

「αode」



# Analisi dei requisiti

2025-03-10

Responsabile	Nicolò Bovo
Redattori	Romeo Calero
Verificatori	Elia Leonetti

## Registro delle modifiche

Vers.	Data	Descrizione	Autore	Verificatore
0.6.0	2025-06-08	Aggiornamento requisiti per stream processor	Alessandro Di Pasquale, Nicolò Bovo	Romeo Calero
0.5.0	2025-04-09	Stesura requisiti funzionali, non funzionali e di vincolo	Giovanni Battista Matteazzi	Manuel Cinnirella
0.4.0	2025-04-02	Proseguito sviluppo e aggiunta casi d'uso	Giovanni Battista Matteazzi	Manuel Cinnirella
0.3.0	2025-03-28	Proseguito sviluppo e aggiunta casi d'uso	Giovanni Battista Matteazzi	Manuel Cinnirella
0.2.0	2025-03-17	Aggiunta sezione casi d'uso e architettura	Romeo Calero	Elia Leonetti
0.1.0	2025-03-10	Bozza del documento	Romeo Calero	Elia Leonetti

## Indice

<b>I. Introduzione</b>	<b>6</b>
I - 1. Scopo del documento	6
I - 2. Scopo del prodotto	6
I - 3. Riferimenti	6
I - 3.1. Riferimenti tecnici	6
I - 3.2. Riferimenti implementativi	6
<b>II. Descrizione del prodotto</b>	<b>6</b>
II - 1. Obiettivi del prodotto	6
II - 1.1. Obiettivi primari:	7
II - 1.2. Obiettivi secondari:	7
II - 2. Funzionalità e vincoli del prodotto	7
II - 2.1. Funzionalità implementate:	7
II - 2.2. Vincoli tecnici:	8
II - 3. Utenti e loro caratteristiche	8
II - 3.1. Tipologie di utenti:	8
<b>III. Architettura del sistema</b>	<b>8</b>
III - 1. Componenti Core	8
III - 2. Flusso Dati	9
<b>IV. Casi d'uso</b>	<b>9</b>
IV - 1. Introduzione	9
IV - 2. Attori	9
IV - 3. Elenco dei casi d'uso	9
IV - 3.1. UC1 - Autenticazione utente alla dashboard	9
IV - 3.2. UC2 - Inserimento Username	10
IV - 3.3. UC3 - Inserimento Password	10
IV - 3.4. UC4 - Visualizzazione Errore Credenziali	10
IV - 3.5. UC5 - Visualizzazione Dashboard	11
IV - 3.6. UC6 - Visualizzazione profilo utente nella dashboard	12
IV - 3.7. UC7 - Visualizzazione statistiche utente nella dashboard	12
IV - 3.8. UC8 - Visualizzazione promozioni recenti nella dashboard	12
IV - 3.9. UC9 - Filtra punti di interesse per categoria	13
IV - 3.10. UC10 - Messaggio di errore «Nessun risultato trovato»	13
IV - 3.11. UC11 - Selezione del marker	14
IV - 3.12. UC12 - Visualizzazione nome del punto di interesse dal marker	14
IV - 3.13. UC13 - Visualizzazione categoria dal punto di interesse dal marker	15
IV - 3.14. UC14 - Inizio percorso con visualizzazione bici sulla mappa	15
IV - 3.15. UC15 - Tracciamento percorso con polyline	16
IV - 3.16. UC16 - Riaccentramento mappa sulla posizione bici	16
IV - 3.17. UC17 - Cancellazione storico percorso dalla mappa	16
IV - 3.18. UC18 - Cambio modalità tema dell'interfaccia	17
IV - 3.19. UC19 - Logout dell'utente dal sistema	17

IV - 3.20.	UC20 - Accesso al sistema di monitoraggio per l'amministratore: Grafana .....	18
IV - 3.21.	UC21 - Navigazione alla dashboard di analisi marketing .....	18
IV - 3.22.	UC22 - Navigazione alla dashboard di monitoraggio del sistema ....	19
IV - 3.23.	UC23 - Visualizzazione metriche eventi giornalieri .....	20
IV - 3.24.	UC24 - Visualizzazione conteggio negozi totali .....	20
IV - 3.25.	UC25 - Visualizzazione lista utenti registrati .....	20
IV - 3.26.	UC26 - Visualizzazione mappa visite per negozio ultime 24h .....	21
IV - 3.27.	UC27 - Visualizzazione mappa negozi selezionati .....	21
IV - 3.28.	UC28 - Visualizzazione grafico utilizzo CPU .....	22
IV - 3.29.	UC29 - Visualizzazione grafico utilizzo memoria .....	22
IV - 3.30.	UC30 - Visualizzazione grafico utilizzo disco .....	23
IV - 3.31.	UC31 - Visualizzazione grafico cache hit ratio .....	23
IV - 3.32.	UC32 - Visualizzazione top 10 negozi più visitati .....	24
<b>V.</b>	<b>Requisiti .....</b>	<b>24</b>
V - 1.	Requisiti funzionali obbligatori .....	24
V - 1.1.	RF1 - Autenticazione e autorizzazione .....	24
V - 1.2.	RF2 - Tracking posizione e generazione eventi .....	24
V - 1.3.	RF3 - Elaborazione stream e proximity detection .....	24
V - 1.4.	RF4 - Generazione messaggi personalizzati .....	24
V - 1.5.	RF5 - Dashboard utente interattiva .....	24
V - 1.6.	RF6 - Storage e persistenza dati .....	25
V - 1.7.	RF7 - Cache e ottimizzazione performance .....	25
V - 1.8.	RF8 - ETL automatico e data management .....	25
V - 2.	Requisiti non funzionali .....	25
V - 2.1.	RNF1 - Performance .....	25
V - 2.2.	RNF2 - Scalabilità .....	25
V - 2.3.	RNF3 - Affidabilità .....	25
V - 2.4.	RNF4 - Sicurezza .....	26
V - 2.5.	RNF5 - Osservabilità .....	26
V - 2.6.	RNF6 - Usabilità .....	26
V - 3.	Requisiti di vincolo .....	26
V - 3.1.	RV1 - Tecnologici .....	26
V - 3.2.	RV2 - Geografici .....	26
V - 3.3.	RV3 - Operativi .....	26
<b>VI.</b>	<b>Riepilogo requisiti .....</b>	<b>26</b>

