



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino



# SDD

## System Design Document

### SmartGym

Riferimento	thISTeam_SDD
Versione	2.0
Data	17/02/2023
Destinatario	Docente di Ingegneria del Software 2022/23 (Carminè Gravino)
Presentato da	thISTeam
Approvato da	



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

## Revision History

Data	Versione	Descrizione	Autori
14/11/2022	0.1	Creazione del documento	thlSTeam
18/11/2022	0.2	Definito l'indice del documento	thlSTeam
01/12/2022	0.3	Aggiunto il logo	thlSTeam
03/12/2022	0.4	Stesura iniziale dei capitoli 1-2	thlSTeam
06/12/2022	0.5	Stesura iniziale del capitolo 3	thlSTeam
07/12/2022	0.6	Continuazione della stesura del capitolo 3 ed aggiustamenti vari al documento	thlSTeam
10/12/2022	0.7	Aggiunto il dizionario dei dati e i casi limite	thlSTeam
17/12/2022	1.0	Revisione	thlSTeam
18/01/2023	1.1	Migliorati i design goal (in linea con i feedback ricevuti)	thlSTeam
17/02/2023	2.0	Revisione	thlSTeam



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

## Team Members

Nome	Ruolo nel progetto	Acronimo	Informazioni di contatto
Daniele Fabiano	Team Member	DF	d.fabiano3@studenti.unisa.it
Mariantonietta Maselli	Team Member	MM	m.maselli3@studenti.unisa.it
Michele Spinelli	Team Member	MS	m.spinelli26@studenti.unisa.it



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

## Indice

<b>1. Introduzione</b>	<b>5</b>
1.1 Scopo del sistema	5
1.2 Obiettivi di Design (Design Goals)	5
1.3 Definizioni, acronimi, e abbreviazioni	7
1.4 Riferimenti	7
1.5 Organizzazione del documento	7
<b>2. Architettura del sistema corrente</b>	<b>8</b>
<b>3. Architettura del sistema proposto</b>	<b>9</b>
3.1 Panoramica sulla sezione	9
3.2 Decomposizione in sottosistemi	9
3.3 Mapping hardware/software	15
3.4 Gestione dei dati persistenti	16
3.5 Controllo degli accessi e sicurezza	23
3.6 Controllo globale del sistema	23
3.7 Condizioni limite	24
<b>4. Servizi dei sottosistemi</b>	<b>27</b>
<b>5. Glossario</b>	<b>31</b>



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

## 1. Introduzione

---

### 1.1 Scopo del sistema

SmartGym propone di semplificare la distribuzione e la fruizione della scheda esercizi degli atleti iscritti alla palestra, al fine di ridurre l'uso del cartaceo creando uno strumento di digitalizzazione dei processi che portano alla realizzazione di una scheda esercizi.

Il sistema, permette agli atleti di creare in completa autonomia la scheda esercizi. Permette inoltre ai trainer di fornire assistenza agli atleti meno esperti che delegheranno al trainer le attività di creazione della scheda esercizi, sancendo quindi un periodo di allenamento assistito.

La componente principale del sistema è la funzione di creazione della scheda esercizi mediante tre diverse modalità fornendo la massima libertà di scelta agli atleti coinvolti nell'utilizzo del sistema realizzato.

Inoltre, al fine di monitorare le attività di esercizio ed allenamento, il sistema offre la possibilità di visualizzare degli storici delle attività effettuate per controllare l'andamento degli esercizi nel corso del tempo.

### 1.2 Obiettivi di Design (Design Goals)

In questa sezione saranno presentati i Design Goals, le qualità del sistema che dovranno essere ottimizzate. I Design Goals scelti sono suddivisi nelle seguenti categorie:

- **End User:** Le qualità desiderabili dal punto di vista dell'utente e non coperte nelle altre categorie
- **Dependability:** Determina le qualità che definiscono quanto lavoro bisogna fare per minimizzare i fallimenti del sistema
- **Performance:** Include le qualità relative ai requisiti di spazio e velocità imposti al sistema

Ogni design goal è descritto da:

- **Rank:** un valore di priorità compreso tra 1 e 6
- **ID Design Goal:** un identificatore univoco ed un nome esplicativo
- **Descrizione:** una descrizione del design goal
- **Categoria:** la categoria di appartenenza del design goal
- **RNF di origine:** il requisito non funzionale che lo ha generato



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

**Design Goals:**

Rank	ID	Descrizione	Categoria	RNF di origine
1	DG_1 Intuitività dell'interfaccia	Il sistema deve esser dotato di una interfaccia grafica semplice ed intuitiva seguendo le Mandel's Golden Rules.	End User	RNF_U_1
2	DG_2 Fallimento del sistema	Il sistema deve notificare l'utente dell'impossibilità di completare un'operazione richiesta a causa di un fallimento del sistema.	Dependability	RNF_A_1
3	DG_3 Always Online	Il sistema deve garantire l'accesso ad internet per rendere disponibili le funzionalità 24 ore su 24.	Dependability	RNF_P_2
4	DG_4 Gestione ruoli utenza	Il sistema deve garantire la netta separazione delle operazioni a seconda del ruolo dell'utente.	Dependability	RNF_A_2
5	DG_5 Tempi di risposta	Il sistema deve garantire che le operazioni abbiano un tempo di risposta entro i 7 secondi.	Performance	RNF_P_1
6	DG_6 Sezione di Help	Il sistema dispone di una sezione di help nella quale saranno contenute le risposte ai dubbi più frequenti riscontrati dagli utenti del sistema.	End User	RNF_U_2

**Trade-Off:**

Trade-Off	Descrizione
<b>Performance vs Usabilità</b>	Affinché il sistema mantenga una interfaccia grafica intuitiva e semplice con un alto tasso di usabilità, sono in parte sacrificabili i tempi di risposta e le performance
<b>Performance vs Affidabilità</b>	Per far sì che il sistema risulti robusto, affidabile e consistente, sono in parte sacrificabili i tempi di risposta e le performance



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

### 1.3 Definizioni, acronimi, e abbreviazioni

- **API:** Application Program Interface
- **FSA:** Fire Store API
- **XML:** eXtensible Markup Language
- **GUI:** Graphical User Interface
- **DAO:** Data Access Object
- **SOW:** Statement Of Work
- **RAD:** Requirements Analysis Document
- **CD:** Class Diagram
- **UCBC:** Use Case Boundary Conditions

### 1.4 Riferimenti

Di seguito una lista di riferimenti ad altri documenti utili durante la lettura.  
Link ai precedenti documenti:

- [SOW](#)
- [RAD](#)

Link per maggiori informazioni sul tipo di database utilizzato:

- [FireBase](#): l'intera suite di servizi
- [Cloud Firestore](#): il servizio di database
- [Firebase Authentication](#): il servizio di autenticazione

Link al documento di tracciabilità dei vincoli:

- [Tracciabilità vincoli](#)

### 1.5 Organizzazione del documento

Il seguente documento di System Design è diviso in quattro sezioni:

**Introduzione:** Una descrizione generale dello scopo del sistema e degli obiettivi di design che il sistema intende raggiungere.

