Setta il prezzo del prodotto ordinato
void	<u>setQuantita</u> (int quantita) Setta la quantita del prodotto ordinato

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Field Detail

idProdotto

int **idProdotto**

quantita

int **quantita**

descrizione

java.lang.String **descrizione**

note

java.lang.String **note**

prezzo

double **prezzo**

Constructor Detail

DettaglioPrenotazioneBar

```
public DettaglioPrenotazioneBar()  
    Costruttore di default
```

DettaglioPrenotazioneBar

```
public DettaglioPrenotazioneBar(int idProdotto,  
                                java.lang.String descrizione,  
                                int quantita,  
                                double prezzo,  
                                java.lang.String note)
```

Costruttore pubblico

Parameters:

idProdotto - int idProdotto

descrizione - String descrizione prodotto

quantita - int quantità prodotto

prezzo - double prezzo totale (prezzo unitario prodotto * quantità)

note - String note opzionali

Method Detail

load

```
public void load(int idProdotto,  
                java.lang.String descrizione,  
                int quantita,  
                double prezzo,  
                java.lang.String note)
```

Carica i dati

Parameters:

idProdotto - int identificatore prodotto

descrizione - String descrizione

quantita - int quantità

prezzo - double prezzo
note - String note opzionali

getDescription

public java.lang.String **getDescription()**
Restituisce la descrizione del prodotto ordinato
Returns:
String descrizione prodotto

setDescription

public void **setDescription**(java.lang.String descrizione)
Setta la descrizione del prodotto ordinato
Parameters:
descrizione - String descrizione prodotto

getIdProdotto

public int **getIdProdotto()**
Restituisce l'id del prodotto ordinato
Returns:
int descrizione prodotto

setIdProdotto

public void **setIdProdotto**(int idProdotto)
Setta l'id del prodotto ordinato
Parameters:
idProdotto - int identificatore prodotto

getNote

public java.lang.String **getNote()**
Restituisce le note del prodotto ordinato
Returns:
String note opzionali

setNote

public void **setNote**(java.lang.String note)
Setta le note del prodotto ordinato
Parameters:
note - String note opzionali

getPrezzo

public double **getPrezzo()**
Restituisce il prezzo del prodotto ordinato
Returns:

double prezzo totale

setPrezzo

public void **setPrezzo**(double prezzo)

Setta il prezzo del prodotto ordinato

Parameters:

prezzo - double prezzo totale

getQuantita

public int **getQuantita**()

Restituisce la quantita del prodotto ordinato

Returns:

int quantità

setQuantita

public void **setQuantita**(int quantita)

Setta la quantita del prodotto ordinato

Parameters:

quantita - int quantità

QsServer.DataLayer

Class Prenotazione

java.lang.Object

└─ QsServer.DataLayer.Prenotazione

All Implemented Interfaces:

java.io.Serializable

public class **Prenotazione**

extends java.lang.Object

implements java.io.Serializable

Title: Prenotazione

Description: Descrive una prenotazione

Version:

1.0

Author:

Pizza Ernesto

Constructor Summary

Prenotazione(int idP, int numero, java.lang.String data, char tipo, float costo)

Costruttore pubblico

Method Summary

double	getCosto () Ritorna il costo totale della prenotazione
java.lang.String	getData () Ritorna la data della prenotazione
int	getIdPrenotazione () Restituisce l'identificatore univoco della prenotazione
int	getNumeroCamera () ritorna il numero della camera
char	getTipo () Ritorna il tipo della prenotazione
void	setCosto (double costo) Imposta il costo della prenotazione
void	setData (java.lang.String data) Imposta la data della prenotazione
void	setIdPrenotazione (int idPrenotazione) Imposta l'identificatore univoco della prenotazione
void	setNumeroCamera (int numeroCamera) Imposta il numero della camera
void	setTipo (char tipo) Imposta il tipo della prenotazione

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

Prenotazione

```
public Prenotazione(int idP,  
                    int numero,
```

```
        java.lang.String data,  
        char tipo,  
        float costo)
```

Costruttore pubblico

Parameters:

idP - int identificatore prenotazione
numero - int numero camera
data - String data della prenotazione
tipo - char tipo della prenotazione
costo - float costo totale prenotazione