**Lista delle figure**

**Lista delle tabelle**

## I. Introduzione

### I - 1. Scopo del documento

Il presente documento di Analisi dei Requisiti fornisce una descrizione dettagliata e completa del sistema **NearYou**, una piattaforma di notifiche personalizzate basate sulla posizione che utilizza intelligenza artificiale generativa (AI generativa) per creare messaggi contestuali e rilevanti per gli utenti.

Il documento costituisce la base per lo sviluppo del sistema, definendo chiaramente funzionalità, vincoli, casi d'uso e requisiti tecnici derivanti dall'analisi del codice sorgente e dell'architettura implementata.

### I - 2. Scopo del prodotto

**NearYou** è una piattaforma tecnologica che combina **tracking della posizione in tempo reale**, **database geospaziali**, **intelligenza artificiale generativa** e **analytics avanzate** per fornire notifiche personalizzate agli utenti quando si trovano in prossimità di negozi o punti di interesse.

Il sistema è progettato per:

- **Tracciare la posizione** degli utenti in tempo reale attraverso simulazione GPS
- **Generare messaggi personalizzati** utilizzando LLM (Large Language Models) quando l'utente è vicino a un negozio
- **Fornire una dashboard interattiva** con visualizzazione su mappa delle posizioni, percorsi e notifiche
- **Memorizzare e analizzare** tutti i dati per fornire insights attraverso dashboard analytics
- **Aggiornare automaticamente** il database dei negozi tramite ETL da OpenStreetMap

### I - 3. Riferimenti

#### I - 3.1. Riferimenti tecnici

- Docker Compose: `docker-compose.yml`
- Configurazione ambiente: `.env`
- Documentazione API: FastAPI autodocs ( `/docs` )
- Database schema: Script di inizializzazione ClickHouse e PostgreSQL

#### I - 3.2. Riferimenti implementativi

- Repository GitHub del progetto
- Dockerfile e script di deployment
- Configurazioni di monitoraggio (Grafana, Prometheus)

## II. Descrizione del prodotto

### II - 1. Obiettivi del prodotto

Il sistema NearYou ha come obiettivo principale la **creazione di un'esperienza utente personalizzata e contestuale** attraverso l'integrazione di diverse tecnologie avanzate:

## II - 1.1. Obiettivi primari:

1. **Personalizzazione intelligente:** Utilizzare profili utente (età, professione, interessi) per generare messaggi rilevanti tramite LLM
2. **Geolocalizzazione precisa:** Implementare tracking in tempo reale con calcolo di distanze attraverso PostGIS
3. **Analytics avanzate:** Fornire insights attraverso dashboard Grafana con metriche di business e sistema
4. **Automazione:** Mantenere aggiornato il database negozi attraverso ETL automatici da OpenStreetMap

## II - 1.2. Obiettivi secondari:

- **Caching intelligente:** Ridurre latenza e costi LLM attraverso cache Redis
- **Monitoraggio completo:** Garantire osservabilità del sistema con Prometheus, Loki e metriche custom
- **Facilità di deployment:** Fornire ambiente containerizzato completo con Docker Compose

## II - 2. Funzionalità e vincoli del prodotto

### II - 2.1. Funzionalità implementate:

#### Data Pipeline e Streaming:

- Producer Kafka che simula movimenti utenti su percorsi reali di Milano usando OSRM
- Integrazione con PostGIS per calcolo distanze e identificazione negozi vicini

#### Generazione Messaggi Intelligente:

- Servizio microservice FastAPI per generazione messaggi personalizzati
- Integrazione con LLM (supporto Groq, OpenAI, provider multipli)
- Sistema di cache Redis per ottimizzazione performance e riduzione costi
- Template personalizzabili basati su profilo utente e tipologia negozio

#### Dashboard Utente:

- Interfaccia web responsive con mappa Leaflet interattiva
- Visualizzazione real-time di posizione utente, percorsi e negozi
- Sistema di notifiche in-app per messaggi personalizzati
- Autenticazione JWT con gestione sessioni sicura
- WebSocket per aggiornamenti posizione in tempo reale

#### Storage e Persistenza:

- ClickHouse per analytics e storage eventi temporali ad alta performance
- PostgreSQL/PostGIS per gestione dati geospaziali dei negozi
- Schema ottimizzato per query geospaziali e analytics

#### ETL e Data Management:

- Apache Airflow per orchestrazione ETL automatici
- ETL adattivo che si auto-configura in base ai cambiamenti nei dati
- Integrazione con OpenStreetMap per aggiornamento negozi Milano
- Sistema di tracking cambiamenti con logging completo

**Monitoring e Observability:**

- Dashboard Grafana con pannelli personalizzati per metriche business e sistema
- Prometheus per raccolta metriche con esportatori per tutti i componenti
- Loki per gestione log centralizzata
- Metriche custom per tracking performance LLM, cache hit rate, eventi utente

