



System Design Document

urCoach

Partecipante	Matricola
Salvatore Fasano	0512105068
Mariantonia Candela	0512105374

Sommario

SOMMARIO	2
1. INTRODUZIONE	3
1.1 OBIETTIVI DEL SISTEMA	3
1.2 DESIGN GOALS	4
1.2.1 Trade-offs	5
1.3 DEFINIZIONI, ACRONIMI ED ABBREVIAZIONI	5
1.4 RIFERIMENTI	5
1.5 PANORAMICA	5
2. ARCHITETTURA DI SISTEMI SIMILI	6
3. ARCHITETTURA DEL SISTEMA PROPOSTO	7
3.1 PANORAMICA	7
3.2 SCOMPOSIZIONE IN SOTTOSISTEMI	8
3.3 HARDWARE/SOFTWARE MAPPING	10
3.4 GESTIONE DEI DATI PERSISTENTI	11
3.5 CONTROLLO DEGLI ACCESSI E SICUREZZA	12
3.6 CONTROLLO FLUSSO GLOBALE DEL SISTEMA	14
3.7 BOUNDARY CONDITION	15
3.7.1 Startup sistema	15
3.7.2 Shutdown Sistema	15
3.7.3 Fallimento	15
4. SUBSYSTEM SERVICES	16

1. Introduzione

1.1 Obiettivi del sistema

Sempre più atleti, amanti del body building, ricercano personal trainer con cui iniziare un percorso di miglioramento del proprio fisico con scopo amatoriale o di partecipare a delle gare. Tuttavia c'è una limitazione notevole: potrebbero non essere presenti pt in un determinato paese o semplicemente si preferisce essere seguiti da un pt che si trova distante. L'obiettivo di urCoach è questo: permettere ad atleti di entrare in contatto con diversi pt sparsi sul territorio e poter selezionare quello che più fa al proprio caso (perché specializzato in una determinata area, es. dimagrimento). urCoach inoltre fa una selezione in ingresso dei pt, in modo da permettere l'accesso alla piattaforma solo a coloro che sono davvero preparati valutando il loro curriculum. Ogni pt potrà, una volta accettato dai recruiter, pubblicare i propri servizi sulla piattaforma. Gli atleti potranno invece registrarsi sulla piattaforma, ricercare vari pacchetti ed acquistare i pacchetti messi in vendita.

1.2 Design goals

Illustriamo nella seguente tabella gli obiettivi di design per il sistema e le relative priorità (a numeri più bassi corrispondono priorità più elevate). Per ogni obiettivo riportiamo anche l'origine, facendo riferimento, in particolare, all'identificativo del requisito non funzionale ad esso associato.

Priorità	ID	Descrizione	Categoria	Origine
1	DG_1	Robustezza: Il sistema deve sopravvivere agli input errati degli utenti. In un caso del genere il sistema non deve accettare l'input e notificare l'utente invitandolo a correggere l'errore	Reliability	RNF_R3
3	DG_2	Sicurezza: Il sistema deve comunicare tramite protocollo HTTPS in modo da assicurare una maggiore sicurezza, in particolar modo nel contesto del pagamento	Reliability	RNF_R1 RNF_R2
2	DG_3	Usabilità: Il sistema deve essere facile da apprendere ed intuitivo da utilizzare senza necessariamente consultare la documentazione. I contenuti dovranno essere fruibili attraverso dispositivi sia desktop che mobile ed accessibili attraverso un numero ridotto di interazioni	End User	RNF_U1 RNF_U2 RNF_U3 RNF_U4 RNF_U5
2	DG_4	Costi di sviluppo: Bisogna abbattere i costi di sviluppo del sistema	Costo	Top Management
1	DG_5	Tempi di sviluppo: Bisogna sviluppare il sistema in tempi ridotti	Costo	Top Management
3	DG_6	Throughput: Il sistema deve essere in grado di supportare fino a 100000 visite mensili	Performance	RNF_P2
3	DG_7	Tempi di risposta: Il sistema deve elaborare le richieste e produrre output in meno di 2 secondi (al netto di ritardi dovuti alla trasmissione su rete)	Performance	RNF_P3

2	DG_8	Leggibilità: Il codice prodotto dev'essere semplice da comprendere. Ogni metodo e campo non banale dev'essere documentato opportunamente al fine di aumentarne la comprensione	Supportability	RNF_S1
---	------	---	----------------	--------

Riportiamo ora quelli che sono i compromessi considerati e la posizione del team di in relazione ad ognuno di essi.