Method Detail

getCosto

```
public double getCosto()
```

Ritorna il costo totale della prenotazione

Returns:

double costo totale

setCosto

```
public void setCosto(double costo)
```

Imposta il costo della prenotazione

Parameters:

costo - double costo totale

getData

```
public java.lang.String getData()
```

Ritorna la data della prenotazione

Returns:

String data prenotazione

setData

```
public void setData(java.lang.String data)
```

Imposta la data della prenotazione

Parameters:

data - String data

getIdPrenotazione

```
public int getIdPrenotazione()
```

Restituisce l'identificatore univoco della prenotazione

Returns:

int identificatore prenotazione

setIdPrenotazione

```
public void setIdPrenotazione(int idPrenotazione)
```

Imposta l'identificatore univoco della prenotazione

Parameters:

idPrenotazione - int

setNumeroCamera

public void **setNumeroCamera**(int numeroCamera)

Imposta il numero della camera

Parameters:

numeroCamera - int numero camera

setTipo

public void **setTipo**(char tipo)

Imposta il tipo della prenotazione

Parameters:

tipo - char tipo prenotazione

getTipo

public char **getTipo**()

Ritorna il tipo della prenotazione

Returns:

char tipo prenotazione

getNumeroCamera

public int **getNumeroCamera**()

ritorna il numero della camera

Returns:

int numero camera

QsServer.DataLayer

Class ProdottoBar

java.lang.Object

└─ QsServer.DataLayer.ProdottoBar

All Implemented Interfaces:

java.io.Serializable

public class **ProdottoBar**

extends java.lang.Object

implements java.io.Serializable

Title: ProdottoBar

Description: Descrive un prodotto del bar

Version:

1.0

Author:

Mercurio Antonio

Constructor Summary

ProdottoBar()

Costruttore di default

ProdottoBar(int codCategoria, java.lang.String descrizione, double prezzo, int disponibilita)

Costruttore avanzato

Method Summary

int	<u>getCodCategoria</u> () ritorna il codice della categoria
java.lang.String	<u>getDescrizione</u> () ritorna la descrizione del prodotto
int	<u>getDisponibilita</u> () ritorna la disponibilità di un prodotto
int	<u>getIdProdotto</u> () ritorna l'identificatore univoco del prodotto
double	<u>getPrezzo</u> () ritorna il prezzo unitario del prodotto
void	<u>load</u> (int idProdotto, int codCategoria, java.lang.String descrizione, double prezzo, int disponibilita) Carica i dati dall'esterno
void	<u>setCodCategoria</u> (int codCategoria) imposta codice categoria
void	<u>setDescrizione</u> (java.lang.String descrizione) imposta descrizione prodotto
void	<u>setDisponibilita</u> (int disponibilita) imposta la disponibilità di un prodotto
void	<u>setIdProdotto</u> (int idProdotto) imposta l'identificatore del prodotto
void	<u>setPrezzo</u> (double prezzo) imposta il prezzo unitario del prodotto

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

ProdottoBar

```
public ProdottoBar()  
    Costruttore di default
```

ProdottoBar

```
public ProdottoBar(int codCategoria,  
                    java.lang.String descrizione,  
                    double prezzo,  
                    int disponibilita)
```

Costruttore avanzato

Parameters:

codCategoria - int codice categoria
descrizione - String descrizione prodotto
prezzo - double prezzo unitario prodotto
disponibilita - int disponibilità (-1 per disponibilità ILLIMITATA)

Method Detail

load

```
public void load(int idProdotto,  
                 int codCategoria,  
                 java.lang.String descrizione,  
                 double prezzo,  
                 int disponibilita)
```

Carica i dati dall'esterno

Parameters:

idProdotto - int nome del prodotto
codCategoria - int codice categoria
descrizione - String descrizione prodotto
prezzo - double prezzo unitario prodotto
disponibilita - int disponibilità (-1 per disponibilità ILLIMITATA)

getCodCategoria

```
public int getCodCategoria()  
    ritorna il codice della categoria
```

Returns:

int codice categoria

setCodCategoria

```
public void setCodCategoria(int codCategoria)  
    imposta codice categoria
```

Parameters:

codCategoria - int codice categoria

getDisponibilita

```
public int getDisponibilita()  
    ritorna la disponibilità di un prodotto
```


Returns:
int disponibilità

setDisponibilita

public void **setDisponibilita**(int disponibilita)
imposta la disponibilità di un prodotto