**II - 2.2. Vincoli tecnici:**

- **Geografici:** Sistema focalizzato su Milano (coordinate definite in configurazione)
- **Performance:** Target <200ms per generazione messaggi (con cache)
- **Sicurezza:** Kafka con SSL/TLS, JWT per autenticazione, password protette
- **Disponibilità:** Health checks e restart automatici per tutti i servizi

**II - 3. Utenti e loro caratteristiche****II - 3.1. Tipologie di utenti:****Utente finale:**

- **Descrizione:** Persona che utilizza l'applicazione per ricevere notifiche personalizzate
- **Caratteristiche:** Ha un profilo definito (età, professione, interessi) memorizzato nel sistema
- **Interazioni:** Accede alla dashboard, visualizza la mappa, riceve notifiche, naviga lo storico
- **Competenze richieste:** Utilizzo base di applicazioni web e mobile

**Amministratore di sistema:**

- **Descrizione:** Operatore responsabile del monitoraggio e manutenzione del sistema
- **Caratteristiche:** Competenze tecniche per interpretare metriche e log
- **Interazioni:** Accesso a Grafana, Prometheus, log Loki, configurazione Airflow
- **Competenze richieste:** Conoscenza sistemi distribuiti, database, monitoring

**Sviluppatore/DevOps:**

- **Descrizione:** Persona che sviluppa, deploy e mantiene il sistema
- **Caratteristiche:** Competenze tecniche avanzate su architetture microservizi
- **Interazioni:** Deploy con Docker Compose, configurazione environment, debug
- **Competenze richieste:** Docker, Kafka, database, cloud native technologies

**III. Architettura del sistema**

Il sistema NearYou implementa un'**architettura microservizi event-driven** con i seguenti componenti principali:

**III - 1. Componenti Core****Streaming Data Pipeline:**

- **Apache Kafka:** Message broker per streaming eventi GPS con SSL
- **Producer:** Simulatore movimenti utenti su percorsi OSRM Milano
- **Consumer:** Processore eventi con arricchimento dati e generazione messaggi

**API Services:**

- **Message Generator:** Microservizio FastAPI per generazione messaggi LLM



- **Dashboard Service:** Servizio FastAPI per dashboard utente con WebSocket
- **Authentication:** Sistema JWT per gestione autenticazione e autorizzazione

**Storage Layer:**

- **ClickHouse:** Database colonnare per analytics e storage eventi temporali
- **PostgreSQL/PostGIS:** Database relazionale con estensioni geospaziali per negozi
- **Redis:** Cache distribuita per messaggi LLM e sessioni

**ETL e Data Management:**

- **Apache Airflow:** Orchestratore per ETL automatici da OpenStreetMap
- **OSRM:** Servizio routing per calcolo percorsi realistici su Milano

**Monitoring Stack:**

- **Grafana:** Dashboard per visualizzazione metriche business e sistema
- **Prometheus:** Raccolta metriche con esportatori specializzati
- **Loki:** Gestione log centralizzata
- **Exporters:** Componenti per esposizione metriche database e servizi

## III - 2. Flusso Dati

1. **Generazione Eventi:** Producer simula movimenti utenti su percorsi OSRM
2. **Proximity Detection:** Calcolo distanze con PostGIS, trigger per messaggi
3. **Message Generation:** Chiamata servizio LLM con cache Redis
4. **Storage:** Persistenza eventi arricchiti in ClickHouse
5. **Real-time Updates:** Notifica utenti via WebSocket
6. **Analytics:** Visualizzazione dati in dashboard Grafana

## IV. Casi d'uso

### IV - 1. Introduzione

I casi d'uso del sistema NearYou sono strutturati attorno alle funzionalità core implementate nel codice sorgente. Ogni caso d'uso è identificato da un codice univoco UC[numero] e descrive le interazioni tra attori e sistema.

### IV - 2. Attori

**Utente Autenticato:** Persona che utilizza la dashboard per visualizzare la propria posizione e ricevere notifiche personalizzate, avendo eseguito l'accesso.

**Utente non Autenticato:** Utente che non ha ancora eseguito l'accesso.

**Amministratore:** Persona che usa la dashboard per visualizzare le analisi di marketing e di monitoraggio del sistema.

### IV - 3. Elenco dei casi d'uso

#### IV - 3.1. UC1 - Autenticazione utente alla dashboard

**Attore principale:**

- Utente non autenticato

**Precondizioni:**

- L'utente ha credenziali valide nel sistema

**Postcondizioni:**

- L'utente è autenticato e ha accesso alla dashboard

**Scenario principale:**

1. L'utente accede alla pagina di login
2. Inserisce username e password
3. Il sistema valida le credenziali e autentica l'utente consentendogli l'accesso alla dashboard

**Inclusioni:**

- UC2: Inserimento Username
- UC3: Inserimento Password

**Estensioni:**

- UC4: Visualizzazione Errore Credenziali

**IV - 3.2. UC2 - Inserimento Username**

**Attore principale:**

- Utente non autenticato

**Precondizioni:**

- L'utente ha uno username valido nel sistema
- L'utente sta effettuando l'autenticazione per la pagina di dashboard

**Postcondizioni:**

- Il sistema accetta lo username inserito

**Scenario principale:**

1. L'utente accede alla pagina di login
2. L'utente inserisce lo username

**IV - 3.3. UC3 - Inserimento Password**

**Attore principale:**

- Utente non autenticato

**Precondizioni:**

- L'utente ha una password valida nel sistema
- L'utente sta effettuando l'autenticazione per la pagina di dashboard