1.2.1 Trade-offs

Tempo di rilascio vs Funzionalità

Nonostante i tempi di sviluppo ridotti, il team si impegna nel consegnare il sistema completo di tutte le sue funzionalità tenendo conto di un possibile ritardo nella consegna.

Prestazioni vs Costi

Per rientrare nel budget a disposizione, il team cercherà di ottenere le migliori prestazioni ma nelle ore-lavoro garantite dal budget.

1.3 Definizioni, acronimi ed abbreviazioni

- PT = Personal Trainer
- Piattaforma = Applicazione web
- CRUD = Create, Read, Update, Delete

1.4 Riferimenti

- Requisiti funzionali: Sezione 3.2 del RAD
- Requisiti non funzionali: Sezione 3.3 del RAD

1.5 Panoramica

Nel documento verranno affrontati l'analisi delle architetture di sistemi simili, la scomposizione in sottosistemi del sistema proposto con la definizione della strategia di deploy e le condizioni limite. Verranno quindi definiti i servizi esposti da ciascun sottosistema.

2. Architettura di sistemi simili

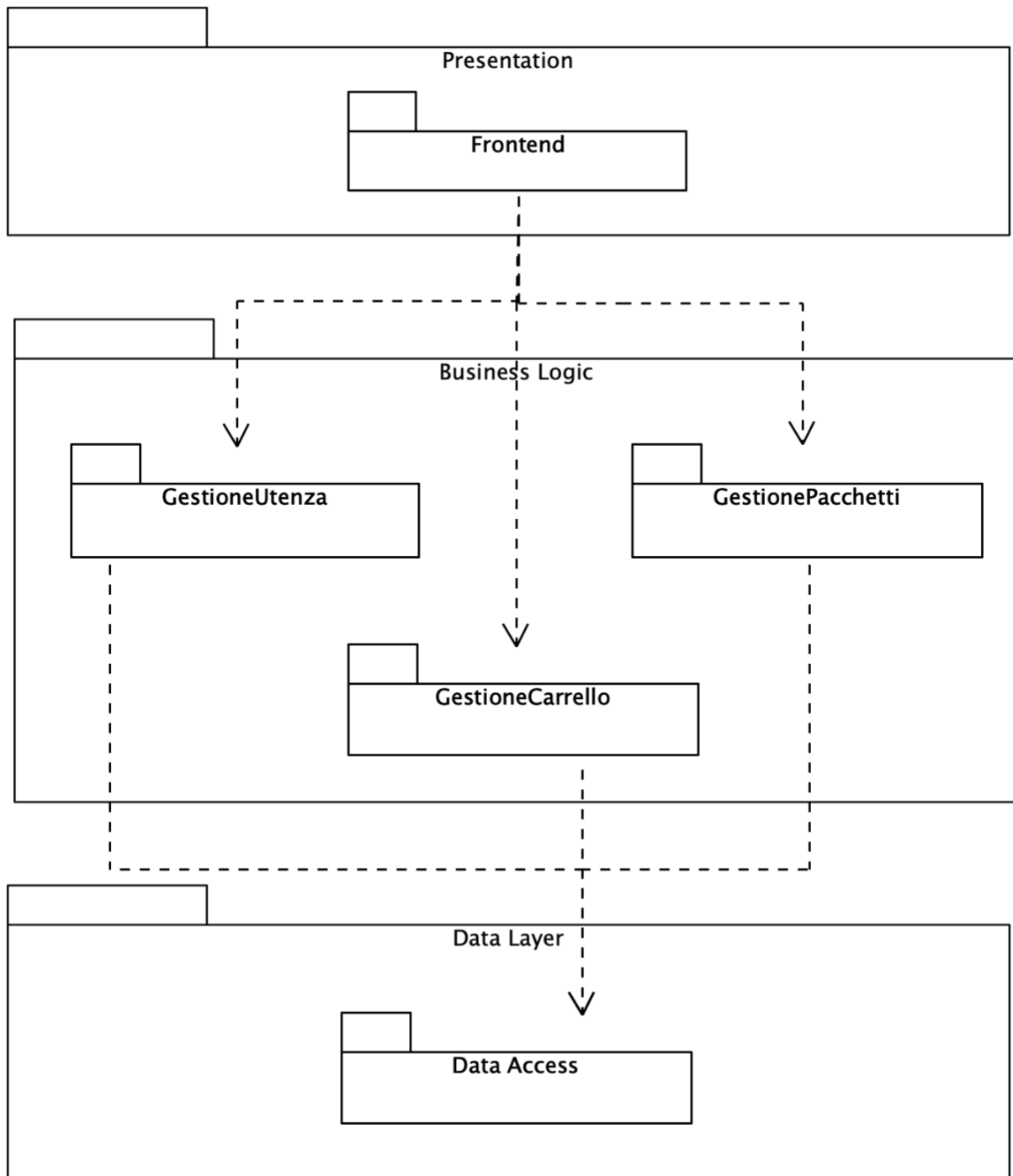
Non ci sono piattaforme simili ad urCoach. Attualmente pt che offrono online coaching utilizzano i social network per farsi conoscere e stabilire relazioni con gli atleti, nel caso in cui l'atleta sia interessato ad intraprendere un percorso con un pt dovrà contattarlo tramite email e così avanzerà il rapporto tra i due. In alcuni casi vengono anche utilizzati sistemi creati per freelancer come Fiverr. Fiverr è probabilmente costruito con un'architettura MVC, la tecnologia di backend usata è Ruby mentre per il frontend utilizza un framework Javascript (React). Non si possono avere informazioni sulla gestione dei dati persistenti perchè hanno deciso di dividere il backend dal frontend: su un server risiede il backend con un probabile DBMS, il backend realizza delle REST API ed il frontend, posizionato su un server differente, sfrutta tali API per poter accedere ai dati con lo scopo di mostrarli, crearne di nuovi, aggiornarli o eliminarli (CRUD operation).