Parameters:
disponibilita - int disponibilità prodotto

getDescrizione

public java.lang.String **getDescrizione**()
ritorna la descrizione del prodotto

Returns:
String descrizione prodotto

setDescrizione

public void **setDescrizione**(java.lang.String descrizione)
imposta descrizione prodotto

Parameters:
descrizione - String descrizione prodotto

getPrezzo

public double **getPrezzo**()
ritorna il prezzo unitario del prodotto

Returns:
double prezzo prodotto

setPrezzo

public void **setPrezzo**(double prezzo)
imposta il prezzo unitario del prodotto

getIdProdotto

public int **getIdProdotto**()
ritorna l'identificatore univoco del prodotto

Returns:
int identificatore prodotto

setIdProdotto

public void **setIdProdotto**(int idProdotto)
imposta l'identificatore del prodotto

Parameters:
idProdotto - int identificatore prodotto

QsServer.DataLayer

Class Utente

java.lang.Object

└─ QsServer.DataLayer.Utente

All Implemented Interfaces:

java.io.Serializable

public class **Utente**

extends java.lang.Object

implements java.io.Serializable

Title: Utente

Description: Descrive l'utente che si logga al sistema

Version:

1.0

Author:

Iacoletti Alessandro

Field Summary

(package private) java.lang.String	<u>cognome</u>
---------------------------------------	-----------------------

(package private) java.lang.String	<u>login</u>
---------------------------------------	---------------------

(package private) java.lang.String	<u>nome</u>
---------------------------------------	--------------------

(package private) java.lang.String	<u>pwd</u>
---------------------------------------	-------------------

Constructor Summary

<u>Utente</u> ()	
Costruttore pubblico di default	

Method Summary

java.lang.String	<u>getCognome</u> ()	ritorna il cognome dell'utente
------------------	------------------------------	--------------------------------

java.lang.String	<u>getLogin</u> ()	Ritorna il login name dell'utente
------------------	----------------------------	-----------------------------------

java.lang.String	<u>getNome</u> ()	
------------------	---------------------------	--

	ritorna il nome dell'utente
java.lang.String	<u>getPwd</u> () ritorna la password dell'utente
void	<u>setCognome</u> (java.lang.String cognome) imposta il cognome dell'utente
void	<u>setLogin</u> (java.lang.String login) imposta il login name
void	<u>setNome</u> (java.lang.String nome) imposta il nome utente
void	<u>setPwd</u> (java.lang.String pwd) imposta la password dell'utente

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Field Detail

nome

java.lang.String **nome**

cognome

java.lang.String **cognome**

login

java.lang.String **login**

pwd

java.lang.String **pwd**

Constructor Detail

Utente

public **Utente**()
Costruttore pubblico di default

Method Detail

getCognome

public java.lang.String **getCognome**()

ritorna il cognome dell'utente

Returns:

String cognome utente

getLogin

public java.lang.String **getLogin**()

Ritorna il login name dell'utente

Returns:

String login name

getNome

public java.lang.String **getNome**()

ritorna il nome dell'utente

Returns:

String nome utente

getPwd

public java.lang.String **getPwd**()

ritorna la password dell'utente

Returns:

String password

setCognome

public void **setCognome**(java.lang.String cognome)

imposta il cognome dell'utente

Parameters:

cognome - String cognome

setLogin

public void **setLogin**(java.lang.String login)

imposta il login name

Parameters:

login - String login name

setNome

public void **setNome**(java.lang.String nome)

imposta il nome utente

Parameters:

nome - String nome dell'utente

setPwd

public void **setPwd**(java.lang.String pwd)

imposta la password dell'utente

Parameters:

pwd - String password utente

Qsclassicomuni.adapters

Class QsCollectionAdapter

java.lang.Object

└ java.util.AbstractCollection

└ java.util.AbstractList

└ java.util.ArrayList

└ Qsclassicomuni.adapters.QsCollectionAdapter

All Implemented Interfaces:

java.lang.Cloneable, java.util.Collection, java.util.List, QsCollection,
java.util.RandomAccess, java.io.Serializable

public class **QsCollectionAdapter**

extends java.util.ArrayList

implements QsCollection, java.io.Serializable

Title: QsCollectionAdapter

Description: Questa Classe invece ha il ruolo di adapter cioè fa da tramite tra la classe target e la classe Adaptee, ovvero la classe che fornisce l'implementazione al target, che nel nostro caso è la classe ArrayList. L'utilizzo di questa classe rende l'applicazione molto versatile e malleabile è possibile cambiare in ogni momento la struttura dati della collezione senza dover apportare modifiche in tutte le classi client della collezione stessa. Ogni invocazione di metodi della classe Adapter vengono indirizzate tutte ai metodi della classe adaptee