**Architettura software corrente:** Lo stato attuale dell'architettura di un eventuale software già presente.



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

**Architettura software proposta:** La descrizione nel dettaglio di come il sistema sarà decomposto in sottosistemi, il mapping Hardware/Software e la gestione dei dati persistenti. Verranno inoltre affrontate le strutture dei singoli sottosistemi e le boundary conditions riguardanti l'intero sistema.

**Glossario:** Una lista dei termini usati nel documento con relativa spiegazione.

## 2. Architettura del sistema corrente

---

L'insieme delle funzionalità comprese in SmartGym non sono attualmente riconducibili ad un unico sistema di riferimento già esistente sul mercato. I software attualmente disponibili comprendono solo una parte delle funzionalità e pertanto non c'è un'architettura da confrontare.





Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

## 3. Architettura del sistema proposto

---

### 3.1 Panoramica sulla sezione

Il sistema proposto si basa sullo stile architetturale Three Tier. Attraverso l'utilizzo di tale architettura, possiamo facilmente separare la logica di presentazione, da quella di elaborazione. La seguente architettura mette in risalto una serie di qualità:

- Leggibilità
- Manutenzione
- Riutilizzo

Relativamente allo sviluppo ed implementazione del sistema, saranno usate tecnologie standard per applicativi Android come XML per la descrizione dell'interfaccia grafica e il linguaggio Java per la logica di elaborazione.

Inoltre si farà uso delle Firestore API per collegamento e comunicazione con il database, salvando e prelevando da esso i dati necessari.

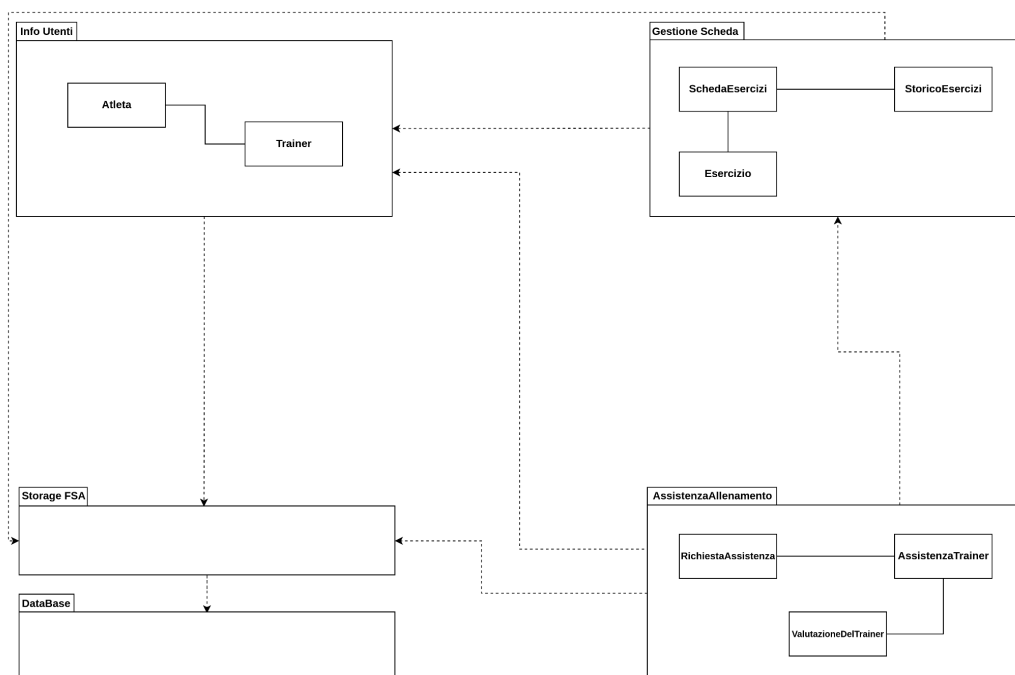
### 3.2 Decomposizione in sottosistemi

I sottosistemi individuati sono:

- **Info Utenti:** si occupa di tutte le operazioni di registrazione, autenticazione e inserimento dati relativi alle varie utenze presenti nel sistema.
- **Gestione Scheda:** si occupa di tutte le operazioni di creazione, modifica, eliminazione e monitoraggio di una scheda esercizi e degli esercizi nel tempo.
- **Assistenza Allenamento:** si occupa di tutte le operazioni che sanciscono l'inizio, il prosieguo e la fine di un periodo di assistenza allenamento stabilito tra un trainer ed un atleta.
- **Storage FSA:** si interpone tra i vari sottosistemi e il sottosistema del Database.
- **Database:** si occupa di gestire le operazioni di persistenza dati con il database di riferimento.

I sottosistemi sono rappresentati attraverso il seguente diagramma di decomposizione dei sottosistemi, indicando le dipendenze tra essi:

### SBSD: SuBSystemDiagram



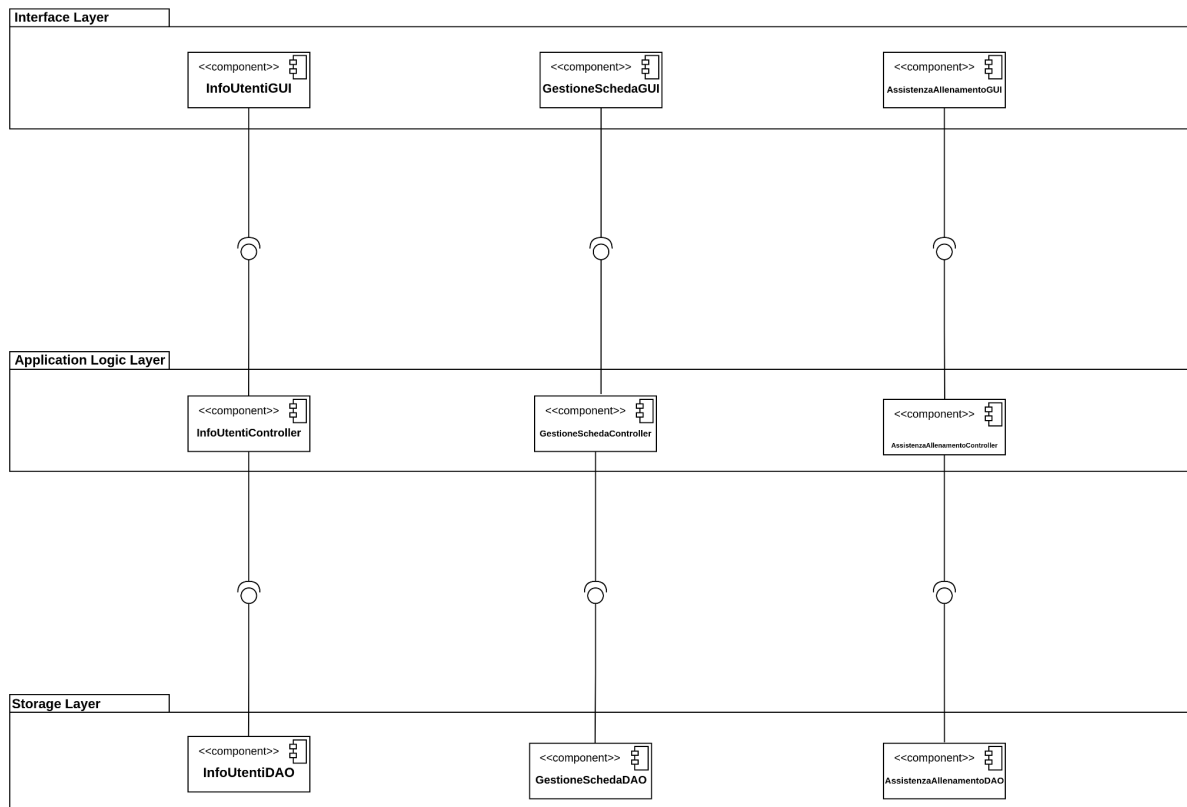
Per ogni sottosistema è illustrata una vista dettagliata che descrive le principali componenti:

- **GUI**: Graphical User Interface, raggruppa la logica di presentazione delle interfacce grafiche delle relative activity presenti nell'applicazione
- **Controller**: si occupa della logica applicativa
- **DAO**: Data Access Object, fornisce l'accesso ai dati persistenti