3. Architettura del sistema proposto

3.1 Panoramica

urCoach è un'applicazione distribuita che fornisce un'interfaccia web. Essa, inoltre, ricorre all'utilizzo di un database relazionale per il salvataggio dei dati persistenti.

3.2 Scomposizione in sottosistemi



Powered By Visual Paradigm Community Edition

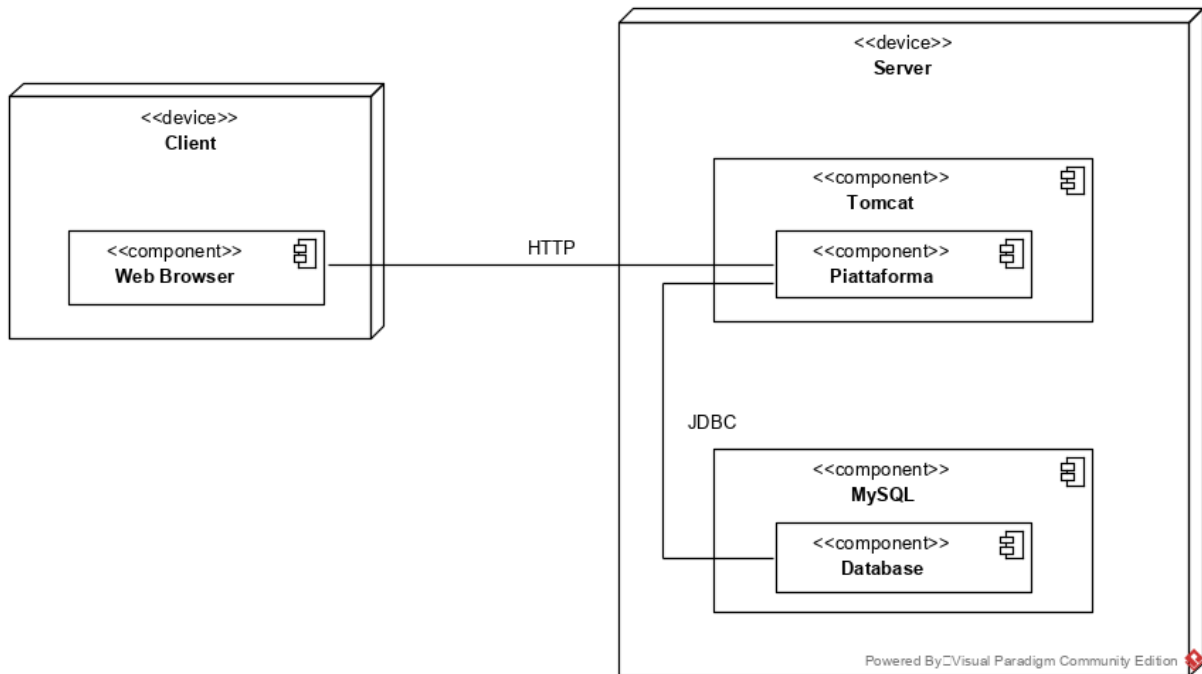
Si è deciso di basare urCoach su un'architettura MVC organizzata in modo da minimizzare l'accoppiamento e favorire un'alta coesione, vediamo il sistema suddiviso su tre livelli:

- Il livello di Presentation che contiene solo il Frontend, ovvero l'interfaccia che gli utenti utilizzeranno per interagire col sistema
- Il livello di Business Logic, ovvero tutta la logica applicativa, troviamo infatti la gestione di utenti, pacchetti e del carrello
- Il livello Data col sottosistema Data Access che ha l'accesso al database per salvare, prelevare o modificare dati.

Realizziamo quindi la View al livello di Presentazione, il Control è localizzato totalmente al livello di Business Logic ed il Model lo troviamo al livello Dati.

3.3 Hardware/software mapping

urCoach consiste di un'applicazione distribuita installabile su un qualsiasi server in grado di eseguire Java e MySQL. L'architettura scelta prevede l'interazione della piattaforma con un database: date le risorse si ritiene opportuno installare sia il web server che il database in un unico nodo. Il sistema sarà accessibile tramite comuni browser web installati sui dispositivi a disposizione degli attori.



3.4 Gestione dei dati persistenti

Per la gestione dei dati persistenti si è deciso di utilizzare MySQL, questo per due motivi principali:

- I dati sono ben strutturati e per questo si prestano particolarmente ad essere memorizzati in questo DBMS
- Per abbattere ulteriormente i costi, strutturare dei file avrebbe richiesto maggior lavoro e quindi più costi; inoltre MySQL è gratuito e di conseguenza comporta un costo in meno.

Infine ovviamente MySQL ci permette di trattare i dati ed effettuare operazioni con estrema semplicità e di aggiungere un ulteriore livello di sicurezza per l'accesso ai dati.

Le seguenti entità, provenienti dal Class Diagram verranno rese persistenti:

- Atleta
- PersonalTrainer
- Pacchetto
- Fattura
- Acquisto
- Categoria
- Amministrativo

Dovranno essere per altro tenuti persistenti i possibili ruoli di un amministrativo.

3.5 Controllo degli accessi e sicurezza

Il controllo degli accessi è garantito tramite l'utilizzo di username e password per gli utenti del sistema che hanno possibilità di creare o modificare gli oggetti che modellano entità di dominio, così da prevenire accessi non autorizzati ad informazioni sensibili. Sottolineiamo che il sistema non fornirà un metodo di recupero della password.. Si ricorrerà all'utilizzo della sessione del server per tenere traccia dell'utente loggato. Per questioni di efficienza, la sessione sarà attiva per soli 30 minuti dopo l'ultima interazione dell'utente col sistema. Il salvataggio delle password nel database sarà cifrato con [CIFRATURA]. Le operazioni che gli utenti dell'applicazione web possono effettuare sugli oggetti sono riportate nella tabella che segue:

	Utenza	Pacchetti	Carrello
Personal Trainer	Login Visualizzazione dati personali Modifica dati personali Cancellazione account Visualizzazione dello storico vendite	Aggiunta pacchetto Modifica pacchetto Eliminazione pacchetto	
Atleta	Login Visualizzazione dati personali Modifica dati personali Cancellazione account Visualizzazione storico ordini effettuati Visualizzazione di tutti i personal trainer Visualizzazione di un personal trainer	Visualizzazione del catalogo dei pacchetti Ricerca di un pacchetto per nome Filtro in una lista di pacchetti per fascia di prezzo Filtro in una lista di pacchetti per categoria Filtro in una lista di pacchetti per personal trainer	Aggiunta pacchetto al carrello Eliminazione pacchetto dal carrello Visualizzazione del carrello Checkout del carrello
Gestore Ordini	Visualizzazione storico ordini globale Login		
Recruiter	Accettazione personal trainer Rifiuto personal trainer Login		
Utente	Registrazione	Visualizzazione del catalogo dei pacchetti Ricerca di un pacchetto per nome	Aggiunta pacchetto al carrello Eliminazione pacchetto dal carrello

		Filtro in una lista di pacchetti per fascia di prezzo Filtro in una lista di pacchetti per categoria Filtro in una lista di pacchetti per personal trainer	Visualizzazione del carrello
--	--	--	------------------------------

3.6 Controllo flusso globale del sistema

Il sistema adotta un controllo del flusso globale di tipo thread-driven, questo perché il web container (Tomcat) web permette l'interazione concorrente tra la WebApp e più client tramite l'intercettazione di eventi generati proprio da questi ultimi.

Questa soluzione permette al sistema di poter rispondere a più utenti contemporaneamente ma richiede che gli accessi in scrittura ai dati persistenti avvengano sequenzialmente, gestendo opportunamente le sezioni critiche. In generale, ogni richiesta da parte di un utente verrà eseguita in un thread dedicato.

3.7 Boundary condition

3.7.1 Startup sistema

Nome Caso Uso	Startup Sistema	
Attori Partecipanti	Amministratore	
	Amministratore	urCoach
Flusso di Eventi	1. L'amministratore scrive in console "./startup.sh"	
		2. Il sistema inizia l'esecuzione creando manager ed inizializzandoli.
Pre-Condizioni	L'amministratore è connesso tramite una shell al server che ospita il web server	
Post-Condizioni	Il sito è raggiungibile da qualsiasi web browser connesso alla rete	

3.7.2 Shutdown Sistema

Nome Caso d'uso	Shutdown Sistema	
Attori Partecipanti	Amministratore	
	Amministratore	urCoach
Flusso di Eventi	1. L'amministratore scrive in console "/shutdown.sh"	
		2. Il sistema esegue lo script di shutdown
Pre-condizioni	L'amministratore è connesso tramite una shell al server che ospita il web server. Il web server è stato precedentemente avviato.	
Post-condizioni	Il sito non è più raggiungibile.	

3.7.3 Fallimento

TirocinioSmart può incorrere in diversi casi di fallimento, riguardanti sia l'hardware che il software:

- Fallimenti Hardware: Crash del disco su cui i dati persistenti sono salvati: il sistema non prevede alcuna strategia di backup e ripristino dei dati
- Fallimenti nell'ambiente di esecuzione: Interruzione della fornitura elettrica al server: il sistema non prevede alcuna strategia che ne garantisca l'operabilità in questo tipo di condizione
- Fallimenti Software: Impossibilità di stabilire una connessione col database: una schermata notifica l'errore all'utente.