Version:

1.0

Author:

Pacifico Marta, Antonio Mercurio

Nested Class Summary

Nested classes inherited from class java.util.ArrayList

Field Summary

Fields inherited from class java.util.AbstractList

modCount

Constructor Summary

QsCollectionAdapter()

Costruttore pubblico

Method Summary

void aggiungi(java.lang.Object o)
Aggiunge un elemento alla collection

int dimensione()
Dimensione cioè numero di elementi all'interno della collection

void modifica(int i, java.lang.Object o)
Modifica un elemento all'interno della collection

java.lang.Object preleva(int i)
ritorna un elemento della collection

void rimuovi(int i)
Elimina un elemento dalla collection

void svuota(java.util.Collection c)
Elimina tutti gli elementi dalla collection

Methods inherited from class java.util.ArrayList

add, add, addAll, addAll, clear, clone, contains, ensureCapacity, get, indexOf, isEmpty, lastIndexOf, remove, removeRange, set, size, toArray, toArray, trimToSize

Methods inherited from class java.util.AbstractList

equals, hashCode, iterator, listIterator, listIterator, subList

Methods inherited from class java.util.AbstractCollection

containsAll, remove, removeAll, retainAll, toString

Methods inherited from class java.lang.Object

finalize, getClass, notify, notifyAll, wait, wait, wait

Methods inherited from interface java.util.List

containsAll, equals, hashCode, iterator, listIterator, listIterator, remove, removeAll, retainAll, subList

Constructor Detail

QsCollectionAdapter

```
public QsCollectionAdapter()  
    Costruttore pubblico
```

Method Detail

aggiungi

```
public void aggiungi(java.lang.Object o)  
    Aggiunge un elemento alla collection  
Specified by:  
    aggiungi in interface QsCollection  
Parameters:  
    o - Object elemento da aggiungere
```

dimensione

```
public int dimensione()  
    Dimensione cioè numero di elementi all'interno della collection  
Specified by:  
    dimensione in interface QsCollection  
Returns:  
    int dimensione
```

preleva

```
public java.lang.Object preleva(int i)  
    ritorna un elemento della collection  
Specified by:  
    preleva in interface QsCollection  
Parameters:
```

i - int indice dell'elemento da prelevare

Returns:

Object elemento della collection

modifica

```
public void modifica(int i,  
                     java.lang.Object o)  
    Modifica un elemento all'interno della collection
```

Specified by:

modifica in interface QsCollection

Parameters:

i - int indice dell'elemento da modificare

o - Object elemento da modificare

rimuovi

```
public void rimuovi(int i)  
    Elimina un elemento dalla collection
```

Specified by:

rimuovi in interface QsCollection

Parameters:

i - int indice dell'elemento da rimuovere

svuota

```
public void svuota(java.util.Collection c)  
    Elimina tutti gli elementi dalla collection
```

Specified by:

svuota in interface QsCollection

Parameters:

c - Collection collection vuota

Qsclassicomuni.collezioni

Class CollectionListino

```
java.lang.Object  
├─ com.borland.dx.dataset.ReadRow  
│   └─ com.borland.dx.dataset.ReadWriteRow  
│       └─ com.borland.dx.dataset.DataSet  
│           └─ com.borland.dx.dataset.StorageDataSet  
│               └─ com.borland.dx.sql.dataset.QueryDataSet  
│                   └─ Qsclassicomuni.collezioni.CollectionListino
```

All Implemented Interfaces:

com.borland.dx.dataset.AccessListener, com.borland.dx.dataset.ColumnDesigner,
com.borland.dx.dataset.Designable, java.util.EventListener,
com.borland.dx.dataset.MasterNavigateListener, java.io.Serializable,
com.borland.dx.dataset.StatusListener

```
public class CollectionListino
extends com.borland.dx.sql.dataset.QueryDataSet
implements java.io.Serializable
```

Title: CollectionListino

Description: Collezione di oggetti di tipo prodottoBar che compongono il listino

Version:

1.0

Author:

Vicidomini Vincenzo

Field Summary

Fields inherited from class com.borland.dx.dataset.StorageDataSet

Constructor Summary

CollectionListino()