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

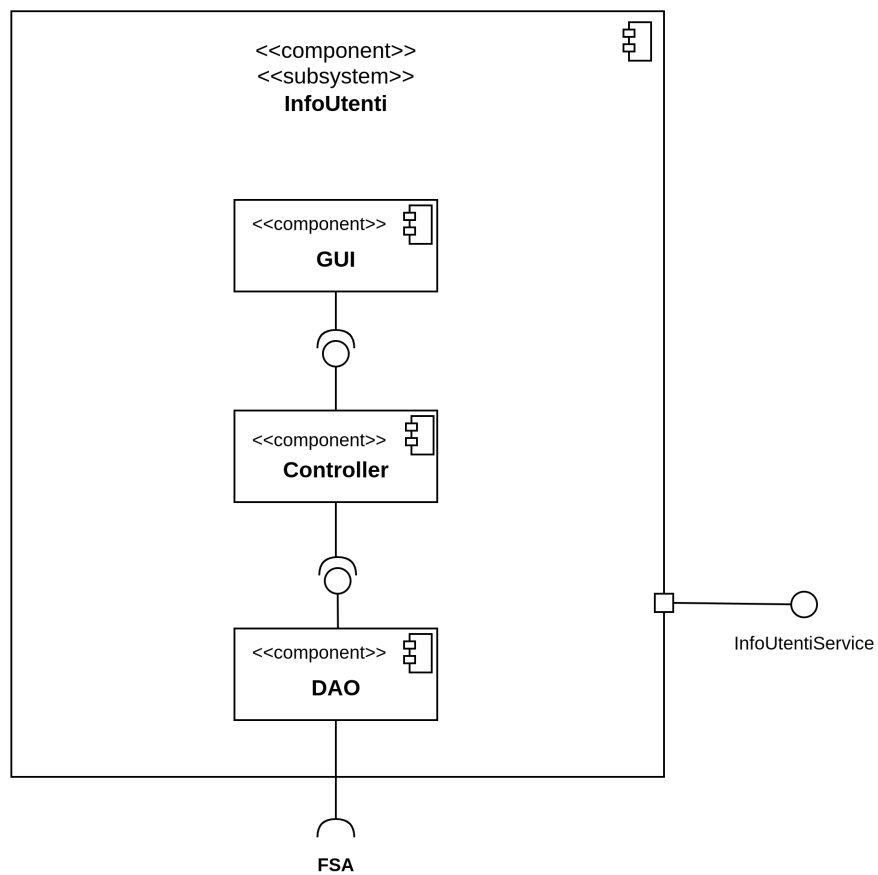
## Diagramma Architeturale





Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

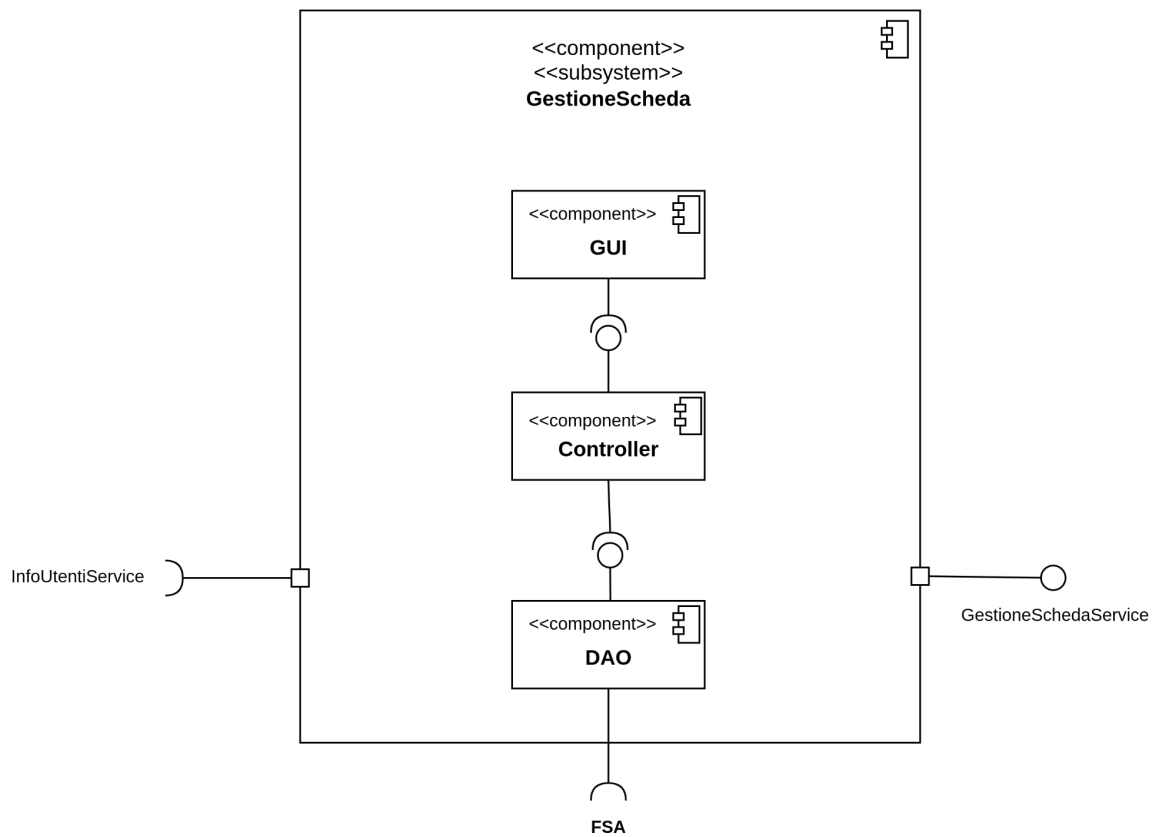
**Sottosistema Info Utenti**





Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

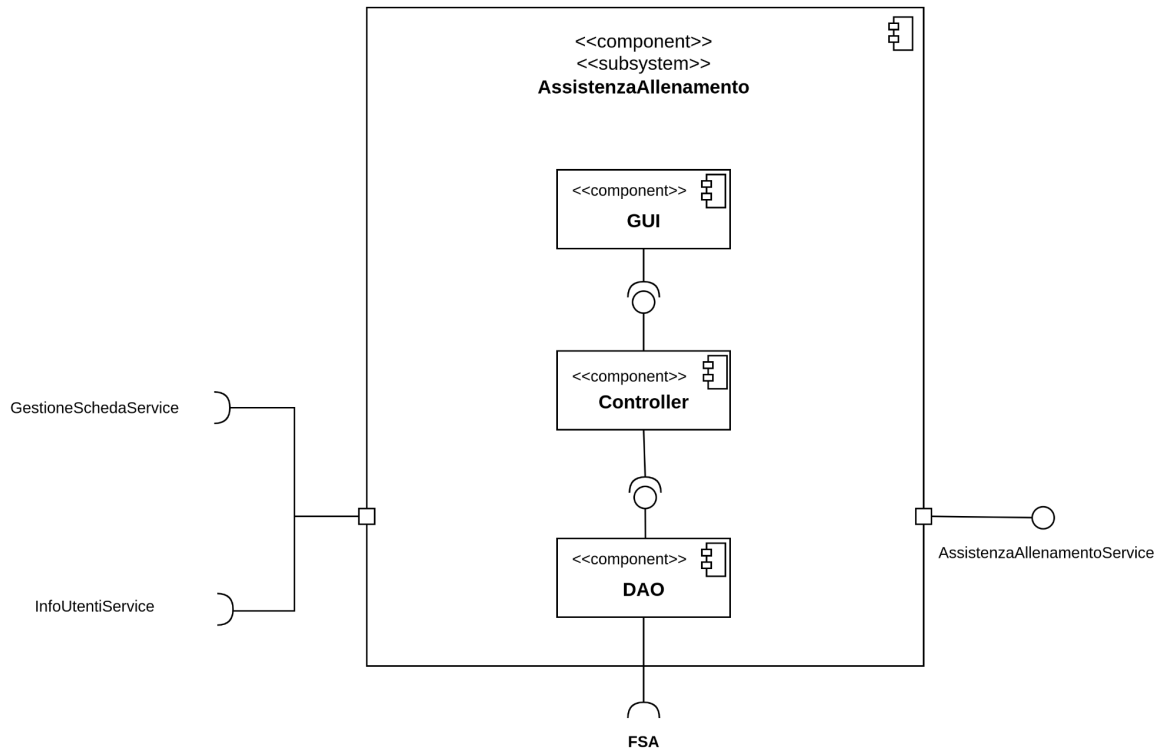
**Sottosistema Gestione Scheda**





Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

**Sottosistema Assistenza Allenamento**





Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

### 3.3 Mapping hardware/software

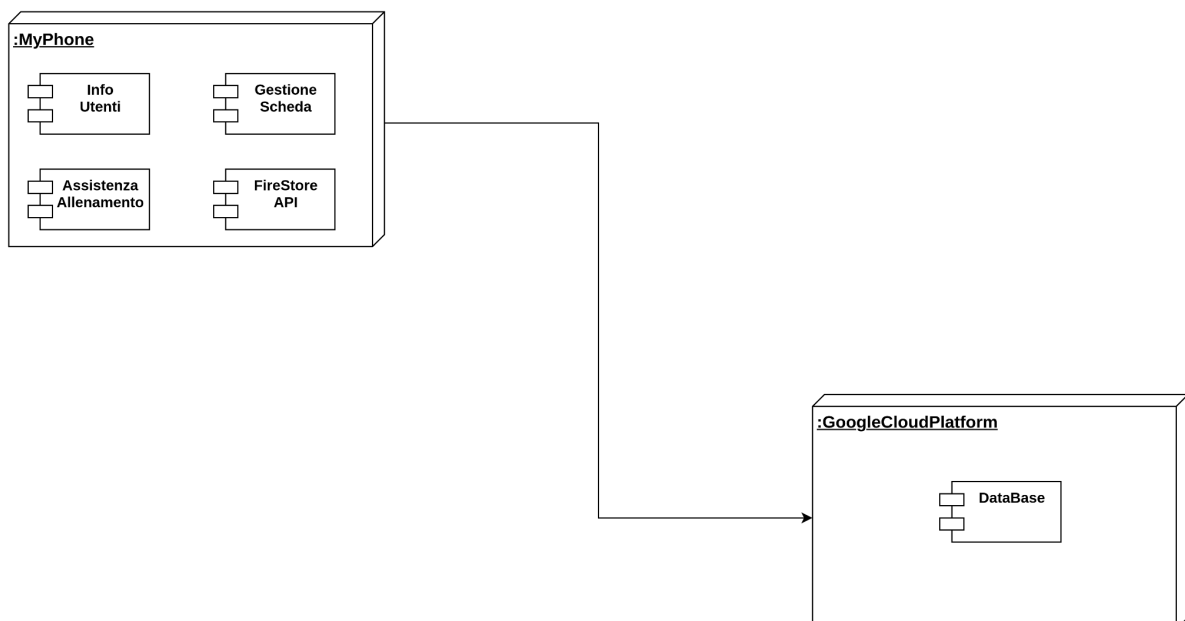
L'applicazione Android che verrà sviluppata prevede tutta la logica di presentazione ed elaborazione sul dispositivo mobile stesso.

Il database Firestore risiede invece su un server messo a disposizione dalla Google Cloud Platform dove è presente un'istanza della suite di servizi FireBase. Le interazioni con esso avverranno attraverso le API messe a disposizione dal servizio.

Il nostro sistema non è dunque centralizzato su un unico nodo, ma prevede l'utilizzo di due dispositivi diversi.

Viene di seguito mostrato il Deployment Diagram di riferimento per il mapping hardware/software.

#### DD: Deployment Diagram





### 3.4 Gestione dei dati persistenti

Per il salvataggio dei dati persistenti è stato previsto l'utilizzo del database Firestore. Questo particolare database rientra nella tipologia dei NoSQL document-oriented. Le ragioni dietro questa scelta ricadono nei seguenti motivi:

- **Limitazioni di SQLite:** ad oggi Android ha supporto nativo a database SQL attraverso SQLite ed è utilizzabile esclusivamente in app, definendo delle classi per la creazione del database che saranno istanziate ed eseguite al primo avvio del sistema.
