

**TrackFit**  
**ProblemStatement\_TrackFit**  
**Versione 1.1**

**LOGO PROGETTO**



Data: 14/10/2025

Progetto: TrackFit	Versione: 1.1
Documento: ProblemStatement_TrackFit	Data: 14/10/2025

**Coordinatore del progetto:**

Nome	Matricola
Gabriele di Palma	0512119257

**Partecipanti:**

Nome	Matricola
Yuri Fierro	0512119524

<b>Scritto da:</b>	Gabriele di Palma
--------------------	-------------------

**Revision History**

Data	Versione	Descrizione	Autore
10/10/2025	1.0	Aggiunta dell'Introduzione	GDP, YF
14/10/2025	1.1	Aggiunta del Problem Statement e Casi d'uso	GDP,YF

# Indice

1. INTRODUZIONE.....	4
1.1. Contesto e motivazioni.....	4
1.2. Obiettivi del progetto .....	4
1.3. Ambito di applicazione .....	5
2. PROBLEM STATEMENT .....	5
2.1. Problem Domain .....	5
2.2. Scenarios .....	6
2.2.1 Scenario 1: Corsa serale .....	6
2.2.2 Scenario 2: Gestione orari da parte dell'allenatore .....	8
2.2.3 Scenario 3: Autenticazione (Portale Business) .....	8
2.2.4 Scenario 4:Prenotazione allenatore .....	9
2.2.5 Scenario 5:Attivazione abbonamento annuale con coupon e pagamento online	10
2.3. Functional Requirements .....	10
2.3.1 Gestione Utenti, Allenatori e Gestori della Palestra .....	10
2.3.2 Gestione Palestre.....	10
2.3.3 Monitoraggio delle Performance .....	11
2.3.4 Reportistica e Analisi Aggregate .....	11
2.3.5 Accessibilità e Interfaccia Intuitiva.....	11
2.4. Non Functional Requirements .....	11
2.4.1 Performance .....	11
2.4.2 Scalabilità .....	12
2.4.3 Sicurezza.....	12
2.4.4 Affidabilità.....	12
2.4.5 Usabilità.....	12
2.4.6 Manutenibilità .....	12
2.5. Target Enviroment .....	13
3. CASI D'USO .....	13
3.1 Casi d'uso scenario 1 .....	13
3.2 Casi d'uso scenario 2 .....	22
3.3 Casi d'uso scenario 3 .....	26
3.4 Casi d'uso scenario 4 .....	36
3.5 Casi d'uso scenario 5 .....	45
4. DELIVERABLES E DEADLINES .....	53

# 1. INTRODUZIONE

## *1.1. Contesto e motivazioni*

Negli ultimi anni l'attenzione verso l'attività fisica e il benessere personale è cresciuta in modo significativo. Sempre più persone, sia a livello amatoriale sia semi-professionale, sentono l'esigenza di monitorare i propri allenamenti al fine di migliorare le prestazioni, mantenere costanza e prevenire infortuni.

Tuttavia, molti strumenti disponibili sul mercato risultano complessi, frammentati o limitati a specifici tipi di attività (ad esempio solo corsa o ciclismo). Questo comporta per l'utente la difficoltà di avere una visione unificata e completa dei propri progressi.

La diffusione di dispositivi mobili e l'evoluzione delle tecnologie web hanno aperto nuove possibilità per lo sviluppo di soluzioni digitali capaci di raccogliere dati in maniera semplice e accessibile, garantendo al contempo un'esperienza d'uso intuitiva.

In questo contesto nasce **TrackFit**, un sistema pensato per offrire un'interfaccia chiara e strumenti di analisi personalizzati, in grado di fornire feedback immediati e report periodici. La motivazione principale alla base del progetto è quella di realizzare una piattaforma che non solo consenta di archiviare e consultare gli allenamenti, ma che stimoli l'utente a mantenere costanza e consapevolezza nel proprio percorso sportivo.

## *1.2. Obiettivi del progetto*

L'obiettivo principale del progetto **TrackFit** è la realizzazione di un sistema software che supporti in maniera semplice ed efficace il monitoraggio e l'analisi degli allenamenti sportivi. Il sistema intende rispondere all'esigenza di centralizzare le informazioni relative all'attività fisica e di presentarle in forma chiara, accessibile e personalizzata.

Gli obiettivi specifici possono essere così articolati:

- **Registrazione delle sessioni di allenamento:** consentire all'utente di inserire i dati relativi ad attività diverse (tipo di sport, durata, distanza, calorie, ecc.), con possibilità di modifica e aggiornamento.
- **Consultazione dello storico:** offrire una sezione dedicata alla visualizzazione cronologica delle attività svolte, con funzioni di ricerca e filtro.
- **Analisi e monitoraggio dei progressi:** generare dashboard interattive con grafici e statistiche che evidenzino l'andamento delle prestazioni nel tempo.
- **Esportazione e reportistica:** permettere l'esportazione dei dati in formati standard (PDF, CSV), per garantire all'utente una gestione autonoma delle informazioni.
- **Interfaccia semplice e intuitiva:** progettare un sistema accessibile sia via web che tramite applicazione mobile, pensato per un'utenza non necessariamente esperta di strumenti tecnologici.

Un ulteriore obiettivo a medio-lungo termine riguarda l'**integrazione con dispositivi indossabili e servizi esterni**, in modo da automatizzare la raccolta dei dati e offrire un'esperienza ancora più completa e fluida.

### ***1.3 Ambito di applicazione***

Il sistema **TrackFit** si colloca nel settore delle applicazioni per il fitness e il benessere personale, con un focus specifico sul monitoraggio e l'analisi delle attività sportive. L'ambito di applicazione riguarda principalmente utenti che praticano sport a livello amatoriale o semi-professionale, i quali desiderano tenere traccia dei propri allenamenti e migliorare le prestazioni nel tempo.

TrackFit è pensato per supportare diverse tipologie di attività fisica, tra cui:

- sport di resistenza (corsa, ciclismo, nuoto);
- attività da palestra (allenamenti con pesi, circuiti, cardio);
- sport misti o personalizzati, che richiedano il monitoraggio di durata, intensità o calorie consumate.

Il sistema potrà essere utilizzato in contesti individuali, dove il singolo sportivo gestisce in autonomia i propri dati, ma potrà in futuro adattarsi anche a scenari più ampi, come community di appassionati o allenatori che monitorano i progressi dei propri atleti.

Dal punto di vista tecnologico, TrackFit sarà accessibile tramite interfaccia web e applicazione mobile, permettendo l'uso in qualsiasi momento e contesto. Ciò garantisce flessibilità e immediatezza, due aspetti fondamentali per un sistema che intende affiancare l'utente nella pratica quotidiana dello sport.

## **2. PROBLEM STATEMENT**

### ***2.1 Problem Domain***

#### **Dominio:**

Monitoraggio e analisi degli allenamenti multi-sport per utenti amatoriali e semi-professionisti.

#### **Motivazione:**

Le soluzioni attuali risultano frammentate o verticali, rendendo difficile ottenere una visione unificata dei progressi.

È necessario sviluppare un'interfaccia semplice e intuitiva, con report chiari e facilmente interpretabili.

#### **Contesto e obiettivi (dati dal documento di proposta):**

- Registrazione delle sessioni di allenamento
- Gestione dello storico
- Analisi e reportistica
- Esportazione dei dati
- Interfaccia web e mobile

**Termini chiave:**

- **Sessione di allenamento:** sport, data/ora, durata, distanza, intensità/frequenza cardiaca (opzionale), calorie, note.
- **Profilo utente**
- **Obiettivi**
- **Dashboard**
- **Report**
- **Esportazione:** CSV, PDF
- **Importazione:** GPX, TCX, API esterne

**Stakeholder:**

- Utente finale
- Coach (in prospettiva futura)
- Amministratore

**Confini del sistema:**

Il sistema include un'app mobile e una web app che raccolgono e mostrano dati, interagendo con servizi esterni standard per l'importazione dei dati.

Sono esclusi dispositivi proprietari e funzioni di tipo clinico.

La definizione esplicita dei confini del sistema sarà parte integrante della fase di identificazione del sistema.

## 2.2 *Scenarios*

**Premessa**

Di seguito vengono analizzati alcuni scenari d'uso comuni che l'ambiente di sviluppo dovrà prevedere e gestire.