Method Summary

java.lang.String	<u>getCategoria()</u> ritorna la categoria dell'elemento
java.lang.String	<u>getDescrizione()</u> Ritorna la descrizione dall'elemento nella collection
int	<u>getDisponibilita()</u> ritorna la disponibilità del prodotto, -1 per ILLIMITATO
double	<u>getPrezzo()</u> ritorna il prezzo
void	<u>setDescrizione</u> (java.lang.String descrizione) Imposta la descrizione

Methods inherited from class com.borland.dx.sql.dataset.QueryDataSet

closeStatement, executeQuery, getDatabase, getOriginalQueryString,

getParameterRow, getQuery, getQueryString, isAccumulateResults, refetchRow, refresh, refreshSupported, saveChanges, saveChangesSupported, setAccumulateResults, setProvider, setQuery

Methods inherited from class com.borland.dx.dataset.StorageDataSet

addCalcAggFieldsListener, addCalcFieldsListener, addColumn, addColumn, addColumn, addColumnChangeListener, addEditListener, addForeignKey, addLoadListener, addLoadRowListener, addUniqueColumn, cancelLoading, cancelOperation, changeColumn, changesPending, cloneColumns, cloneDataSetStructure, closeProvider, deleteDuplicates, dropAllIndexes, dropColumn, dropColumn, dropIndex, empty, endLoading, freeAllIndexes, getCalcAggFieldsListener, getCalcFieldsListener, getDataFile, getDeletedRowCount, getDeletedRows, getDuplicates, getForeignKeys, getInsertedRowCount, getInsertedRows, getLocale, getMaxDesignRows, getMaxResolveErrors, getMaxRows, getMetaDataUpdate, getNeedsRestructure, getOriginalRow, getProvider, getReferenceForeignKeys, getResolveOrder, getResolver, getSchemaName, getStore, getStoreClassFactory, getStoreName, getTableName, getUpdatedRowCount, getUpdatedRows, hasRowIds, indexExists, isReadOnly, isResolvable, loadRow, loadRow, moveColumn, postAllDataSets, provideMoreData, recalc, removeCalcAggFieldsListener, removeCalcFieldsListener, removeColumnChangeListener, removeEditListener, removeForeignKey, removeLoadListener, removeLoadRowListener, reset, resetPendingStatus, resetPendingStatus, restructure, setAllRowIds, setColumns, setDataFile, setForeignKeys, setLocale, setMaxDesignRows, setMaxResolveErrors, setMaxRows, setMetaDataUpdate, setReadOnly, setResolvable, setResolveOrder, setResolver, setRowId, setSchemaName, setStore, setStoreClassFactory, setStoreName, setTableName, startLoading, startLoading, startLoading

Methods inherited from class com.borland.dx.dataset.DataSet

accessChange, addAccessListener, addDataChangeListener, addMasterNavigateListener, addNavigationListener, addOpenListener, addRow, addRowFilterListener, addRowReturnInternalRow, addStatusListener, allocateValues, atFirst, atLast, cancel, canNavigate, canSet, clearStatus, cloneDataSetView, close, columnIsVisible, deleteAllRows, deleteRow, dittoRow, dittoRow, dropIndex, editRow, emptyAllRows, emptyRow, enableDataSetEvents, first, format, format, getDataRow, getDataRow, getDetail, getDetails, getDisplayVariant, getInternalRow, getLastColumnVisited, getMasterLink, getRow, getRowCount, getRowFilterListener, getSort, getStatus, getStorageDataSet, getVariant, getVariant, goToClosestRow, goToInternalRow, goToRow, goToRow, hasDetail, hasValidations, inBounds, insertRow, interactiveLocate, isDetailDataSetWithFetchAsNeeded, isDisplayErrors, isEditable, isEditing, isEditingNewRow, isEmpty, isEnableDelete, isEnableInsert, isEnableUpdate, isModified, isModified, isNew, isOpen, isSortable, last, locate, lookup, masterNavigated, masterNavigating, moveRow, next, open, openDetails, post, prior, refilter, removeAccessListener, removeDataChangeListener, removeMasterNavigateListener, removeNavigationListener, removeOpenListener, removeRowFilterListener, removeStatusListener, resetInBounds, row, rowCount, saveChanges, setDefaultValues, setDefaultValues, setDisplayErrors, setDisplayVariant, setEditable, setEnableDelete, setEnableInsert, setEnableUpdate, setLastColumnVisited, setMasterLink, setSort, startEdit, startEditCheck, statusMessage, statusMessage, toggleViewOrder, updateRow, validate, validate