- **Database centralizzato:** il seguente database si trova su un server della Google Cloud Platform che fornisce l'intera suite di servizi Firebase. Questo ci permette di avere un database facilmente accessibile da qualunque dispositivo.
- **Integrazione nel linguaggio:** Android ha completo supporto alle API di Firestore, questo permette di effettuare operazioni strutturate e complesse attraverso il set di classi e metodi a disposizione della specifica interfaccia di applicazione.
- **Servizio di autenticazione:** Firestore include un servizio di autenticazione che permette di gestire facilmente i login, le registrazioni e le sessioni degli utenti del sistema.
- **Flessibilità dello schema:** Firestore non impone un particolare schema di organizzazione dei dati. Sono presenti le possibilità di organizzare i dati sotto forma di documenti e di raccogliarli in raccolte e sotto raccolte. Inoltre è possibile definire dei riferimenti a documenti presenti in altre raccolte.
- **Tipi di dati "rilassati":** I tipi di dati sono "rilassati" e sono definiti in maniera più semplice rispetto ad altri servizi di database:
  - **number:** qualunque dato di tipo numerico
  - **string:** qualunque dato di tipo alfanumerico
  - **boolean:** qualunque dato di tipo booleano
  - **timestamp:** qualunque dato che indichi data e/o ora
  - **reference:** tipo di dato che identifica un riferimento ad un altro documento del database
  - **sottoraccolta:** tipo di dato strutturato che identifica una collezione di documenti come un campo del documento
  - **array/mappe:** tipi di dati strutturati per una collezione di dati semplici