**Postcondizioni:**

- Il sistema accetta la password inserita

**Scenario principale:**

1. L'utente accede alla pagina di login
2. L'utente inserisce la password

**IV - 3.4. UC4 - Visualizzazione Errore Credenziali**

**Attore principale:**

- Utente non autenticato

**Precondizioni:**

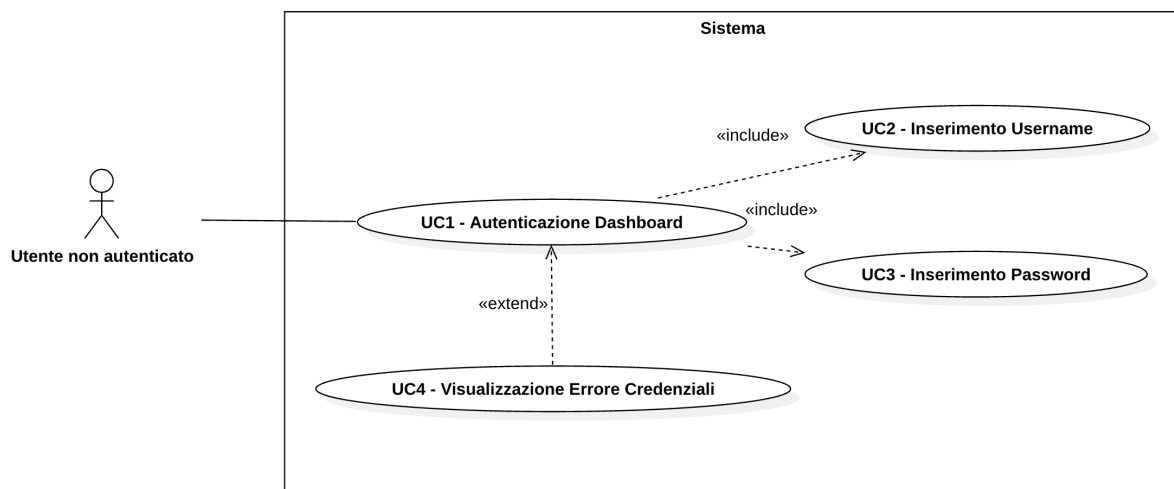
- L'utente sta effettuando l'autenticazione per la pagina di dashboard

**Postcondizioni:**

- Il sistema rifiuta le credenziali d'accesso inserite dall'utente poiché discrepanti con quelle memorizzate

**Scenario principale:**

1. L'utente accede alla pagina di login
2. L'utente inserisce lo username
3. L'utente inserisce la password
4. L'utente riceve un messaggio di errore che lo notifica delle credenziali errate

**IV - 3.5. UC5 - Visualizzazione Dashboard****Attore principale:**

- Utente Autenticato

**Precondizioni:**

- L'utente ha effettuato il login con successo

**Postcondizioni:**

- La dashboard principale è completamente caricata e funzionale
- Tutte le informazioni dell'utente sono visualizzate

**Scenario principale:**

1. L'utente accede alla dashboard
2. Il sistema carica la dashboard principale
3. Il sistema carica la mappa con i punti di interesse
4. Il sistema inizializza tutti i componenti della dashboard

**Inclusioni:**

- Visualizzazione profilo utente nella dashboard (UC6)
- Visualizzazione statistiche utente nella dashboard (UC7)
- Visualizzazione promozioni recenti nella dashboard (UC8)

**IV - 3.6. UC6 - Visualizzazione profilo utente nella dashboard****Attore principale:**

- Utente Autenticato

**Precondizioni:**

- L'utente è autenticato nel sistema e quindi visualizza la dashboard
- Il sistema mantiene le informazioni del profilo utente

**Postcondizioni:**

- Le informazioni del profilo utente sono visualizzate nella sidebar sinistra

**Scenario principale:**

1. L'utente accede alla dashboard principale
2. Il sistema carica e visualizza le informazioni del profilo nella sidebar (ID, Età, Professione, Interessi)

**IV - 3.7. UC7 - Visualizzazione statistiche utente nella dashboard****Attore principale:**

- Utente Autenticato

**Precondizioni:**

- L'utente è autenticato nel sistema e quindi visualizza la dashboard
- Il sistema traccia e memorizza le attività dell'utente

**Postcondizioni:**

- Le statistiche dell'utente sono visualizzate nella sezione dedicata della sidebar

**Scenario principale:**

1. L'utente accede alla dashboard principale
2. Il sistema calcola e mostra le statistiche aggiornate (km percorsi, negozi vicini, notifiche, minuti attivi)

**IV - 3.8. UC8 - Visualizzazione promozioni recenti nella dashboard****Attore principale:**

- Utente Autenticato

**Precondizioni:**

- L'utente è autenticato nel sistema e quindi visualizza la dashboard
- Sono disponibili promozioni nel sistema

**Postcondizioni:**

- Le promozioni recenti vengono visualizzate nella sezione dedicata della sidebar

**Scenario principale:**

1. L'utente accede alla dashboard principale
2. Il sistema recupera le promozioni più recenti disponibili
3. Il sistema mostra le promozioni nella sezione «Promozioni Recenti» della sidebar



#### IV - 3.9. UC9 - Filtra punti di interesse per categoria

##### Attore principale:

- Utente Autenticato

##### Precondizioni:

- L'utente è autenticato e si trova nella dashboard principale
- Il sistema contiene punti di interesse di diverse categorie
- La barra di filtri è visibile in basso alla mappa

##### Postcondizioni:

- Vengono visualizzati sulla mappa solo i punti di interesse della categoria selezionata

##### Scenario principale:

1. L'utente accede alla dashboard principale
2. L'utente seleziona una categoria specifica (Ristoranti, Abbigliamento, Supermercati, Elettronica, Bar & Caffè)