I corrispondenti **Casi d'Uso** sono riportati nella *Scheda 3 – Casi\_d'uso\_TrackFit*.

### 2.2.1 *Scenario 1: Corsa Serale*

**Attore principale:** Tizio Caio (Utente amatoriale)

**Data di riferimento:** 27/10/2025

**Obiettivo:** Svolgere una corsa serale, registrarla con l'app mobile, verificarne i risultati sul web, aggiornare gli obiettivi e generare un report.

**Descrizione**

Tizio Caio si autentica sull'app **TrackFit** da smartphone e accede alla **Home**, dove visualizza:

- Chilometri della settimana
- Ritmo medio

- Grafico ultimi 30 giorni
- Avanzamento obiettivo

Dalla sezione *Allenamenti* seleziona **Pianifica** → **Nuova sessione**, con schede *Dettagli*, *Obiettivo*, *Sensori*, *Note/Tag*.

Imposta:

- Tipo: Corsa (outdoor)
- Data/Ora: 27/10/2025, ore 19:00
- Durata stimata: 45:00
- Intensità: Media
- Obiettivo: Durata 00:45:00
- Sensori: GPS ON, Cardio OFF
- Tag: *serale, parco*
- Nota: “Riscaldamento 10’, poi ritmo costante”
- 

Salva la sessione: il sistema attiva un **promemoria 10 minuti prima** dell’inizio.

Alle 18:50 riceve una **notifica push**:

“Allenamento corsa alle 19:00 — Obiettivo 45 min.”

Dalla notifica accede alla sessione pianificata e avvia il tracciamento. Il sistema acquisisce il fix GPS.

Nel pannello **Live** visualizza tempo, distanza, ritmo attuale/medio, barra di progresso; sono attivi auto-lap e annunci vocali a ogni km.

Durante la sessione:

- Al minuto 20 mette in pausa → cronometro fermato e evento registrato.
- Dopo 60” riprende.
- Alle 19:46 termina l’attività.

Visualizza il **riepilogo provvisorio**:

- Durata: 00:45:20
- Distanza: 8,05 km
- Ritmo medio: 5’38”/km
- Calorie: ≈616 kcal
- Dislivello+: 32 m

Aggiunge:

- Percezione sforzo: 7/10
- Nota: “Gambe ok, leggera fatica finale”
- Tag: *serale*

Il sistema sincronizza via Wi-Fi con il backend e archivia la sessione in **Storico/Calendario**, con mappa GPS e grafici ritmo/distanza.

La **Dashboard** aggiorna lo stato dell’obiettivo: *20,1/20 km (raggiunto)*.

Tizio pubblica la sessione sulla **Bacheca personale**, conferma, e riceve un’e-mail di conferma.

Da web accede al portale **TrackFit** → **Area Utente / Bacheca Allenamenti**, visualizza e

apre il dettaglio della corsa del 27/10/2025 (riepilogo, grafici, mappa, tag, note).

Il sistema:

- Segnala la sessione come *candidata al conteggio obiettivo settimanale*
- Verifica assenza duplicati (import GPX/TCX)
- Dopo conferma, marca la sessione come **Consolidata**

Infine, genera un **Report PDF** per ottobre 2025 (distanza totale, tempo, ritmo medio, trend settimanale, PR).

Scarica il documento (archiviato in “Report salvati”, disponibile anche in CSV).

Imposta un nuovo obiettivo settimanale di **25 km** e aggiorna la dashboard.

Da *Profilo*, esporta i dati come pacchetto ZIP (CSV + GPX) ricevuto via e-mail.

### 2.2.2 Scenario 2: Gestione orari da parte dell'allenatore

**Attore principale:** Marco Bianchi (Coach)

**Atleta associato:** Tizio Caio

**Data/Ora di riferimento:** 24/10/2025 – 09:00

#### Descrizione

Il Coach accede al portale web **TrackFit** → Home Coach (widget “Piani attivi”, “Atleti seguiti”, “Prossime sessioni”).

Pianifica un allenamento personale (25/10/2025 – 07:00-08:00, Parco Nord, nota “Lento rigenerante”).

Poi apre:

**Roster** → **Tizio Caio** → **Calendario Atleta** e trova la sessione “Corsa medio-lunga — 27/10/2025 19:00–20:00”.

Avvia una modifica oraria: propone **27/10/2025 18:00–19:00**.

Il sistema calcola il preavviso (81h) → R1 soddisfatta.

Inserisce motivazione: “Anticipazione per temperatura più bassa e luce diurna”.

Il sistema:

- Aggiorna il calendario atleta
- Invia e-mail automatica (R2)
- Registra log audit (timestamp, utente, variazioni, motivazione)
- Mostra banner “Orario aggiornato dal Coach per 27/10/2025” in Dashboard atleta

Successivamente (26/10/2025 – 14:15) il Coach tenta un ulteriore spostamento a 17:30 → preavviso ~27h15m → R1 violata.

Il sistema blocca l'operazione e mostra messaggio di errore con opzioni “Annulla” o “Proponi nota all'atleta”.

### 2.2.3 Scenario 3: Autenticazione (Portale Business)

**Attore principale:** Amministratore FitCorp ([admin@fitcorp.it](mailto:admin@fitcorp.it))



### **Descrizione**

L'amministratore accede a **TrackFit Business** → *Home Azienda*.

Naviga in **Area Azienda** → **Strutture** → **Nuova Struttura** e compila:

"FitCorp — Palestra Roma Centro",

P.IVA: 12345678901, CUN: FIT-ROMA-001,

Indirizzo: Via Appia 10, 00179 Roma (RM).

Il sistema esegue la verifica anti-duplicato (P.IVA, CUN, Denominazione+Indirizzo):

- Nessun duplicato → creazione STR-000912 → stato *Attiva*
- Registra audit e invia e-mail di conferma

Poi aggiunge un allenatore:

**Staff** → **Aggiungi Allenatore** → **m.bianchi.coach@fitcorp.it**

- Se utente esistente → associazione a STR-000912
- Se nuovo → creazione profilo e invito e-mail

Il sistema registra l'audit ("grant: role=Coach, STR-000912 → USR-00421") e invia e-mail di onboarding con ruolo, permessi e link di accesso.

## **2.2.4 Scenario 4: Prenotazione allenatore**

**Attore principale:** Tizio Caio (Utente)

**Coach:** Marco Bianchi

**Struttura:** FitCorp — Palestra Roma Centro

**Data/Ora di riferimento:** 26/10/2025 – 10:00

### **Descrizione**

L'utente accede a **TrackFit (web/mobile)** → *Dashboard* → *Coach / Trova Allenatore*.

Filtra per *Città=Roma, Struttura=FitCorp — Roma Centro, Disciplina=Corsa*.

Seleziona **Marco Bianchi**, legge bio, certificazioni e tariffe: "Sessione individuale 60' — €40".

Prenota lo slot **30/10/2025 – 19:00–20:00** → nessun conflitto.

Pagamento con carta VISA (SCA 3-D Secure OK, ID PAY-2025-10-26-00041).

Il sistema:

- Crea prenotazione **PRN-000873**, stato "Confermata"
- Invia e-mail a utente e coach con allegato *.ics*
- Blocca slot su entrambi i calendari
- Genera ricevuta **RCPT-000873.pdf**

In *Area Utente* → *Prenotazioni* appare PRN-000873 con pulsanti *Dettagli, Annulla, Sposta*. Promemoria automatici a T-48h e T-2h; check-in rapido via QR a T-15'.

### 2.2.5 Scenario 5: Attivazione abbonamento annuale con coupon e pagamento online

**Attore principale:** Tizio Caio (Utente)

**Struttura:** FitCorp — Palestra Roma Centro

**Prodotto:** Abbonamento Annuale (12 mesi)

**Data/Ora di riferimento:** 27/10/2025 – 18:00

#### **Descrizione**

L'utente apre l'e-mail "Benvenuto in FitCorp — Roma Centro: -20% sull'abbonamento annuale" (codice ROMA-ANN20-2025).

Segue il link "Attiva ora" → accede con *t.caio19@trackfit.app*.

Il sistema verifica i vincoli (struttura, piano, validità, unicità).

Prezzo base: €480,00 → Sconto 20% (−€96,00) + Quota attivazione €20,00 → **Totale €404,00.**

Effettua pagamento VISA (3-D Secure OK, ID PAY-2025-10-27-00421).