4. Subsystem services

Sottosistema	Descrizione
Utenza	Consente ad un Utente non registrato di effettuare la registrazione e a un Utente registrato di effettuare il login. Permette a quest'ultimo di accedere alla propria pagina utente per modificare i dati personali inseriti all'atto di registrazione ed inoltre è possibile eliminare definitivamente il proprio account

Servizio	Descrizione
login	Consente ad un Atleta/Personal Trainer di accedere al sistema tramite una coppia di username e password
logout	Consente ad un Atleta/Personal Trainer di rimuovere l'accesso al sistema
loginAmministrativo	Consente ad un Recruiter/Gestore Ordini di accedere alle rispettive funzionalità protette
registrazioneAtleta	Permette ad un Atleta non registrato di creare un'account nel sistema
registrazionePersonalTrainer	Permette ad un Personal Trainer non registrato di creare un'account nel sistema
visualizzazioneDatiPersonaliAtleta	Consente ad un Atleta registrato di visualizzare le informazioni relative al proprio account
visualizzazioneDatiPersonaliPersonalTrainer	Consente ad un Personal Trainer registrato di visualizzare le informazioni relative al proprio account
modificaDatiAtleta	Consente ad un Atleta registrato di modificare le informazioni relative al proprio account
modificaDatiPersonalTrainer	Consente ad un Personal Trainer registrato di modificare le informazioni relative al proprio account
eliminazioneAtleta	Consente ad un Atleta registrato di cancellare il proprio account dal sistema
eliminazionePersonalTrainer	Consente ad un Personal Trainer registrato di modificare le informazioni relative al proprio account
visualizzazioneStoricoAcquisti	Consente ad un Atleta registrato di visualizzare gli acquisti da lui effettuati

visualizzazioneStoricoVendite	Consente ad un Personal Trainer registrato di visualizzare le vendite da lui effettuate
visualizzazioneStoricoOrdini	Consente ad un Gestore degli ordini di visualizzare tutti gli ordini effettuati sull'eCommerce
visualizzazioneListaPersonalTrainer	Consente ad un Atleta di visualizzare una lista di tutti i Personal Trainer iscritti all'eCommerce
visualizzazionePersonalTrainer	Consente ad un Atleta di visualizzare una pagina che contiene maggiori informazioni su un Personal Trainer
accettazionePersonalTrainer	Consente ad un Recruiter di accettare l'iscrizione di un Personal Trainer
rifiutoPersonalTrainer	Consente ad un Recruiter di rifiutare l'iscrizione di un Personal Trainer

Sottosistema	Descrizione
Carrello	Permette di visualizzare il carrello, aggiungere e rimuovere un servizio da quest'ultimo. Consente di inserire dati di pagamento e ad un Atleta registrato di effettuare il checkout.

Servizio	Descrizione
visualizzaCarrello	Consente di visualizzare i pacchetti aggiunti nel carrello e il prezzo totale
aggiungiProdottoAlCarrello	Consente di aggiungere un pacchetto al carrello
rimuoviProdottoDalCarrello	Consente di rimuovere un pacchetto dal carrello
inserisciDatiPagamento	Permette di inserire i dati della carta di credito, necessari per effettuare il checkout
checkoutCarrello	Permette ad un Atleta registrato di acquistare i pacchetti inseriti nel carrello. Dopo tale operazione il carrello verrà svuotato

Sottosistema	Descrizione
Pacchetti	Permette l'aggiunta, la modifica e la rimozione di un pacchetto da parte di un Personal Trainer. Consente inoltre la visualizzazione e la ricerca all'interno di tutto il catalogo da parte di ogni Atleta

Servizio	Descrizione
aggiuntaPacchetto	Permette ad un Personal Trainer di inserire un nuovo pacchetto al catalogo
eliminazionePacchetto	Permette ad un Personal Trainer di rimuovere un pacchetto dal catalogo
modificaPacchetto	Permette ad un Personal Trainer di modificare le informazioni relative ad un pacchetto precedentemente inserito
visualizzaCatalogo	Permette di visualizzare i pacchetti presenti nel catalogo
ricercaPacchettoNome	Consente la ricerca di un pacchetto nel catalogo tramite una keyword
filtroPacchettiPrezzo	Consente di filtrare all'interno di una lista di pacchetti per il costo
filtroPacchettiCategoria	Consente di filtrare all'interno di una lista di pacchetti per la categoria
filtroPacchettiPersonalTrainer	Consente di filtrare all'interno di una lista di pacchetti per il personal trainer che l'ha creato