##### Inclusioni:

- Visualizzazione nome del punto di interesse dal marker (UC12)
- Visualizzazione categoria dal punto di interesse dal marker (UC13)

##### Estensioni:

- UC10: Visualizzazione messaggio «Nessun risultato trovato»

#### IV - 3.10. UC10 - Messaggio di errore «Nessun risultato trovato»

##### Attore principale:

- Utente Autenticato

##### Precondizioni:

- L'utente è autenticato e si trova nella dashboard principale
- L'utente ha applicato un filtro per categoria
- Il sistema non dispone di punti di interesse inerenti alla categoria selezionata

##### Postcondizioni:

- Il sistema non carica alcun punto di interesse
- Il sistema mostra il messaggio di errore

##### Scenario principale:

1. L'utente accede alla dashboard principale
2. L'utente seleziona una categoria specifica
3. L'utente visualizza il messaggio di errore



#### IV - 3.11. UC11 - Selezione del marker

##### Attore principale:

- Utente Autenticato

##### Precondizioni:

- L'utente è autenticato e si trova nella dashboard con la mappa visualizzata
- Il sistema tiene traccia dei marker
- Sono presenti marker dei punti di interesse sulla mappa

##### Postcondizioni:

- Il sistema mostra i dettagli del punto di interesse nel popup del marker.

##### Scenario principale:

1. L'utente visualizza la mappa con i marker dei punti di interesse
2. L'utente clicca su un marker specifico

##### Inclusioni:

- Visualizzazione nome del punto di interesse dal marker (UC12)
- Visualizzazione categoria del punto di interesse dal marker (UC13)

#### IV - 3.12. UC12 - Visualizzazione nome del punto di interesse dal marker

##### Attore principale:

- Utente Autenticato

##### Precondizioni:

- L'utente è autenticato e si trova nella dashboard principale
- Sono presenti marker dei punti di interesse sulla mappa
- L'utente ha selezionato o sta visualizzando un marker specifico

##### Postcondizioni:

- Il sistema visualizza il nome del punto di interesse del marker selezionato

**Scenario principale:**

1. L'utente accede alla dashboard principale
2. L'utente seleziona una categoria specifica
3. L'utente visualizza il nome del punto di interesse dal marker

#### IV - 3.13. UC13 - Visualizzazione categoria dal punto di interesse dal marker

**Attore principale:**

- Utente Autenticato

**Precondizioni:**

- L'utente è autenticato e si trova nella dashboard principale
- Sono presenti marker dei punti di interesse sulla mappa
- L'utente ha selezionato o sta visualizzando un marker specifico

**Postcondizioni:**

- Il sistema visualizza la categoria del punto di interesse del marker selezionato

**Scenario principale:**

1. L'utente accede alla dashboard principale
2. L'utente seleziona una categoria specifica
3. L'utente visualizza la categoria del punto di interesse dal marker



#### IV - 3.14. UC14 - Inizio percorso con visualizzazione bici sulla mappa

**Attore principale:**

- Utente Autenticato

**Precondizioni:**

- L'utente è autenticato e si trova nella dashboard con la mappa visualizzata
- Il GPS dell'utente è attivo e funzionante
- L'utente non ha un percorso già attivo

**Postcondizioni:**

- L'icona della bici viene visualizzata sulla mappa nella posizione corrente dell'utente
- Il tracciamento del percorso è attivato

**Scenario principale:**

1. L'utente avvia un nuovo percorso dalla dashboard (percorso generato)
2. Il sistema rileva la posizione GPS corrente dell'utente
3. Il sistema posiziona l'icona della bici sulla mappa nella posizione rilevata

**Inclusioni:**

- Tracciamento percorso con polyline (UC15)

**IV - 3.15. UC15 - Tracciamento percorso con polyline****Attore principale:**

- Utente Autenticato

**Precondizioni:**

- L'utente ha iniziato un percorso attivo
- Il GPS dell'utente è attivo e fornisce posizioni valide
- L'icona della bici è visualizzata sulla mappa

**Postcondizioni:**

- Una polyline blu traccia il percorso dell'utente sulla mappa

**Scenario principale:**

1. Il sistema rileva continuamente la posizione GPS dell'utente in movimento
2. Il sistema aggiorna la posizione dell'icona della bici
3. Il sistema disegna progressivamente una polyline blu collegando le posizioni precedenti

**IV - 3.16. UC16 - Riaccentramento mappa sulla posizione bici****Attore principale:**

- Utente Autenticato

**Precondizioni:**

- L'utente ha un percorso attivo
- L'icona della bici è visualizzata sulla mappa
- L'utente ha spostato manualmente la vista della mappa

**Postcondizioni:**

- La mappa viene centrata automaticamente sulla posizione corrente della bici

**Scenario principale:**

1. L'utente clicca sull'icona del mirino presente sui controlli della mappa
2. Il sistema identifica la posizione corrente della bici
3. Il sistema centra la vista della mappa sulla posizione della bici

**IV - 3.17. UC17 - Cancellazione storico percorso dalla mappa****Attore principale:**

- Utente Autenticato



**Precondizioni:**

- L'utente è autenticato e si trova nella dashboard con la mappa visualizzata
- È presente una polyline blu dello storico percorso sulla mappa

**Postcondizioni:**

- La polyline blu dello storico percorso viene rimossa completamente dalla mappa

**Scenario principale:**

1. L'utente clicca sull'icona del pennello presente sui controlli della mappa in alto a destra
2. Il sistema rimuove la polyline blu dello storico percorso
3. La mappa mostra solo la posizione corrente della bici senza traccia storica