Il sistema crea abbonamento **SUB-001139**, stato *Attivo*, validità 27/10/2025 → 26/10/2026.

Coupon marcato *Usato*, fattura **INV-2025-10-00421 (PDF)** generata, tessera digitale con QR.

E-mail di conferma a utente e struttura.

In *Area Utente* → *I miei abbonamenti* compare SUB-001139 con opzioni *Mostra QR*, *Scarica fattura*, *Gestisci rinnovo*.

Alla prima visita, la reception valida il QR e registra l'ingresso.

## 2.3 Functional Requirements

### 2.3.1 Gestione Utenti, Allenatori e Gestori della Palestra

- Registrazione e autenticazione di utenti, allenatori e gestori.
- Modifica e aggiornamento dei profili personali.
- Gestione dei ruoli (utente, allenatore, gestore, amministratore).
- Possibilità di associare utenti a specifici allenatori o corsi.

#### **Implementazione Prevista per far fronte al requisito:**

- Utilizzo token per la gestione sicura delle sessioni.
- Architettura RESTful API per comunicazione tra frontend e backend.
- Database relazionale (MySQL) per la gestione delle tabelle.
- Funzioni CRUD per aggiornamento dei profili.

### 2.3.2 Gestione Palestre

- Creazione, modifica e cancellazione delle palestre.
- Configurazione dei corsi: nome, descrizione, orario, allenatore, capacità.
- Gestione degli abbonamenti: tipologia, durata, prezzo e rinnovo.

- Prenotazione dei corsi da parte degli utenti.

**Implementazione prevista per far fronte al requisito:**

- Struttura dati relazionale con tabelle Palestre, Corsi, Abbonamenti, Prenotazioni.
- Pannello di controllo per gestori tramite interfaccia web o mobile.
- Funzionalità di notifica email o push per conferma prenotazioni.

### **2.3.3 Monitoraggio delle Performance**

- Registrazione dei progressi individuali degli utenti (allenamenti, peso, parametri fisici).
- Visualizzazione dei progressi tramite dashboard personali e grafici.
- Notifiche automatiche per obiettivi raggiunti o scadenza dell'abbonamento.

**Implementazione prevista per far fronte al requisito:**

- App mobile con sincronizzazione automatica dei dati tramite API.
- Dashboard con grafici interattivi (libreria Chart.js o simili).

### **2.3.4 Reportistica e Analisi Aggregate**

- Generazione di report su iscrizioni, frequenza ai corsi e utilizzo delle strutture.
- Report specifici per gli allenatori sui progressi dei propri utenti.
- Report per i gestori relativi alla gestione complessiva della palestra.
- Report per l'amministratore centrale con dati aggregati di tutte le palestre connesse.

**Implementazione prevista per far fronte al requisito:**

- Generazione dinamica dei report tramite query SQL aggregate e viste predefinite.
- Dashboard con filtri temporali
- Accesso differenziato ai dati in base al ruolo.

### **2.3.5 Accessibilità e Interfaccia Intuitiva**

- Interfaccia responsiva e mobile-friendly.
- Accesso mobile con tutte le principali funzionalità.
- Ricerca e filtraggio di corsi, utenti e allenatori.
- Dashboard personalizzate per ogni ruolo.

**Implementazione prevista per far fronte al requisito:**

- Organizzazione del personale riguardo alla coerenza grafica.
- Design responsive (media query e componenti adattivi).

## **2.4 Non Functional Requirements**

### **2.4.1 Performance**

- Il sistema deve rispondere alle richieste standard (login, visualizzazione dashboard, prenotazione corsi) in meno di 2 secondi.
- La generazione di report complessi deve avvenire in meno di 5 secondi.

**Implementazione prevista per far fronte al requisito:**

- Caching lato server per query frequenti.
- Ottimizzazione delle query SQL con indici.

### **2.4.2 Scalabilità**

- Il sistema deve supportare un numero crescente di utenti, corsi e palestre senza degradare le performance.
- Deve essere possibile aggiungere nuove palestre o funzionalità senza interrompere il servizio.

#### **Implementazione prevista per far fronte al requisito:**

- Architettura microservizi o modulare con API indipendenti.
- Deploy su cloud provider (AWS, Azure o GCP) con bilanciamento del carico (In caso di espansione del progetto).

### **2.4.3 Sicurezza**

- Autenticazione sicura (password hashing, gestione token per sessioni).
- Gestione dei permessi in base al ruolo (utente, allenatore, gestore).
- Protezione dei dati sensibili (informazioni personali e parametri fisici).

#### **Implementazione prevista per far fronte al requisito:**

- Connessioni HTTPS e crittografia SSL/TLS.
- Controllo dei permessi lato backend con middleware di autorizzazione.

### **2.4.4 Affidabilità**

- Backup automatici giornalieri dei dati principali.
- Possibilità di recupero rapido in caso di guasto o perdita di dati.

#### **Implementazione prevista per far fronte al requisito:**

- Backup automatici giornalieri del database.
- Storage dei log e metriche su sistema centralizzato.
- Strategie di disaster recovery (replica dei dati).

### **2.4.5 Usabilità**

- Interfaccia chiara e intuitiva, adatta anche a utenti poco esperti.
- Supporto multilingua (almeno Italiano e Inglese).
- Compatibilità con i principali browser e dispositivi mobili.

#### **Implementazione prevista per far fronte al requisito:**

- Test di usabilità con utenti campione.
- Traduzioni gestite con librerie.
- Layout coerente, pulsanti e icone chiari.

### **2.4.6 Manutenibilità**

- Codice modulare e ben documentato per facilitare aggiornamenti e manutenzione.

#### **Implementazione prevista per far fronte al requisito:**

- Struttura del codice modulare e commentata.
- Versionamento del codice con Git e GitHub.

## 2.5 *Target Enviroment*

### 1. Utenti

- Dispositivi mobili (smartphone e tablet) con sistema operativo **Android**.
- Desktop e laptop con browser aggiornati.

### 2. Backend

- Storage sicuro per i dati di performance e reportistica.

### 3. Rete

- Supporto per connessioni mobili **4G/5G** sull'app mobile.
- Connessione Internet stabile per accesso web e sincronizzazione.

## 3. CASI D'USO

### 3.1 *Casi d'uso scenario 1:*

#### 1. UC 1 – Autenticazione (mobile)

**Attore:** Utente

**Entry Condition:** L'utente è sulla schermata di autenticazione dell'app *TrackFit*

**Flusso di eventi:**

1. L'utente inserisce username *t.caio@example.com* e password
2. L'utente invia i dati al sistema
3. Il sistema verifica le credenziali con esito positivo
4. Il sistema mostra la *Home/Dashboard*

**Exit Condition:** Utente autenticato sulla Home

**Alternativa:**

Se al punto 3 le credenziali non sono valide → **UC 1.1**

#### 2. UC 1.1 – Autenticazione fallita

**Attore:** Utente

**Entry Condition:** Tentativo di autenticazione non riuscito

**Flusso di eventi**

1. Il sistema mostra "username o password non corretti"

2. Il sistema ripresenta la schermata di autenticazione

**Exit Condition:** Utente sulla schermata di autenticazione

### 3. UC 2 – Pianificazione sessione

**Attore:** Utente

**Entry Condition:** Utente autenticato sulla Home

#### Flusso di eventi

1. L'utente apre *Allenamenti* → *Pianifica*
2. Il sistema mostra “Nuova sessione” con schede: **Dettagli**, **Obiettivo**, **Sensori**, **Note/Tag**
3. L'utente compila:
  - **Dettagli:** Corsa (outdoor), data/ora 27/10/2025 19:00, durata stimata 45:00, intensità *Media*
  - **Obiettivo:** tipo *Durata*, valore 00:45:00
  - **Sensori:** GPS *ON*; HR *OFF*
  - **Note/Tag:** “serale, parco”; nota “Riscaldamento 10”, poi ritmo costante”
4. L'utente salva la pianificazione
5. Il sistema conferma e attiva promemoria 10 minuti prima