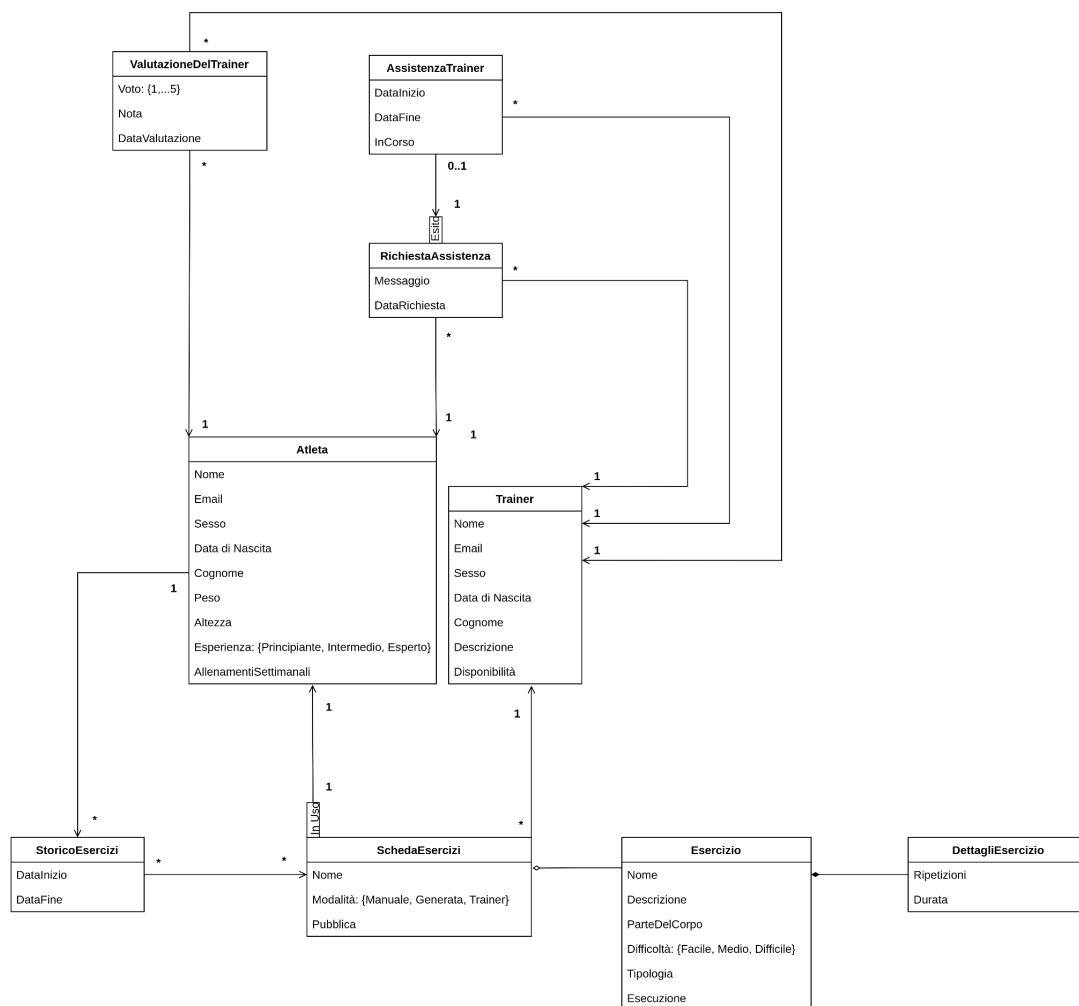




Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

Di seguito è riportato il class diagram ristrutturato. Si è deciso di partizionare la classe esercizio affinché si potessero dividere i dati statici da quelli variabili, evitando lo spreco di memoria e ridondanza di informazioni all'interno del database. La classe utente è stata invece accorpata sia nella classe atleta che nella classe trainer, questo ci permette di effettuare delle interrogazioni in maniera più efficiente senza dover accedere a due documenti separati. Usando un database document-oriented che riprende concetti di un database object-oriented, il mapping dal class diagram è pressoché 1:1. Per questa ragione non è presente nessun altro schema che rappresenti il database ma verrà riportato solamente il class diagram.

### CD\_SDD: Entity Class Diagram ristrutturato





Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

**Dizionario dei dati:**

Nome Raccolta: atleti		
Descrizione	contiene tutti i documenti relativi a specifici dati degli utenti atleti	
Struttura dei documenti		
Nome IDentificativo del documento: ID generato		
Nome campo	Tipo	Vincoli
email	string	campo non nullo
nome	string	campo non nullo
cognome	string	campo non nullo
sex	string	valori ammissibili: {M,F}
dataDiNascita	timestamp	si memorizza solo la data (ora settata a 00:00:00)
allenamentiSettimanali	number	min = 1 ; max = 7
altezza	number	campo non nullo
peso	number	campo non nullo
esperienza	string	valori ammissibili: {"Principiante", "Intermedio", "Esperto"}



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

Nome Raccolta: trainer		
Descrizione	contiene tutti i documenti relativi a specifici dati degli utenti trainer	
Struttura dei documenti		
Nome IDentificativo del documento: ID generato		
Nome campo	Tipo	Vincoli
email	string	campo non nullo
nome	string	campo non nullo
cognome	string	campo non nullo
sex	string	valori ammissibili: {M,F}
dataDiNascita	timestamp	si memorizza solo la data (ora settata a 00:00:00)
descrizione	string	
disponibilita	boolean	campo non nullo

Nome Raccolta: esercizi		
Descrizione	contiene tutti i documenti relativi ai dati statici degli esercizi	
Struttura dei documenti		
Nome IDentificativo del documento: campo nome		
Nome campo	Tipo	Vincoli
nome	string	ID
descrizione	string	campo non nullo
parteDelCorpo	string	campo non nullo
tipologia	string	valori ammissibili: {"Aerobico", "Anaerobico"}
difficolta	string	valori ammissibili: {"Facile", "Medio", "Difficile"}
esecuzione	string	valori ammissibili: {"Serie", "Tempo"}



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

Nome Raccolta: dettagli_esercizi		
Descrizione	contiene tutti i documenti relativi ai dati variabili degli esercizi	
Struttura dei documenti		
Nome IDentificativo del documento: ID generato		
Nome campo	Tipo	Vincoli
durata	number	il campo inutilizzato viene settato a -1
ripetizioni	number	il campo inutilizzato viene settato a -1
esercizio	reference	punta al corrispondente documento dell'esercizio di riferimento

Nome Raccolta: schede_esercizi		
Descrizione	contiene tutti i documenti relativi ai dati delle schede esercizi	
Struttura dei documenti		
Nome IDentificativo del documento: ID generato		
Nome campo	Tipo	Vincoli
nome	string	campo non nullo
modalita	string	valori ammissibili: {"Manuale", "Generata", "Trainer"}
pubblica	boolean	campo non nullo
inUso	boolean	campo non nullo
autore	reference	punta al corrispondente documento dell'utente che ha creato la scheda
ricevente	reference	punta al corrispondente documento dell'utente che utilizza la scheda
eserciziScelti	array	insieme di riferimenti che puntano ai corrispondenti documenti di dettagli degli esercizi scelti nella scheda



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

Nome Raccolta: richieste_assistenza		
Descrizione	contiene tutti i documenti relativi ai dati delle richieste assistenza	
Struttura dei documenti		
Nome IDentificativo del documento: ID generato		
Nome campo	Tipo	Vincoli
messaggio	string	campo non nullo
dataOraRichiesta	timestamp	campo non nullo
esito	boolean	campo non nullo
mittente	reference	punta al corrispondente documento dell'utente che ha effettuato la richiesta
destinatario	reference	punta al corrispondente documento dell'utente che riceve la richiesta

Nome Raccolta: assistenze_trainer		
Descrizione	contiene tutti i documenti relativi ai dati delle assistenze effettuate dai trainer ai relativi atleti	
Struttura dei documenti		
Nome IDentificativo del documento: ID generato		
Nome campo	Tipo	Vincoli
dataOralnizio	timestamp	campo non nullo
dataOraFine	timestamp	campo non nullo
inCorso	boolean	campo non nullo
richiesta	reference	punta al corrispondente documento della richiesta assistenza
addetto	reference	punta al corrispondente documento del trainer addetto



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

Nome Raccolta: valutazione_trainer		
Descrizione	contiene tutti i documenti relativi ai dati delle valutazioni degli atleti sul periodo di assistenza del trainer	
Struttura dei documenti		
Nome IDentificativo del documento: ID generato		
Nome campo	Tipo	Vincoli
voto	number	campo non nullo
nota	string	
dataOraValutazione	timestamp	campo non nullo
autore	reference	punta al corrispondente documento dell'utente che ha scritto la valutazione
ricevente	reference	punta al corrispondente documento dell'utente che riceve la valutazione