**IV - 3.18. UC18 - Cambio modalità tema dell'interfaccia****Attore principale:**

- Utente Autenticato

**Precondizioni:**

- L'utente è autenticato e si trova nella dashboard con la mappa visualizzata
- L'interfaccia è attualmente in modalità chiara o scura

**Postcondizioni:**

- L'interfaccia cambia da tema chiaro a scuro o viceversa
- La preferenza del tema viene salvata e mantenuta per l'utente

**Scenario principale:**

1. L'utente clicca sull'icona del sole/luna presente in alto a destra
2. Il sistema alterna la modalità tema corrente
3. L'interfaccia viene aggiornata immediatamente con il nuovo tema

**IV - 3.19. UC19 - Logout dell'utente dal sistema****Attore principale:**

- Utente Autenticato

**Precondizioni:**

- L'utente è autenticato e si trova nella dashboard con la mappa visualizzata

**Postcondizioni:**

- L'utente viene disconnesso dal sistema
- L'utente viene reindirizzato alla pagina di login

**Scenario principale:**

1. L'utente clicca sul pulsante «Esci» presente in alto a destra nella dashboard
2. Il sistema termina la sessione dell'utente
3. Il sistema reindirizza l'utente alla pagina di autenticazione



#### IV - 3.20. UC20 - Accesso al sistema di monitoraggio per l'amministratore: Grafana

##### Attore principale:

- Amministratore

##### Precondizioni:

- L'amministratore ha accesso al link diretto del sistema Grafana
- Il sistema Grafana è configurato e disponibile

##### Postcondizioni:

- Il sistema Grafana è completamente caricato
- L'amministratore può navigare tra le dashboard disponibili

##### Scenario principale:

1. L'amministratore accede al link diretto del sistema Grafana
2. Il sistema carica l'interfaccia principale di Grafana
3. Il sistema Grafana carica automaticamente tutti i pannelli di analisi business
4. Il sistema mostra le dashboard disponibili (Dashboard Utente e Dashboard Sistema)

##### Inclusioni:

- Navigazione alla dashboard di analisi marketing (UC21)
- Navigazione alla dashboard di monitoraggio sistema (UC22)

#### IV - 3.21. UC21 - Navigazione alla dashboard di analisi marketing

##### Attore principale:

- Amministratore

##### Precondizioni:

- L'amministratore ha accesso al sistema Grafana

- La dashboard di analisi marketing è configurata

**Postcondizioni:**

- La dashboard di analisi marketing è caricata con tutti i suoi relativi pannelli

**Scenario principale:**

1. L'amministratore seleziona la dashboard di analisi marketing
2. Il sistema carica tutti i pannelli di business
3. Il sistema visualizza le metriche e dati di marketing

**Inclusioni:**

- Visualizzazione metriche eventi giornalieri (UC23)
- Visualizzazione conteggio negozi totali (UC24)
- Visualizzazione lista utenti registrati (UC25)
- Visualizzazione mappa visite per negozio ultime 24h (UC26)
- Visualizzazione mappa negozi selezionati (UC27)
- Visualizzazione top 10 negozi più visitati (UC32)

**IV - 3.22. UC22 - Navigazione alla dashboard di monitoraggio del sistema****Attore principale:**

- Amministratore

**Precondizioni:**

- L'amministratore ha accesso al sistema Grafana
- La dashboard di monitoraggio del sistema è configurata

**Postcondizioni:**

- La dashboard di monitoraggio del sistema è caricata con tutti i suoi relativi pannelli

**Scenario principale:**

1. L'amministratore seleziona la dashboard di monitoraggio del sistema
2. Il sistema carica tutti i pannelli di monitoraggio delle risorse
3. Il sistema inizializza i grafici e le metriche di sistema

**Inclusioni:**

- Visualizzazione grafico utilizzo CPU (UC28)
- Visualizzazione grafico utilizzo memoria (UC29)
- Visualizzazione grafico utilizzo disco (UC30)
- Visualizzazione grafico cache hit ratio (UC31)



#### IV - 3.23. UC23 - Visualizzazione metriche eventi giornalieri

##### Attore principale:

- Amministratore

##### Precondizioni:

- L'amministratore si trova nella dashboard di analisi marketing
- Il sistema traccia gli eventi utente

##### Postcondizioni:

- Il conteggio degli eventi del giorno corrente è visualizzato (numero)

##### Scenario principale:

1. L'amministratore visualizza il pannello «Eventi Oggi»
2. Il sistema calcola il numero di eventi del giorno corrente
3. Il sistema mostra il valore numerico nel pannello

#### IV - 3.24. UC24 - Visualizzazione conteggio negozi totali

##### Attore principale:

- Amministratore

##### Precondizioni:

- L'amministratore si trova nella dashboard di analisi marketing
- Il sistema conteggia il numero dei negozi registrati

##### Postcondizioni:

- Il numero totale di negozi nel sistema è visualizzato (numero)

##### Scenario principale:

1. L'amministratore visualizza il pannello «Negozi Totali»
2. Il sistema conta tutti i negozi registrati nel database
3. Il sistema mostra il conteggio totale

#### IV - 3.25. UC25 - Visualizzazione lista utenti registrati

##### Attore principale:

- Amministratore

**Precondizioni:**

- L'amministratore si trova nella dashboard di analisi marketing
- Esistono utenti registrati nel sistema

**Postcondizioni:**

- La tabella completa degli utenti con tutti i dettagli è visualizzata

**Scenario principale:**

1. L'amministratore visualizza la sezione «Lista Utenti»
2. Il sistema recupera tutti i dati degli utenti dal database
3. Il sistema mostra la tabella con tutte le colonne utente
4. Il sistema implementa la paginazione per navigare tra gli utenti