**Exit Condition:** Sessione pianificata con promemoria attivo

**Alternativa:**

Se al punto 3 mancano dati obbligatori → UC 2.1

### 4. UC 2.1 – Dati pianificazione incompleti

**Attore:** Utente

**Entry Condition:** Tentativo di salvataggio pianificazione con campi obbligatori mancanti

#### Flusso di eventi

1. Il sistema evidenzia i campi mancanti e mostra messaggio di errore

2. L'utente completa i campi richiesti
3. L'utente salva nuovamente

**Exit Condition:** Pianificazione salvata

## 5. UC 3 – Promemoria avviso e tracciamenti

**Attore:** Utente

**Entry Condition:** Sessione pianificata con promemoria attivo

### Flusso di eventi

1. Alle 18:50 il sistema invia notifica “Allenamento corsa alle 19:00 — obiettivo 45 min.”
2. L'utente apre la notifica e accede a “Sessione pianificata”
3. L'utente avvia il tracciamento
4. Il sistema verifica stato GPS: ricerca → ok

**Exit Condition:** Sessione in stato “Live”

### Alternativa:

Se al punto 4 il GPS non è disponibile → **UC 3.1**

## 6. UC 3.1 – Gps non disponibile

**Attore:** Utente

**Entry Condition:** Tentativo di avvio con fix GPS non acquisito

### Flusso di eventi

1. Il sistema segnala “GPS non disponibile”
2. L'utente abilita il GPS e/o attende il fix
3. Il sistema conferma fix acquisito

**Exit Condition:** Tracciamento avviabile

## 7. UC 4 – Tracciamenti live

**Attore:** Utente

**Entry Condition:** Sessione in stato “Live”

**Flusso di eventi**

1. Il sistema mostra tempo, distanza, ritmo attuale/medio, barra progresso obiettivo
2. Il sistema emette annuncio vocale ogni 1 km con split e ritmo medio
3. Il sistema registra auto-lap e stima calorie

**Exit Condition:** Sessione in corso con dati registrati

**Alternativa:** Pausa e ripresa → **UC 4.1**

**8. UC 4.1 – Pausa e ripresa**

**Attore:** Utente

**Entry Condition:** Sessione in stato “Live”

**Flusso di eventi**

1. Al minuto 20 l'utente mette in Pausa
2. Il sistema congela cronometro e registra l'evento
3. Dopo 60" l'utente seleziona *Riprendi*
4. Il sistema prosegue il tracciamento

**Exit Condition:** Sessione continua dopo pausa

**9. UC 5 – Chiusura e salvataggio sessione**

**Attore:** Utente

**Entry Condition:** Sessione in corso

**Flusso di eventi**

1. Alle 19:46 l'utente seleziona *Termina*
2. Il sistema mostra riepilogo provvisorio:
  - Durata: 00:45:20
  - Distanza: 8,05 km
  - Ritmo medio: 5'38"/km
  - Calorie: ≈616 kcal
  - Dislivello+: 32 m



3. L'utente compila/modifica:
  - Percezione sforzo: 7/10
  - Note: "Gambe ok, leggera fatica finale"
  - Tag: "serale"
4. L'utente salva
5. Il sistema conferma il salvataggio

**Exit Condition:** Sessione salvata sul dispositivo

#### 10. UC 6 – Sincronizzazione e storicizzazione

**Attore:** Sistema

**Entry Condition:** Sessione salvata; connettività Wi-Fi disponibile

##### **Flusso di eventi**

1. Il sistema sincronizza la sessione verso backend
2. Il sistema archivia in *Storico* → *Calendario*
3. Il sistema associa mappa GPS, grafici ritmo/distanza, split per km
4. Il sistema aggiorna la Dashboard: 20,1/20 km (obiettivo raggiunto)

**Exit Condition:** Sessione presente in Storico; Dashboard aggiornata

**Alternativa:**

Assenza connettività → ritenta in background (stato "In attesa di sync")

#### 11. UC 7 – Pubblicazione su bacheca personale

**Attore:** Utente

**Entry Condition:** Sessione presente in Storico

##### **Flusso di eventi**

1. L'utente apre la sessione in Storico
2. L'utente seleziona "Pubblica su bacheca"
3. Il sistema richiede conferma
4. L'utente conferma

5. Il sistema imposta stato “Pubblicata” e invia email di conferma

**Exit Condition:** Sessione pubblicata

**Alternativa:**

Annullamento alla conferma → stato invariato

## 12. UC 8 – Accesso via web e verifica

**Attore:** Utente

**Entry Condition:** Sessione pubblicata; utente con credenziali web

**Flusso di eventi**

1. L'utente accede al portale *TrackFit* e si autentica
2. L'utente apre *Area Utente / Bacheca Allenamenti*
3. L'utente visualizza elenco sessioni Pubblicate
4. L'utente apre il dettaglio della corsa del 27/10/2025
5. Il sistema mostra riepilogo, grafici, mappa, tag, note

**Exit Condition:** Sessione visualizzata sul web

## 13. UC 9 – Accettazione per obiettivi e deduplica

**Attore:** Utente

**Entry Condition:** Sessione pubblicata; candidata al conteggio obiettivo settimanale

**Flusso di eventi**

1. Il sistema verifica duplicati su file import (GPX/TCX)
2. Il sistema mostra esito “nessun duplicato trovato”
3. L'utente seleziona “Accetta per obiettivo”
4. Il sistema richiede conferma
5. L'utente conferma
6. Il sistema imposta stato “Consolidata”

**Exit Condition:** Sessione consolidata e conteggiata nell'obiettivo

**Alternativa:**

Se al punto 1 viene rilevato un duplicato → **UC 9.1**

#### **14. UC 9.1 – Duplicativo rilevato**

**Attore:** Utente

**Entry Condition:** Esito verifica duplicati positivo

##### **Flusso di eventi**

1. Il sistema indica la sessione duplicata e le possibili azioni
2. L'utente sceglie se unire, tenere solo una sessione o annullare
3. Il sistema applica l'azione scelta

**Exit Condition:** Duplicazione risolta; sessione pronta o esclusa dal conteggio

#### **15. UC 10 – Generazione report**

**Attore:** Utente

**Entry Condition:** Sessioni presenti in Storico

##### **Flusso di eventi**

1. L'utente apre *Report* → *Genera PDF*
2. L'utente seleziona periodo "Ottobre 2025" e metriche:
  - Distanza totale
  - Tempo
  - Ritmo medio
  - Trend settimane
  - PR
3. L'utente avvia generazione
4. Il sistema crea il PDF
5. Il sistema rende disponibile il download e archivia in "Report salvati"
6. Il sistema offre anche "Esporta CSV"

**Exit Condition:** Report PDF scaricato; CSV disponibile

## 16. UC 11 – Aggiornamento obiettivi

**Attore:** Utente

**Entry Condition:** Obiettivo settimanale corrente raggiunto (100%)

**Flusso di eventi**

1. L'utente apre *Obiettivi*
2. L'utente imposta nuovo obiettivo 25 km per la prossima settimana
3. L'utente salva
4. Il sistema aggiorna la Dashboard con il nuovo target

**Exit Condition:** Nuovo obiettivo impostato

## 17. UC 12 – Esportazione dati personali

**Attore:** Utente

**Entry Condition:** Utente autenticato

**Flusso di eventi**

1. L'utente apre *Profilo* → *Esporta dati*
2. L'utente seleziona *Pacchetto ZIP (CSV + GPX)*
3. L'utente conferma
4. Il sistema prepara l'archivio
5. Il sistema invia link di download via email

**Exit Condition:** Link di download ricevuto via email

## 3.2 Casi d'uso scenario 2:

### 1. UC 1 – Autenticazione Coach

**Attore:** Coach

**Entry Condition:** Coach sulla schermata di autenticazione del portale *TrackFit*

### **Flusso di eventi**

1. Inserisce username `m.bianchi@coach.trackfit.it` e password
2. Invia le credenziali
3. Il sistema verifica e autentica
4. Il sistema mostra *Home Coach* con widget “Piani attivi”, “Atleti seguiti”, “Prossime sessioni”

**Exit condition:** Coach autenticato sulla Home coach

**Alternativa:** Credenziali errate → UC 1.1

### **2. UC 1.1 – Autenticazione fallita**

**Attore:** Coach

**Entry Condition:** Tentativo di autenticazione non riuscito

#### **Flusso di eventi:**

1. Il sistema mostra “username o password non corretti”
2. Ripresenta la schermata di autenticazione

**Exit condition:** Coach sulla schermata di autenticazione

### **3. UC 2 – Pianificazione allenamento personale del Coach**

**Attore:** Coach

**Entry Condition:** Coach autenticato sulla Home Coach

#### **Flusso di eventi:**

1. Apre *Calendario Coach* → *Nuovo allenamento personale*
2. Compila:

- **Sport:** Corsa
- **Data/ora:** 25/10/2025 07:00-08:00
- **Luogo:** Parco Nord
- **Note:** “Lento rigenerante”