Nome Raccolta: storico_esercizi		
Descrizione	contiene tutti i documenti relativi ai dati dello storico degli esercizi effettuati nel corso del tempo	
Struttura dei documenti		
Nome IDentificativo del documento: ID generato		
Nome campo	Tipo	Vincoli
dataOralnizio	timestamp	campo non nullo
dataOraFine	timestamp	campo non nullo
atleta	reference	punta al corrispondente documento che identifica l'atleta
eserciziSalvati	array	insieme di riferimenti che puntano alle corrispondenti schede esercizi



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

### 3.5 Controllo degli accessi e sicurezza

Viene di seguito riportata la matrice di accesso relative alle funzionalità disponibili per ogni tipo di utenza del sistema

Oggetti \ Attori	Trainer	Atleta
<b>Info Utenti</b>	Registrazione Login Logout AccessoAreaPersonale ModificaDatiAnagraficiEPersonali InserimentoInformazioni ModificaInformazioni SelezioneStatoDisponibilità CancellazioneProfilo	Registrazione Login Logout AccessoAreaPersonale ModificaDatiAnagraficiEPersonali InserimentoCaratteristiche ModificaCaratteristiche CancellazioneProfilo
<b>Gestione Scheda</b>	CreazioneSchedaEsercizi ModificaSchedaEsercizi CancellazioneSchedaEsercizi VisualizzazioneSchedeEsercizi AssegnazioneSchedeEsercizi CondivisioneSchedeEsercizi	CreazioneSchedaEsercizi GenerazioneSchedaEsercizi ModificaSchedaEsercizi CancellazioneSchedaEsercizi VisualizzazioneSchedeEsercizi SelezioneSchedaInUso VisualizzazioneStoricoEsercizi
<b>Assistenza Allenamento</b>	VisualizzazioneRichiesteAssistenza AccettazioneAtleta RifiutoAtleta VisualizzazioneAtletiAssistiti VisualizzazioneValutazioni	RicercaDiUnTrainer RichiestaAssistenzaTrainer RichiestaModificaSchedaEsercizi TerminazioneAssistenzaTrainer ValutazioneDiUnTrainer

### 3.6 Controllo globale del sistema

Il sistema in questione utilizza un flusso di controllo di tipo event-driven. Essendo un'applicazione mobile, ogni evento sarà scatenato dall'interazione dell'utente con l'interfaccia grafica. I relativi listener si occuperanno di gestire e controllare l'evento per poi recuperare i risultati delle relative operazioni della logica applicativa.



### 3.7 Condizioni limite

In questa sezione sono presentati gli use case boundary relativi al lancio dell'applicazione, configurazione e connessione al servizio di database ed errori relativi al servizio di database stesso.

<b>Identificativo</b>		<b>UCBC_1</b> Lancio dell'applicazione
<b>Descrizione</b>		Lo UC permette il lancio dell'applicazione
<b>Attore principale</b>		Utente
<b>Attori secondari</b>		NA
<b>Entry condition</b>		L'utente accede al dispositivo  AND  Il dispositivo è stato precedentemente acceso
<b>Exit condition</b> (on success)		L'applicazione viene lanciata
<b>Exit condition</b> (on failure)		L'applicazione non viene lanciata
<b>FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO</b>		
<b>1</b>	<b>Utente</b>	Seleziona l'applicazione dal listato di quelle attualmente installate sul dispositivo
<b>2</b>	<b>Sistema</b>	Verifica che l'utente abbia precedentemente effettuato il login e rende le funzionalità disponibili all'utente
<b>Scenario/Flusso di eventi Alternativo: L'utente non si era precedentemente loggato</b>		
<b>2.a1</b>	<b>Utente</b>	Mostra una schermata dove poter effettuare login
<b>2.a2</b>	<b>Sistema</b>	Esegue il passaggio 2





Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

Identificativo	UCBC_2 Configurazione e Connessione al servizio di Database	
Descrizione	Lo UC permette la connessione al database	
Attore principale	Amministratore	
Attori secondari	NA	
Entry condition	Il database è stato creato  AND  L'amministratore ha accesso alla console del database  AND  L'applicazione include i parametri e moduli necessari alla connessione al db	
Exit condition (on success)	Il database è connesso all'applicazione	
Exit condition (on failure)	La connessione tra database e applicazione non va a buon fine	
FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO		
1	Amministratore	Seleziona l'istanza del database sulla console del servizio di database FireStore
2	Sistema	Collega il servizio di database all'applicazione
3	Amministratore	Verifica che il servizio sia stato collegato e preleva i parametri di connessione del database
4	Sistema	Istanzia la connessione al database



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

<b>Identificativo</b>		<b>UCBC_3</b> Impossibilità di recuperare i dati dal sistema di database
<b>Descrizione</b>		Lo UC gestisce l'impossibilità di prelevare i dati dal servizio di database
<b>Attore principale</b>		Amministratore
<b>Attori secondari</b>		NA
<b>Entry condition</b>		Il database è stato collegato  AND  L'amministratore ha accesso alla console del database
<b>Exit condition</b> (on success)		I dati dal sistema di database sono nuovamente recuperabili
<b>Exit condition</b> (on failure)		I dati dal sistema di database continuano a non essere recuperabili
<b>FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO</b>		
<b>1</b>	<b>Amministratore</b>	Utilizza i dati precedentemente memorizzati nella cache
<b>2</b>	<b>Sistema</b>	Disconnette temporaneamente il servizio di database dell'applicazione
<b>3</b>	<b>Amministratore</b>	Include <b>UCBC_2</b>
<b>4</b>	<b>Sistema</b>	Preleva i dati dal database e aggiorna la cache
<b>Scenario/Flusso di eventi Alternativo: I dati continuano a non essere prelevabili</b>		
<b>4.a1</b>	<b>Utente</b>	Forza il logout dell'utente dal sistema
<b>4.a2</b>	<b>Sistema</b>	Invita l'utente a provare ad autenticarsi più tardi, non appena saranno risolti i problemi relativi al database



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

## 4. Servizi dei sottosistemi

In questa sezione sono elencati e descritti i servizi di ogni sottosistema.