Methods inherited from class com.borland.dx.dataset.ReadWriteRow

clearValues, requiredColumnsCheck, requiredColumnsCheck, setAssignedNull, setAssignedNull, setBigDecimal, setBigDecimal, setBinaryStream, setBinaryStream, setBoolean, setBoolean, setByte, setByte, setByteArray, setByteArray, setDate, setDate, setDate, setDate, setDefault, setDefault, setDouble, setDouble, setFloat, setFloat, setInputStream, setInputStream, setInt, setInt, setLong, setLong, setObject, setObject, setShort, setShort, setString, setString, setTime, setTime, setTime, setTime, setTimestamp, setTimestamp, setTimestamp, setTimestamp, setUnassignedNull, setUnassignedNull, setVariant, setVariant

Methods inherited from class com.borland.dx.dataset.ReadRow

columnCount, copyTo, copyTo, equals, findDifference, findModified, findOrdinal, getArrayLength, getBigDecimal, getBigDecimal, getBinaryStream, getBinaryStream, getBoolean, getBoolean, getByte, getByte, getByteArray, getByteArray, getColumn, getColumn, getColumnCount, getColumnNames, getColumns, getDate, getDate, getDouble, getDouble, getFloat, getFloat, getInputStream, getInputStream, getInt, getInt, getLong, getLong, getObject, getObject, getShort, getShort, getString, getString, getStringPadded, getStringPadded, getTime, getTime, getTimestamp, getTimestamp, getVariant, getVariant, getVariants, hasColumn, isAssignedNull, isAssignedNull, isCompatibleList, isModified, isNull, isNull, isUnassignedNull, isUnassignedNull, toString

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Methods inherited from interface com.borland.dx.dataset.ColumnDesigner

getColumn, getColumnCount, hasColumn

Constructor Detail

CollectionListino

public **CollectionListino**()

Method Detail

getDescription

public java.lang.String **getDescription**()

Ritorna la descrizione dall'elemento nella collection

Returns:

String descrizione elemento

getCategoria

```
public java.lang.String getCategoria()
```

ritorna la categoria dell'elemento

Returns:

String categoria

getPrezzo

```
public double getPrezzo()
```

ritorna il prezzo

Returns:

double prezzo

getDisponibilita

```
public int getDisponibilita()
```

ritorna la disponibilità del prodotto, -1 per ILLIMITATO

Returns:

int disponibilità

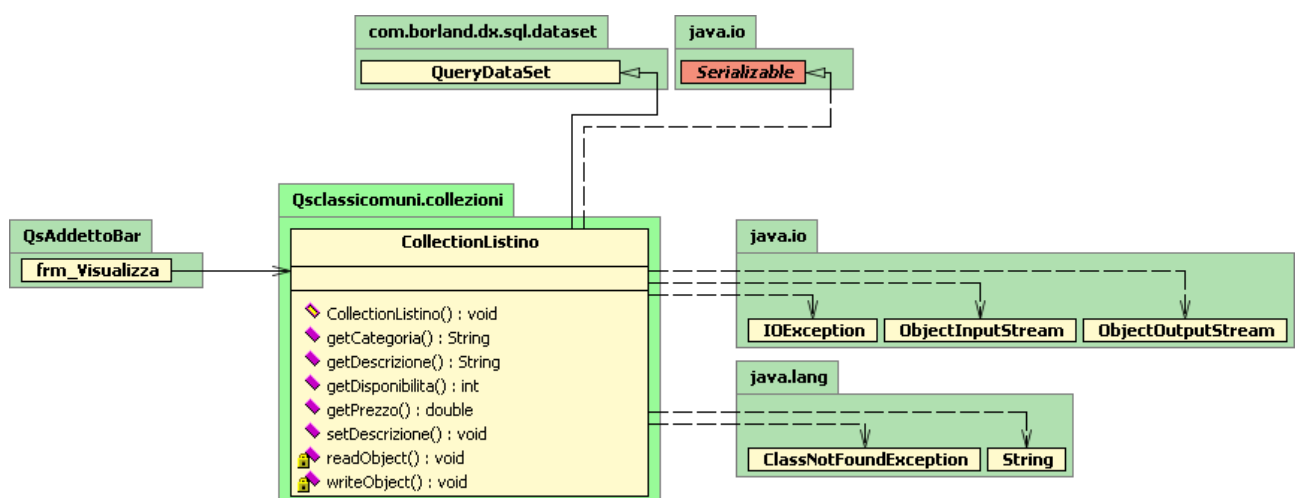
setDescrizione

```
public void setDescrizione(java.lang.String descrizione)
```