3. Salva

4. Il sistema registra l’evento nel Calendario Coach

5. Il sistema verifica conflitti con impegni del Coach

6. Il sistema applica badge “Personale” all’evento

**Exit condition:** Evento personale creato e visibile nel Calendario Coach

**Alternative:**

- Conflitto rilevato → UC 2.1
- Campi obbligatori mancanti → UC 2.2

#### **4. UC 2.1 – Conflitto nel Calendario Coach**

**Attore:** Coach

**Entry Condition:** Salvataggio evento con sovrapposizione ad altro impegno

**Flusso di eventi:**

1. Il sistema evidenzia il conflitto (slot occupato)

2. Offre opzioni: *modifica orario / salva come provvisorio / annulla*

3. Il Coach seleziona un'opzione

4. Il sistema applica la scelta

**Exit condition:** Conflitto risolto o operazione annullata

## 5. UC 2.2 – Dati incompleti per allenamento personale

**Attore:** Coach

**Entry Condition:** Tentativo di salvataggio con campi obbligatori vuoti

**Flusso di eventi:**

1. Il sistema segnala i campi mancanti

2. Il Coach completa i campi

3. Salva nuovamente

**Exit condition:** Evento personale salvato

## 6. UC 3 – Verifica piano atleta

**Attore:** Coach

**Entry Condition:** Coach autenticato; atleta presente nel roster

**Flusso di eventi:**

1. Apre Area Coach → Roster → selezione Tizio Caio

2. Apre Calendario Atleta

3. Individua la seduta “Corsa medio-lunga – 27/10/2025 19:00-20:00”

**Exit condition:** Seduta dell'atleta identificata nel Calendario Atleta

#### 7. UC 4 – Modifica orario atleta con preavviso valido ( $\geq 48h$ )

**Attore:** Coach

**Entry Condition:** Seduta atleta selezionata; data/ora corrente 24/10/2025 09:00

**Flusso di eventi:**

1. Selezione “Modifica orario” sulla seduta del 24/10/2025 19:00 – 20:00
2. Propone nuovo slot 27/10/2025 18:00 – 19:00
3. Il sistema calcola il preavviso 24/10/2025 09:00 → 27/10/2025=81h
4. Il sistema valida R1 → esito OK
5. Il Coach inserisce **Motivazione:** “Anticipazione per temperatura più bassa e luce diurna”
6. Conferma la modifica
7. Il sistema aggiorna il Calendario Atleta allo slot 18:00 – 19:00
8. Il sistema invia e-mail automatica all'atleta R2 con:
  - **Oggetto:** “TrackFit – Modifica orario allenamento 27/10/2025 18:00 – 19:00
  - **Corpo:** “Ciao Tizio, il tuo allenamento Corsa medio-lunga del 27/10/2025 è stato modificato da 19:00 – 20:00 a 18:00 – 19:00. Motivazione: Anticipazione per temperatura più bassa e luce diurna. Calendario aggiornato in TrackFit – Coach:



“Marco Bianchi”

° **Allegato opzionale:** invito .ics con il nuovo slot

9. Il sistema registra **log di audit:** timestamp, utente, valori prima/dopo, motivazione

**Exit condition:** Seduta aggiornata; e-mail inviata; audit registrato

## 8. UC 5 – Notifiche e controllo post-modifica lato Atleta

**Attore:** Sistema

**Entry Condition:** Modifica approvata e applicata (UC 4 completata)

### Flusso di eventi

1. Il sistema mostra banner su *Dashboard Atleta*: “Orario aggiornato dal Coach per 27/10/2025”

2. Il sistema aggiunge in *Attività recenti*: “Modifica orario da 19:00 a 18:00 – 27/10/2025

**Exit Condition:** Notifiche presenti sull’account dell’atleta

## 9. UC 6 – Tentativo di modifica non consentita ( < 48h)

**Attore:** Coach

**Entry Condition:** Data/ora corrente 26/10/2025 14:15; seduta atleta 27/10/2025 18:00–19:00

### Flusso di eventi:

1. Il Coach tenta di spostare la seduta da 18:00 a 17:30

2. Il sistema calcola il pravviso: 27h 15m

3. Il sistema valuta R1 → violata

4. Il sistema blocca l'operazione

5. Il sistema mostra il messaggio: "Modifica orario negata: preavviso minimo 48 ore non rispettato"

6. Il sistema offre azioni: *Annulla* | *Proponi nota all'atleta nel piano*

**Exit condition:** Nessuna variazione a calendario; nessuna e-mail inviata

**Alternativa:** Selezione "Proponi nota all'atleta" → UC 6.1

### 10. UC 6.1 – Proposta nota al piano atleta

**Attore:** Coach

**Entry Condition:** Modifica bloccata per R1; Coach sceglie "Proponi nota all'atleta"

**Flusso di eventi:**

1. Inserisce testo nota (es. "Valuta anticipo a 17:30 se possibile; conferma via messaggi").

2. Invia la nota

3. Il sistema registra la nota nel piano atleta

4. Il sistema notifica l'atleta (notifica in-app: nessuna e-mail di modifica oraria)

**Exit condition:** Nota registrata; pianificazione inviata.

## 3.3 Casi d'uso scenario 3:

### 1. UC 1 – Autenticazione Amministratore

**Attore:** Amministratore Azienda

**Entry Condition:** Schermata di accesso *TrackFit Business*

**Flusso di eventi:**

1. Inserisce username [admin@fitcorp.it](mailto:admin@fitcorp.it) e password FitCorp-2025!
2. Invia le credenziali
3. Il sistema autentica
4. Il sistema mostra *Home Azienda* con widget **Strutture, Staff, Notifiche, Attività recenti**

**Exit Condition:** Amministratore autenticato sulla Home Azienda.

**Alternativa:** Credenziali errate → UC 1.1

## **2. UC 1.1 – Autenticazione fallita**

**Attore:** Amministratore Azienda

**Entry Condition:** Tentativo di accesso non riuscito

**Flusso di eventi:**

1. Il sistema mostra “username o password non corretti”
2. Ripresenta la schermata di autenticazione

**Exit condition:** Utente sulla schermata di autenticazione.

## **3. UC 2 – Avvio creazione Struttura**

**Attore:** Amministratore Azienda

**Entry Condition:** Utente autenticato sulla Home Azienda

**Flusso di eventi:**

1. Apre *Area Azienda* → *Strutture* → *Nuova Struttura*

2. Il sistema mostra il form “Crea Struttura” con sezioni **Dati anagrafici, Contatti, Localizzazione, Classificazione**

**Exit condition:** Form di creazione Struttura aperto.

#### 4. UC 3 – Compilazione dati Struttura

**Attore:** Amministratore Azienda

**Entry Condition:** Form “Crea Struttura” aperto

**Flusso di eventi:**

1. Compila:

◦ **Denominazione:** “FitCorp – Palestra Roma Centro”

◦ **Partita IVA:** 12345678901

◦ **Codice Struttura (CUN):** FIT-ROMA-001

◦ **Categoria:** Palestra

◦ **Indirizzo:** Via Appia 10, 00179 Roma (RM)

◦ **Telefono:** 06 1234567

◦ **E-mail:** [roma.centro@fitcorp.it](mailto:roma.centro@fitcorp.it)

◦ **Servizi:** sala pesi; percorsi cardio; spogliatoi M/F

2. Seleziona “Prosegui”

3. Il sistema valida i campi obbligatori

**Exit condition:** Dati pronti per verifica anti-duplicato

## **5. UC 3.1 – Dati incompleti Struttura**

**Attore:** Amministratore Azienda

**Entry Condition:** Tentativo di proseguire con campi mancanti

**Flusso di eventi:**

1. Il sistema evidenzia i campi mancanti e mostra messaggio di errore
2. L'amministratore completa i campi
3. Riprova "Prosegui"

**Exit condition:** Dati validati

## **6. UC 4 – Verifica anti-duplicato Struttura**

**Attore:** Sistema

**Entry Condition:** Dati Struttura validati

**Flusso di eventi:**

1. Esegue controllo su chiavi: **P.IVA, CUN, Denominazione+Indirizzo**
2. Se nessun duplicato certo o probabile → **Esito A**
3. Se rileva possibili duplicati → **Esito B**, con lista risultati
4. Applica vincoli **R1-R2**

**Exit condition:** Uno tra Esito A o Esito B mostrato all'utente.

## **7. UC 4.1 – Esito A – Nessun duplicato**

**Attore:** Amministratore Azienda

**Entry Condition:** Anti-duplicato senza corrispondenze

**Flusso di eventi:**

1. Il sistema mostra banner verde “Struttura non presente, puoi procedere con la creazione”
2. L'amministratore avvia *Riepilogo* → *Conferma*.