**IV - 3.26. UC26 - Visualizzazione mappa visite per negozio ultime 24h**

**Attore principale:**

- Amministratore

**Precondizioni:**

- L'amministratore si trova nella dashboard di analisi marketing
- Il sistema traccia le visite degli utenti ai negozi

**Postcondizioni:**

- Il grafico delle visite per negozio è visualizzato

**Scenario principale:**

1. L'amministratore visualizza il pannello «Visite per Negozio ultime 24h»
2. Il sistema analizza i dati delle visite alle varie attività commerciali
3. Il sistema mostra le visite per negozio

**IV - 3.27. UC27 - Visualizzazione mappa negozi selezionati**

**Attore principale:**

- Amministratore

**Precondizioni:**

- L'amministratore si trova nella dashboard di analisi marketing
- Esistono negozi con coordinate geografiche nel sistema

**Postcondizioni:**

- La mappa con i marker rossi dei negozi selezionati è visualizzata

**Scenario principale:**

1. L'amministratore visualizza il pannello «Mappa dei Negozi Selezionati»
2. Il sistema carica la mappa geografica della regione
3. Il sistema posiziona i marker rossi per ogni negozio selezionato



#### IV - 3.28. UC28 - Visualizzazione grafico utilizzo CPU

##### Attore principale:

- Amministratore

##### Precondizioni:

- L'amministratore si trova nella dashboard di monitoraggio sistema
- Il sistema di monitoraggio raccoglie metriche CPU

##### Postcondizioni:

- Il grafico temporale dell'utilizzo CPU è visualizzato con dati aggiornati

##### Scenario principale:

1. L'amministratore visualizza il pannello «CPU Utilizzo»
2. Il sistema recupera le metriche CPU dal node-exporter
3. Il sistema genera il grafico a linee temporali con percentuali di utilizzo

#### IV - 3.29. UC29 - Visualizzazione grafico utilizzo memoria

##### Attore principale:

- Amministratore

##### Precondizioni:

- L'amministratore si trova nella dashboard di monitoraggio sistema
- Il sistema di monitoraggio raccoglie metriche di memoria

##### Postcondizioni:

- Il grafico temporale dell'utilizzo memoria è visualizzato

##### Scenario principale:

1. L'amministratore visualizza il pannello «Memoria Utilizzo»
2. Il sistema recupera le metriche di memoria dal node-exporter
3. Il sistema genera il grafico a linee temporali con percentuali di utilizzo memoria

**IV - 3.30. UC30 - Visualizzazione grafico utilizzo disco****Attore principale:**

- Amministratore

**Precondizioni:**

- L'amministratore si trova nella dashboard di monitoraggio sistema
- Il sistema di monitoraggio raccoglie metriche disco

**Postcondizioni:**

- Il grafico temporale dell'utilizzo disco è visualizzato

**Scenario principale:**

1. L'amministratore visualizza il pannello «Disco Utilizzo»
2. Il sistema recupera le metriche di spazio disco dal node-exporter
3. Il sistema genera il grafico a linee temporali con percentuali di spazio utilizzato

**IV - 3.31. UC31 - Visualizzazione grafico cache hit ratio****Attore principale:**

- Amministratore

**Precondizioni:**

- L'amministratore si trova nella dashboard di monitoraggio sistema
- Il sistema di cache è attivo e tracciato

**Postcondizioni:**

- Il grafico circolare del cache hit ratio è visualizzato con colori indicativi

**Scenario principale:**

1. L'amministratore visualizza il pannello «Cache Hit Ratio»
2. Il sistema calcola il rapporto tra hit e miss della cache
3. Il sistema genera il grafico circolare colorato (verde/rosso in base alle performance)



#### **IV - 3.32. UC32 - Visualizzazione top 10 negozi più visitati**

**Attore principale:**

- Amministratore

**Precondizioni:**

- L'amministratore si trova nella dashboard di analisi marketing
- Il sistema traccia le visite ai negozi

**Postcondizioni:**

- La classifica dei 10 negozi più visitati è visualizzata

**Scenario principale:**

1. L'amministratore visualizza il pannello «Top 10 Negozi Più Visitati»
2. Il sistema analizza i dati delle visite degli ultimi 24 ore
3. Il sistema genera la classifica ordinata per numero di visite

### **V. Requisiti**

#### **V - 1. Requisiti funzionali obbligatori**

**V - 1.1. RF1 - Autenticazione e autorizzazione**

- **RF1.1:** Il sistema deve supportare autenticazione JWT con username/password
- **RF1.2:** I token devono avere scadenza configurabile (default 1 ora)
- **RF1.3:** Le password devono essere memorizzate in forma sicura nel database
- **RF1.4:** Il sistema deve supportare logout con invalidazione token

**V - 1.2. RF2 - Tracking posizione e generazione eventi**

- **RF2.1:** Il sistema deve simulare movimenti utenti su percorsi realistici Milano
- **RF2.2:** Gli eventi GPS devono essere generati ogni 2 secondi per utente attivo
- **RF2.3:** I percorsi devono essere calcolati usando OSRM con profilo bicicletta
- **RF2.4:** Gli eventi devono contenere: user\_id, latitudine, longitudine, timestamp

**V - 1.3. RF3 - Elaborazione stream e proximity detection**

- **RF3.1:** Il sistema deve processare eventi GPS in tempo reale via Kafka
- **RF3.2:** Deve calcolare distanza da negozi usando funzioni PostGIS
- **RF3.3:** Deve triggerare generazione messaggi per distanze < 200 metri
- **RF3.4:** Deve evitare messaggi duplicati per stesso negozio durante singolo noleggio