Imposta la descrizione

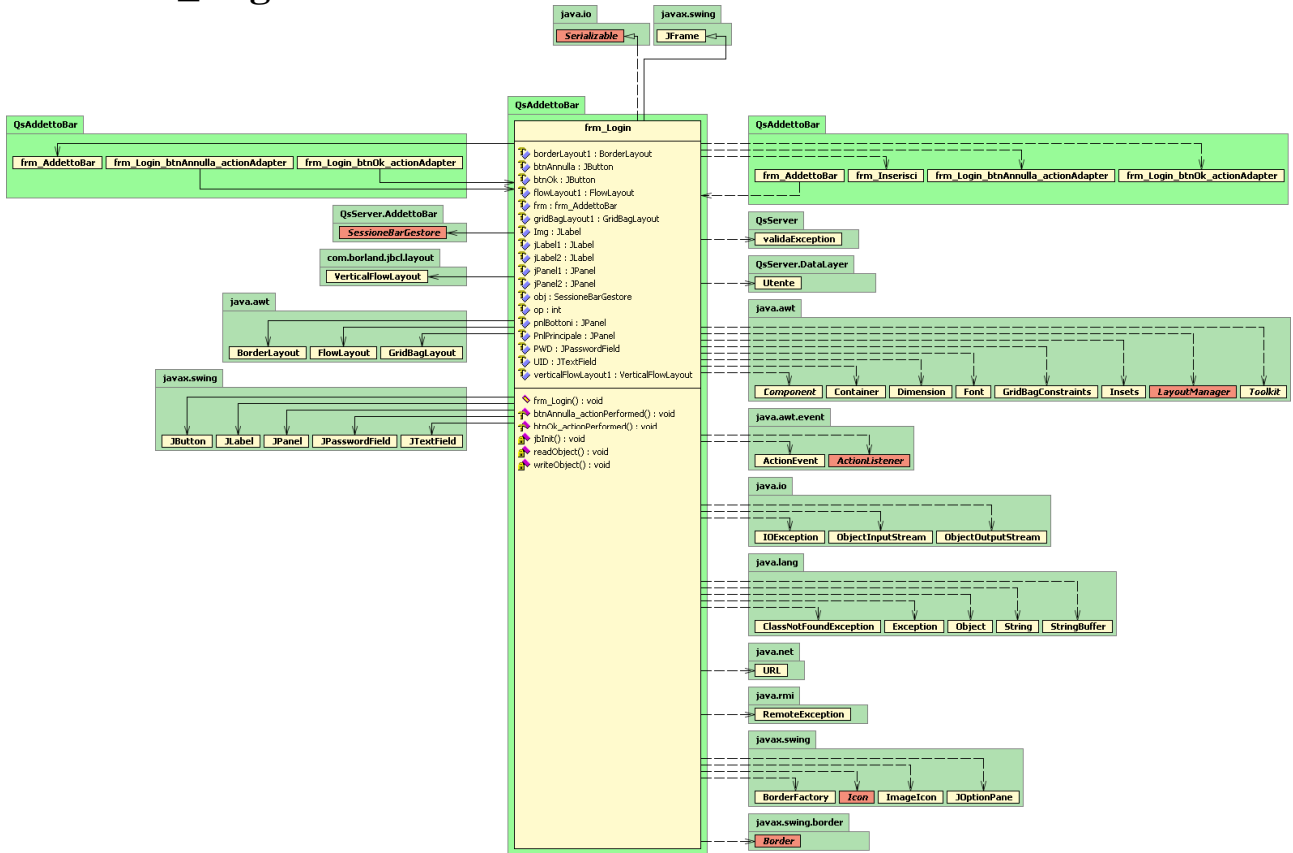
Parameters:

descrizione - String descrizione elemento



QsAddettoBar

QsCliente
QsAddettoBar
Class frm_Login



QsCliente

Class frm_Bar