**Exit condition:** Passaggio alla creazione (UC 5).

## 8. UC 4.2 – Esito B – Possibile duplicato

**Attore:** Amministratore Azienda

**Entry Condition:** Lista possibili duplicati mostrata

**Flusso di eventi:**

1. L'amministratore sceglie un'azione:

° **B1:** *Apri scheda struttura trovata* → se **Proprietario = nessuno** → “Associa alla mia azienda” → *Conferma*.

° **B2:** *Richiedi subentro* → invia richiesta al proprietario corrente

° **B3:** *Ignora e continua* → consentito solo se nessuna chiave univoca coincide

2. Il sistema applica l'azione scelta e registra audit

**Exit condition:**

° **B1** → Struttura associata; si prosegue a gestione **Staff (UC 6)**

° **B2** → Stato “In attesa di subentro”; operazioni bloccate fino ad

approvazione

° **B3** → Passaggio alla creazione (UC 5)

**Alternativa:** Collisione chiavi univoche → UC 4.3

## 9. UC 4.3 – Blocco per collisione chiavi univoche

**Attore:** Sistema

**Entry Condition:** P.IVA o CUN coincidono con un record esistente

**Flusso di eventi:**

1. Mostra errore: “Creazione bloccata: P.IVA/CUN già presenti”
2. Offre solo azioni consentite: *Apri scheda* → *Associa* / *Richiedi subentro*

**Exit condition:** Creazione impedita; percorso forzato su associazione o subentro.

## 10. UC 5 – Creazione Struttura

**Attore:** Amministratore Azienda

**Entry Condition:** Anti-duplicato superato (**Esito A** o **B3**)

**Flusso di eventi:**

1. Visualizza *Riepilogo*.
2. Conferma creazione
3. Il sistema crea la Struttura, assegna **ID STR-000912**
4. Il sistema registra audit “create: STR-000912, by

[admin@fitcorp.it](mailto:admin@fitcorp.it), timestamp

5. Il sistema invia e-mail di conferma a [roma.centro@fitcorp.it](mailto:roma.centro@fitcorp.it)

6. Il sistema aggiunge voce in *Attività recenti*

7. Il sistema imposta **Stato struttura = Attiva**

**Exit condition:** STR-000912 creata e attivata

## 11. UC 6 – Associazione Struttura esistente alla mia azienda

**Attore:** Amministratore Azienda

**Entry Condition:** Scelta **B1** “Associa alla mia azienda”, proprietario assente

**Flusso di eventi:**

1. Conferma associazione

2. Il sistema associa la struttura all’azienda

3. Il sistema registra audit “associate: structure → FitCorp, by [admin@fitcorp.it](mailto:admin@fitcorp.it), timestamp”

4. Il sistema notifica in *Attività recenti*

**Exit condition:** Struttura associata a **FitCorp**, disponibile la gestione **Staff**.

## 12. UC 7 – Aggiunta allenatore alla struttura

**Attore:** Amministratore Azienda

**Entry Condition:** Struttura attiva o associata all’azienda

**Flusso di eventi:**

1. Apre *Strutture* → “FitCorp – Palestra Roma Centro” → *Staff* →  
*Aggiungi allenatore*



2. Il sistema mostra “Nuovo allenatore” con modalità:

° **Invita tramite e-mail** (crea utente Coach se non esiste)

° **Associa utente esistente** (ricerca per e-mail/CF)

**Exit condition:** Form pronto per verifica allenatore

### 13. UC 8 – Verifica esistenza allenatore

**Attore:** Amministratore Azienda

**Entry Condition:** Form “Nuovo Allenatore” aperto

**Flusso di eventi:**

1. Inserisce e-mail [m.bianchi.coach@fitcorp.it](mailto:m.bianchi.coach@fitcorp.it)

2. Seleziona “Verifica”

3. Il sistema risponde:

° **Esito C – Utente esistente:** Mostra nome, cognome, **ID** **USR-00421**, certificazioni, strutture associate

° **Esito D – Utente non trovato:** richiede dati minimi per creazione

**Exit condition:** Percorso **C** o **D**

### 14. UC 8.1 – Associazione utente esistente

**Attore:** Amministratore Azienda

**Entry Condition:** **Esito C — Utente esistente**

**Flusso di eventi:**

1. Seleziona “Associa a **STR-000912**”

## 2. Conferma

**Exit condition:** Passaggio a definizione permessi (UC 9).

### 15. UC 8.2 – Creazione account coach e invito

**Attore:** Amministratore Azienda

**Entry Condition:** Esito D — Utente non trovato

**Flusso di eventi:**

#### 1. Inserisce:

° **Nome:** Marco

° **Cognome:** Bianchi

° **E-mail:** [m.bianchi.coach@fitcorp.it](mailto:m.bianchi.coach@fitcorp.it)

° **Discipline:** forza; cardio

° **Certificazioni:** “FIPE B-Level”

° **Ruolo:** Coach

#### 2. Seleziona “Crea e invia invito”

**Exit condition:** Account coach creato; passaggio a definizione permessi (UC 9)

### 16. UC 9 – Conferma permessi e associazione alla struttura

**Attore:** Amministratore Azienda

**Entry Condition:** Allenatore selezionato o creato

**Flusso di eventi:**

1. Rivede il riepilogo abilitazioni per **STR-000912**:

◦ **Visibilità calendario:** solo struttura Roma Centro

◦ **Gestione piani/obiettivi atleti:** consentita

◦ **Modifica orari atleti:** consentita (**R4**)

2. Seleziona “Conferma” → “Salva”

3. Il sistema associa l’allenatore alla struttura

4. Il sistema invia l’email automaticamente all’allenatore:

◦ Se **nuovo**: link “Attiva account”

◦ Se **esistente**: notifica di nuova associazione

5. Il sistema registra audit “grant: role=Coach, STR-000912 → USR-00421

**Exit condition:** Allenatore associato con permessi attivi

## **17. UC 10 – Notifiche e onboarding allenatore**

**Attore:** Sistema

**Entry Condition:** Associazione completata (**UC 9**)

**Flusso di eventi:**

1. Invia e-mail a [m.bianchi.coach@fitcorp.it](mailto:m.bianchi.coach@fitcorp.it)

◦ **Oggetto:** “TrackFit – Assegnazione alla struttura FitCorp – Palestra Roma Centro”

◦ **Contenuto:** Riepilogo ruolo, permessi, link *Attiva account/Accedi*

° **Allegato:** invito .ics per briefing introduttivo

2. Registra voce in *Attività recenti* della struttura

**Exit condition:** Coach notificato, onboarding predisposto.

### 3.4 *Casi d'uso scenario 4:*

#### 1. UC 1 – Autenticazione utente

**Attore:** Utente

**Entry Condition:** Schermata di accesso *TrackFit*

**Flusso di eventi:**

1. Inserisce username [t.caio19@trackfit.it](mailto:t.caio19@trackfit.it) e password
2. Invia le credenziali
3. Il sistema autentica
4. Il sistema mostra *Home/Dashboard* con **progresso obiettivi, ultimo storico, notifiche.**

**Exit condition:** Utente autenticato sulla Dashboard

**Alternativa:** Credenziali errate → UC 1.1

#### 2 UC 1.1 – Autenticazione fallita

**Attore:** Utente

**Entry Condition:** Tentativo di accesso non riuscito

**Flusso di eventi:**

1. Il sistema mostra “username o password non corretti”

2. Ripresenta la schermata di autenticazione

**Exit condition:** Utente sulla schermata di autenticazione

### 3. UC 2 – Ricerca allenatore

**Attore:** Utente

**Entry Condition:** Utente autenticato

**Flusso di eventi:**

1. Apre *Area utente* → *Coach / Trova allenatore*
2. Imposta filtri: **Città=Roma; Struttura=FitCorp; Disciplina = Corsa.**
3. Il sistema mostra risultati tra cui “Marco Bianchi – Coach – Disponibile lun/mer/sab – Rating 4.8/5”
4. L’utente apre il profilo del coach