**V - 1.4. RF4 - Generazione messaggi personalizzati**

- **RF4.1:** Il sistema deve generare messaggi usando LLM basati su profilo utente
- **RF4.2:** I messaggi devono essere personalizzati per: età, professione, interessi
- **RF4.3:** Deve supportare multiple provider LLM (OpenAI, Groq)
- **RF4.4:** I messaggi devono essere limitati a massimo 30 parole
- **RF4.5:** Deve fornire messaggi fallback in caso di errore LLM

**V - 1.5. RF5 - Dashboard utente interattiva**

- **RF5.1:** Deve visualizzare mappa interattiva con Leaflet
- **RF5.2:** Deve mostrare posizione utente real-time con marker personalizzato



- **RF5.3:** Deve disegnare percorso storico come polyline
- **RF5.4:** Deve visualizzare negozi nell'area visibile con marker categorizzati
- **RF5.5:** Deve supportare filtri per categoria negozio
- **RF5.6:** Deve mostrare notifiche in sidebar con lazy loading

#### **V - 1.6. RF6 - Storage e persistenza dati**

- **RF6.1:** Deve memorizzare eventi utente in ClickHouse per analytics
- **RF6.2:** Deve gestire negozi in PostgreSQL/PostGIS per query geospaziali
- **RF6.3:** Deve mantenere profili utenti con dati demografici
- **RF6.4:** Deve tracciare storico completo movimenti e notifiche

#### **V - 1.7. RF7 - Cache e ottimizzazione performance**

- **RF7.1:** Deve implementare cache Redis per messaggi LLM
- **RF7.2:** Le chiavi cache devono essere generate con fuzzy matching
- **RF7.3:** Deve supportare TTL adattivo basato su popolarità categoria
- **RF7.4:** Deve fornire statistiche cache (hit rate, misses)

#### **V - 1.8. RF8 - ETL automatico e data management**

- **RF8.1:** Deve aggiornare database negozi da OpenStreetMap
- **RF8.2:** ETL deve essere adattivo basandosi su:
  - Frequenza cambiamenti storici
  - Periodi speciali (festività, saldi)
  - Anomalie nei pattern di modifica
- **RF8.3:** Deve registrare audit trail di tutti i cambiamenti
- **RF8.4:** Deve supportare gestione conflitti nell'aggiornamento dati

### **V - 2. Requisiti non funzionali**

#### **V - 2.1. RNF1 - Performance**

- **RNF1.1:** Latenza generazione messaggi < 200ms (con cache)
- **RNF1.2:** Latenza aggiornamenti posizione < 100ms via WebSocket
- **RNF1.3:** Throughput Kafka > 1000 eventi/secondo
- **RNF1.4:** Query geospaziali PostGIS < 50ms per calcolo distanze

#### **V - 2.2. RNF2 - Scalabilità**

- **RNF2.1:** Supporto elaborazione multi-worker con Bytewax
- **RNF2.2:** Partizionamento Kafka per distribuzione carico
- **RNF2.3:** Connection pooling per database
- **RNF2.4:** Cache distribuita Redis per multiple istanze

#### **V - 2.3. RNF3 - Affidabilità**

- **RNF3.1:** Health checks automatici per tutti i servizi
- **RNF3.2:** Restart automatico in caso di failure
- **RNF3.3:** Gestione graceful disconnessioni WebSocket
- **RNF3.4:** Retry logic per chiamate LLM fallite

**V - 2.4. RNF4 - Sicurezza**

- **RNF4.1:** Comunicazione Kafka con SSL/TLS
- **RNF4.2:** Autenticazione JWT con secret sicuro
- **RNF4.3:** Validazione input per prevenire injection
- **RNF4.4:** Password database protette in variabili ambiente

**V - 2.5. RNF5 - Osservabilità**

- **RNF5.1:** Metriche Prometheus per tutti i componenti
- **RNF5.2:** Log strutturati con livelli configurabili
- **RNF5.3:** Dashboard Grafana per monitoring business e sistema
- **RNF5.4:** Alert automatici per condizioni anomale

**V - 2.6. RNF6 - Usabilità**

- **RNF6.1:** Interfaccia responsive per desktop e mobile
- **RNF6.2:** Tema scuro/chiaro configurabile dall'utente
- **RNF6.3:** Caricamento lazy per performance ottimali
- **RNF6.4:** Feedback visuale per azioni utente

**V - 3. Requisiti di vincolo****V - 3.1. RV1 - Tecnologici**

- **RV1.1:** Deployment tramite Docker Compose
- **RV1.2:** Architettura microservizi event-driven
- **RV1.3:** Streaming data con Apache Kafka
- **RV1.4:** Frontend web-based (no app mobile native)

**V - 3.2. RV2 - Geografici**

- **RV2.1:** Sistema focalizzato su Milano (bbox configurata)
- **RV2.2:** Coordinate sistema: WGS84 (EPSG:4326)
- **RV2.3:** Percorsi limitati a rete stradale esistente (OSRM)

**V - 3.3. RV3 - Operativi**

- **RV3.1:** Ambiente development single-machine
- **RV3.2:** Configurazione tramite variabili ambiente
- **RV3.3:** Logging su stdout per container orchestration
- **RV3.4:** Gestione segreti tramite Docker secrets o env vars

**VI. Riepilogo requisiti**

Categoria	Obbligatorie	Desiderabili	Facoltative	Totale
Funzionali	35	0	0	35
Non Funzionali	24	0	0	24
Vincolo	11	0	0	11
<b>Totale</b>	<b>70</b>	<b>0</b>	<b>0</b>	<b>70</b>