### Sottosistema Info Utenti

Servizio	Descrizione	Interfaccia
<b>Registrazione Atleta</b>	Questa funzionalità permette di registrarsi come atleta.	<b>InfoUtentiService</b>
<b>Registrazione Trainer</b>	Questa funzionalità permette di registrarsi come trainer.	<b>InfoUtentiService</b>
<b>Login</b>	Questa funzionalità permette di effettuare l'accesso al sistema tramite le proprie credenziali per sfruttare tutte le funzionalità offerte.	<b>InfoUtentiService</b>
<b>Logout</b>	Questa funzionalità permette di disconnettersi dal sistema.	<b>InfoUtentiService</b>
<b>Modifica dati anagrafici/personali</b>	Questo servizio permette di effettuare la modifica dei dati anagrafici e personali.	<b>InfoUtentiService</b>
<b>Inserimento Caratteristiche Atleta</b>	Questo servizio permette l'inserimento delle caratteristiche fisiche e di esperienza dell'atleta.	<b>InfoUtentiService</b>
<b>Modifica Caratteristiche Atleta</b>	Questo servizio permette la modifica delle caratteristiche fisiche e di esperienza dell'atleta.	<b>InfoUtentiService</b>
<b>Inserimento Informazioni Trainer</b>	Questo servizio permette l'inserimento delle	<b>InfoUtentiService</b>



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

	informazioni personali del trainer.	
<b>Modifica Informazioni Trainer</b>	Questo servizio permette la modifica delle informazioni personali del trainer.	<b>InfoUtentiService</b>
<b>Cancellazione profilo</b>	Questa funzionalità permette la cancellazione del profilo.	<b>InfoUtentiService</b>

Sottosistema Gestione Scheda

Servizio	Descrizione	Interfaccia
<b>Creazione Scheda</b>	Il servizio permette di creare una scheda esercizi attraverso una lista di esercizi già noti	<b>GestioneSchedaService</b>
<b>Generazione Scheda</b>	Il servizio permette di generare una scheda esercizi basandosi sull'inserimento delle caratteristiche dell'atleta	<b>GestioneSchedaService</b>
<b>Modifica Scheda</b>	Il servizio permette la modifica di una scheda esercizi precedentemente creata	<b>GestioneSchedaService</b>
<b>Cancellazione Scheda</b>	Il servizio permette la cancellazione di una scheda esercizi già esistente	<b>GestioneSchedaService</b>
<b>Visualizzazione Schede</b>	Il servizio permette la visualizzazione di schede esercizi precedentemente create	<b>GestioneSchedaService</b>
<b>Inserimento Esercizi</b>	Questa funzionalità permette l'inserimento di nuovi esercizi	<b>GestioneSchedaService</b>
<b>Modifica Esercizi</b>	Questa funzionalità	<b>GestioneSchedaService</b>



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

	permette la modifica degli esercizi precedentemente inseriti	
<b>Cancellazione Esercizi</b>	Questa funzionalità permette la cancellazione di esercizi già esistenti	<b>GestioneSchedaService</b>
<b>Selezione Scheda In Uso</b>	Il servizio permette di fissare una scheda esercizi nella sezione di benvenuto del sistema	<b>GestioneSchedaService</b>
<b>Visualizzazione Storico</b>	Il servizio permette di monitorare le attività di esercizi nel corso del tempo	<b>GestioneSchedaService</b>
<b>Condivisione Schede</b>	Il servizio permette di condividere pubblicamente delle schede esercizi precedentemente create	<b>GestioneSchedaService</b>

**Sottosistema Assistenza Allenamento**

<b>Servizio</b>	<b>Descrizione</b>	<b>Interfaccia</b>
<b>Ricerca Trainer</b>	Il servizio permette di ricercare uno specifico trainer tra la lista di quelli presenti	<b>AssistenzaAllenamentoService</b>
<b>Richiesta Assistenza</b>	Il servizio permette di inviare una richiesta assistenza per stabilire un periodo di assistenza allenamento tra un atleta ed un trainer	<b>AssistenzaAllenamentoService</b>
<b>Visualizzazione Richieste Assistenza</b>	Questa funzionalità permette al trainer di visualizzare le richieste assistenza degli atleti	<b>AssistenzaAllenamentoService</b>
<b>Assegnazione Schede</b>	Questa funzionalità permette ad un trainer	<b>AssistenzaAllenamentoService</b>



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

	di assegnare una scheda esercizi ad un atleta assistito	
<b>Richiesta Modifica Scheda</b>	Questa funzionalità permette all'atleta di richiedere una modifica ad una scheda esercizi precedentemente assegnata dal trainer	<b>AssistenzaAllenamentoService</b>
<b>Selezione Stato Disponibilità</b>	Il servizio permette al trainer di selezionare uno stato disponibilità per accettare o meno nuovi atleti	<b>AssistenzaAllenamentoService</b>
<b>Accettazione Atleta</b>	Questa funzionalità permette al trainer di accettare la richiesta assistenza di un nuovo trainer	<b>AssistenzaAllenamentoService</b>
<b>Rifiuto Atleta</b>	Questa funzionalità permette al trainer di rifiutare la richiesta assistenza di un trainer	<b>AssistenzaAllenamentoService</b>
<b>Visualizzazione Atleti Assistiti</b>	Questa funzionalità permette di visualizzare la lista degli atleti assistiti	<b>AssistenzaAllenamentoService</b>
<b>Terminazione Assistenza</b>	Il servizio permette di terminare il periodo di assistenza	<b>AssistenzaAllenamentoService</b>
<b>Valutazione Trainer</b>	Il servizio permette all'atleta di lasciare una valutazione sul trainer alla fine del periodo di assistenza	<b>AssistenzaAllenamentoService</b>
<b>Visualizzazione Valutazioni</b>	Il servizio permette di visualizzare le valutazioni ricevute dal trainer	<b>AssistenzaAllenamentoService</b>



Laurea Triennale in Informatica - Università di Salerno  
Corso di Ingegneria del Software - Prof. C.Gravino

## 5. Glossario

---

Nella presente sezione sono raccolte le sigle o i termini del documento che necessitano di una definizione.

Sigla/Termine	Definizione
<b>Firebase</b>	Suite online di Google Cloud Platform. Eroga servizi di backend per qualsiasi tipo di applicazione software.
<b>Firestore</b>	Servizio di database della suite Firebase.
<b>Database document-oriented</b>	Particolare sistema di persistenza dati basata sull'utilizzo di documenti che memorizzano coppie chiavi-valore. Fanno parte della categoria di database NoSql e non necessariamente relazionali.