**Exit condition:** Profilo coach aperto

**Alternativa:** Nessun risultato → UC 2.1

### 4. UC 2.1 – Nessun coach trovato

**Attore:** Utente

**Entry Condition:** Ricerca senza risultati

**Flusso di eventi:**

1. Il sistema mostra “Nessun coach corrispondente ai filtri”
2. Azioni disponibili: *modifica filtri / annulla.*

**Exit condition:** Utente decide se ripetere la ricerca o uscire

## **5. UC 3 – Visualizzazione profilo coach e slot disponibili**

**Attore:** Utente

**Entry Condition:** Profilo del coach aperto

**Flusso di eventi:**

1. Il sistema mostra **bio, certificazioni, struttura associata, tariffa:** “Sessione individuale 60’ – 40 Euro”, **politiche di modifica/cancellazione.**

2. Il sistema mostra calendario integrato con slot disponibili

3. L’utente seleziona lo slot **30/10/2025 19:00 – 20:00** e preme **Prenota**

**Exit condition:** Avvio prenotazione sullo slot selezionato

**Alternativa:** Slot non più disponibile → UC 3.1

## **6. UC 3.1 – Slot non più disponibile**

**Attore:** Utente

**Entry Condition:** Selezione slot invalidata da prenotazione concorrente

**Flusso di eventi:**

1. Il sistema mostra messaggio “Slot non disponibile”

2. Il sistema riporta al calendario del coach

**Exit condition:** Prenotazione non avviata; calendario aggiornato.

## **7. UC 4 – Compilazione modulo Prenota sessione**

**Attore:** Utente  
**Entry Condition:** Avvio prenotazione

**Flusso di eventi:**

1. Seleziona/compila:

° **Servizio:** Personal Training 1:1

° **Struttura:** FitCorp – Palestra Roma Centro

° **Data/ora:** 30/10/2025 19:00 – 20:00

° **Note per il coach:** “Focus ritmo costante”

2. Continua

**Exit condition:** Dati pronti per verifica

**Alternativa:** Campi obbligatori mancanti → UC 4.1

**8. UC 4.1 – Dati incompleti nel modulo**

**Attore:** Utente  
**Entry Condition:** Tentativo di continuare con campi mancanti

**Flusso di eventi:**

1. Il sistema evidenzia i campi obbligatori

2. L'utente completa

3. Continua

**Exit condition:** Modulo validp

## 9. UC 5 – Verifica di sistema pre-riepilogo

**Attore:** Sistema

**Entry Condition:** Modulo valido

**Flusso di eventi:**

1. Verifica conflitti sul **Calendario utente** → nessuna sovrapposizione
2. Verifica disponibilità de coach sullo slot → **libero**
3. Applica regole **R1**: soddisfatta
4. Calcola preventivo: **40 Euro IVA compresa**

**Exit condition:** Esito verifiche OK; si procede al riepilogo

**Alternative:**

- ° Conflitto utente rilevato → **UC 5.1**
- ° Slot occupato durante la verifica → **UC 5.2**

## 10. UC 5.1 – Conflitto calendario utente

**Attore:** Utente

**Entry Condition:** Evento personale sovrapposto

**Flusso di eventi:**

1. Il sistema segnala conflitto con dettaglio dell'evento
2. Azioni: *cambia slot / annulla*

**Exit condition:** Conflitto risolto o prenotazione annullata



## 11. UC 5.2 – Disponibilità coach cambiata

**Attore:** Sistema

**Entry Condition:** Lo slot diventa occupato

**Flusso di eventi:**

1. Il sistema segnala l'indisponibilità
2. Ritorno al calendario con slot aggiornati

**Exit condition:** Scelta di un nuovo slot o annullamento

## 12. UC 6 – Riepilogo e pagamento con SCA

**Attore:** Utente

**Entry Condition:** Verifiche OK

**Flusso di eventi:**

1. Il sistema mostra il riepilogo: **servizio, struttura, data/ora, tariffa, politiche, note**
2. L'utente seleziona metodo: **Carta VISA \*\*\*\*\* 4242 (salvata)**
3. "Paga e conferma" → apertura **3-D Secure** dal provider
4. Immissione **OTP 998877 + PIN 123456** → Conferma
5. Esito pagamento **OK**, il sistema genera **ID transazione PAY-2025-10-26-00041**

**Exit condition:** Pagamento riuscito

**Alternativa:** Pagamento rifiutato/SCA fallita → UC 6.1

## 13. UC 6.1 – Pagamento non riuscito

**Attore:** Utente

**Entry Condition:** Errore gateway, SCA fallita o fondi insufficienti

**Flusso di eventi:**

1. Il sistema mostra il motivo del rifiuto
2. Il sistema riporta al riepilogo

**Exit condition:** Prenotazione non creata; utente può riprovare con altro metodo

#### 14. UC 7 – Creazione prenotazione e notifiche

**Attore:** Sistema

**Entry Condition:** Pagamento riuscito

**Flusso di eventi:**

1. Crea **PRN-000873** con stato “**Confermata**”
2. Invia e-mail all’utente:
  - **Oggetto:** “Conferma prenotazione – Marco Bianchi – 30/10/2025 19:00”
  - **Corpo:** riepilogo, politiche, link **Annulla/Sposta**
  - **Allegato:** invito .ics
3. Invia e-mail al coach:
  - **Oggetto:** “Nuova sessione prenotata – Tizio Caio – 30/10/2025 19:00”
  - **Corpo:** note dell’utente, link **Visualizza / Proponi modifica**

4. Blocca lo slot su **Calendario coach** e **Calendario Utente**

5. Genera **ricevuta RCPT-000873.pdf**

**Exit condition:** Prenotazione confermata, notifiche inviate, calendari aggiornati

**Alternativa:** Policy di conferma manuale del coach → stato in “**Attesa di conferma**”

## 15. UC 8 – Schermata post-azione e gestione prenotazione

**Attore:** Utente

**Entry Condition:** PRN-000873 creata

**Flusso di eventi:**

1. *Area utente* → *Prenotazioni* mostra **PRN-000873** con stato “**Conferma**”

2. Azioni disponibili:

◦ **Dettagli**

◦ **Annulla**

◦ **Sposta**

3. **Ricevuta RCPT-000873.pdf** scaricabile

**Exit condition:** Prenotazione visibile e gestibile

## 16. UC 8.1 – Annullamento prenotazione

**Attore:** Utente

**Entry Condition:** PRN-000873 in stato “**Confermata**”

**Flusso di eventi:**

1. L'utente seleziona **“Annulla”**
2. Il sistema calcola tempo residuo:
  - ° Se  $\geq 24h$  → **penale 0% (R2)**
  - ° Se  $< 24h$  → **penale 50% (R2)**
3. L'utente conferma
4. Il sistema libera lo slot sui calendari
5. Il sistema invia e-mail di annullamento a utente e coach con esito economico (eventuale penale)
6. Il sistema registra eventuale **rimborso/parziale** e aggiorna la **ricevuta**

**Exit condition:** Prenotazione annullata secondo policy; calendari aggiornati

**17. UC 8.2 – Spostamento prenotazione**

**Attore:** Utente

**Entry Condition:** PRN-000873 in stato “Confermata”

**Flusso di eventi:**

1. L'utente seleziona **“Sposta”**
2. Il sistema mostra slot compatibili del coach
3. L'utente sceglie un nuovo slot
4. Il sistema verifica disponibilità e **aggiorna la prenotazione**

5. Il sistema invia e-mail di aggiornamento a utente e coach con nuovo **.ics**

6. Il sistema applica le **policy economiche** se lo spostamento implica annullamento del precedente slot a **<24h**

**Exit condition:** Prenotazione spostata o operazione annullata

## 18. UC 9 – Promemoria e check-in

**Attore:** Sistema

**Entry Condition:** Prenotazione confermata

**Flusso di eventi:**

1. Invia **promemoria automatici** a **T-48h** e **T-2h** (push/e-mail)
2. Abilita **check-in rapido in app** da **T-15'**
3. Genera **QR** per accoglienza in struttura

**Exit condition:** Promemoria inviati; check-in disponibile

## 3.5 Casi d'uso scenario 5:

### 1. UC 1 – Apertura e-mail con coupon e deep link

**Attore:** Utente

**Entry Condition:** E-mail promozionale ricevuta

**Flusso di eventi:**

1. Apre l'e-mail con oggetto "Benvenuto in FitCorp — Roma Centro: -20% sull'abbonamento annuale"
2. Visualizza codice **ROMA-ANN20-2025** e seleziona "**Attiva ora**"
3. Il sistema apre *TrackFit* alla pagina **Abbonamenti** della struttura

**Exit condition:** Pagina *Abbonamenti* aperta via deep link

**Alternativa:** E-mail/codice non più valido → **UC 3.1**

## **2. UC 2 – Autenticazione e precompilazione coupon**

**Attore:** Utente

**Entry Condition:** Deep link attivo; utente non autenticato

**Flusso di eventi:**

1. Inserisce username [t.caio19@trackfit.it](mailto:t.caio19@trackfit.it) e password
2. Invia le credenziali
3. Il sistema autentica e reindirige alla scheda “**Annuale 12 mesi**” con campo coupon pre-compilato **ROMA-ANN20-2025**

**Exit condition:** Utente autenticato su scheda prodotto con coupon valorizzato

**Alternativa:** Credenziali errate → **UC 2.1**

## **3. UC 2.1 Autenticazione fallita**

**Attore:** Utente

**Entry Condition:** Tentativo di accesso non riuscito

**Flusso di eventi:**

1. Il sistema mostra “username o password non corretti”
2. Ripresenta la schermata di autenticazione

**Exit condition:** Utente sulla schermata di accesso

#### 4. UC 3 – Selezione piano e verifiche preliminari

**Attore:** Utente

**Entry Condition:** Scheda prodotto visibile con coupon valorizzato

**Flusso di eventi:**

1. Seleziona “Scegli piano”

2. Il sistema verifica:

° **Struttura target** = Roma Centro → OK

° **Piano compatibile** = Annuale → OK

° **Assenza abbonamenti attivi** dell'utente sulla stessa struttura → OK

° **Validità coupon** al tempo  $T = 27/10/2025\ 18:00 < 31/10/2025\ 23:59$  → OK

° **Limiti d'uso:** per utente 0/1; globale < limite → OK

° **Non cumulabile:** assenza di altri sconti nel carrello → OK

**Exit condition:** Verifiche superate; passaggio al carrello

**Alternative:** Una qualunque verifica fallisce → UC 3.1

#### 5. UC 3.1 – Coupon non valido / non applicabile

**Attore:** Sistema

**Entry Condition:** Almeno una condizione **R1** fallita

**Flusso di eventi:**

1. Il sistema indica il motivo specifico: **scaduto; struttura non**

**corrispondente; piano non compatibile; già utilizzato;  
abbonamento attivo presente; sconto cumulabile vietato**

2. Impedisce l'applicazione del coupon e propone **proseguimento  
senza sconto o annullamento**

**Exit condition:** Coupon rimosso/inefficace; l'utente decide se continuare o uscire

## **6. UC 4 – Carrello e applicazione coupon**

**Attore:** Sistema

**Entry Condition:** Verifiche superate

**Flusso di eventi:**

1. Calcola il riepilogo economico secondo **R2**:

° **Prezzo base** €480,00

° **Sconto 20% (coupon)** -€96,00

° **Quota attivazione (non scontabile)** €20,00

° **Totale** €404,00 (*IVA inclusa da listino*)

2. Mostra **“Procedi al pagamento”**

**Exit Condition:** Carrello pronto al checkout

**Alternativa:** Presenza di un altro sconto → **rimozione automatica** per non cumulabilità e **ricalcolo**

## **7. UC 5 – Consensi e condizioni**

**Attore:** Utente

**Entry Condition:** Carrello pronto



### **Flusso di eventi**

1. Visualizza **Termini e condizioni**: uso struttura, privacy, **rinnovo automatico (default OFF)**
2. Seleziona i **flag obbligatori**
3. Continua

**Exit Condition:** Consensi registrati

**Alternativa:** Flag mancanti → messaggio errore; impossibile proseguire

## **8. UC 6 – Pagamento con SCA (3-D Secure)**

**Attore:** Utente

**Entry Condition:** Consensi registrati; importo **€404,00**

### **Flusso di eventi**

1. Seleziona **Carta VISA \*\*\*\* 4242**
2. Clic su **“Paga €404,00”** → apertura **3-D Secure**
3. Inserisce **OTP 998877** → *Conferma*
4. Esito **“Autorizzato”**; il sistema genera **ID transazione PAY-2025-10-27-00421**

**Exit Condition:** Pagamento autorizzato (**R3**)

**Alternativa:** Pagamento rifiutato/SCA fallita → **UC 6.1**

## **9. UC 6.1 – Pagamento non riuscito**

**Attore:** Utente

**Entry Condition:** Errore gateway, OTP errata, fondi insufficienti

**Flusso di eventi**

1. Il sistema mostra motivo del rifiuto
2. Ritorna al riepilogo pagamento

**Exit Condition:** Nessuna creazione abbonamento; l'utente può riprovare

**10. UC 7 – Creazione abbonamento e documenti**

**Attore:** Sistema

**Entry Condition:** Pagamento autorizzato

**Flusso di eventi**

1. Crea **SUB-001139** con:
  - **Stato:** Attivo
  - **Validità:** 27/10/2025 → 26/10/2026
  - **Rinnovo:** disattivato
2. Marca il coupon **ROMA-ANN20-2025** come “Usato” (R1)
3. Emette **fattura INV-2025-10-00421 (PDF)**
4. Genera **tessera digitale con QR** per accesso tornelli
5. Genera **.ics “Benvenuto/Briefing” (opzionale)**
6. Registra **audit**.

**Exit Condition:** Abbonamento attivo; documenti generati

## 11. UC 8 – Notifiche e archiviazione

**Attore:** Sistema

**Entry Condition:** SUB creato

### Flusso di eventi

1. Invia e-mail all'utente:
  - **Oggetto:** “Conferma abbonamento annuale — FitCorp Roma Centro”
  - **Allegati:** fattura PDF; tessera digitale (link wallet); .ics
2. Invia e-mail alla struttura: **“Nuovo abbonamento attivato — Tizio Caio — SUB-001139”**
3. Archivia i documenti nell'**Area Utente**

**Exit Condition:** Notifiche inviate; documenti disponibili

## 12. UC 9 – Gestione da Area Utente

**Attore:** Utente

**Entry Condition:** SUB-001139 attivo

### Flusso di eventi

1. Apre **“I miei abbonamenti”**
2. Visualizza **SUB-001139** con azioni:
  - **Mostra QR**
  - **Scarica fattura**
  - **Gestisci rinnovo** (abilita/disabilita)

**Exit Condition:** Strumenti di gestione disponibili

### 13. UC 9.1 – Abilitazione rinnovo automatico

**Attore:** Utente

**Entry Condition:** SUB attivo con rinnovo disattivato

#### Flusso di eventi

1. Seleziona “**Gestisci rinnovo**” → **abilita**
2. Il sistema registra il consenso e mostra la **data di rinnovo previsto**

**Exit Condition:** Rinnovo automatico attivato

### 14. UC 10 – Accesso in struttura con tessera digitale

**Attore:** Utente

**Entry Condition:** Visita in struttura; SUB attivo

#### Flusso di eventi

1. Mostra il **QR** della tessera digitale
2. Reception scansiona
3. Il sistema valida **SUB-001139** (stato/validità/struttura)
4. Registra **log ingresso**
5. Nessun addebito extra (compreso nell’annuale)

**Exit Condition:** Accesso consentito; log registrato

#### 4. DELIVERABLES E DEADLINES

Formazione gruppi di progetto	2/10/2025
Start-up progetto	7/10/2025
Problem Statement	14/10/2025
Requisiti e casi d'uso	28/10/2025
Requirements Analysis Document	11/11/2025
System Design Document	25/11/2025
Specifica delle interfacce dei moduli del sottosistema da implementare	16/12/2025
Piano di test di sistema e specifica dei casi di test per il sottosistema da implementare	16/12/2025
Object Design Document	20/12/2025
Documenti di Esecuzione del Test	22/12/2025
Discussione progetto	20/01/2026