



# Specifica Tecnica

NearYou  
Smart custom advertising platform

[sevenbits.swe.unipd@gmail.com](mailto:sevenbits.swe.unipd@gmail.com)



**Registro modifiche**

Versione	Data	Autore	Verificatore	Descrizione
0.1.0	2025-02-26	Leonardo Trolese	Manuel Gusella	Inizio redazione del documento

# Indice

<b>1</b>	<b>Introduzione</b>	<b>5</b>
1.1	Scopo del documento . . . . .	5
1.2	Glossario . . . . .	5
1.3	Riferimenti . . . . .	5
1.3.1	Riferimenti normativi . . . . .	5
1.3.2	Riferimenti informativi . . . . .	5
<b>2</b>	<b>Tecnologie</b>	<b>6</b>
2.1	Linguaggi . . . . .	6
2.1.1	Python . . . . .	6
2.1.1.1	Versione . . . . .	6
2.1.1.2	Documentazione . . . . .	6
2.1.1.3	Utilizzo operato nel progetto . . . . .	6
2.1.1.4	Librerie e framework . . . . .	6
2.2	Strumenti e servizi . . . . .	7
2.2.1	Docker . . . . .	7
<b>3</b>	<b>API</b>	<b>7</b>
<b>4</b>	<b>Architettura di sistema</b>	<b>7</b>
<b>5</b>	<b>Architettura di deployment</b>	<b>7</b>
<b>6</b>	<b>Stato dei requisiti funzionali</b>	<b>7</b>

## Elenco delle figure

## Elenco delle tabelle

# 1 Introduzione

## 1.1 Scopo del documento

Il presente documento si propone come una risorsa completa per la comprensione degli aspetti tecnici e progettuali della piattaforma "NearYou", dedicata alla creazione di soluzioni di advertising personalizzato tramite intelligenza artificiale. L'obiettivo principale è fornire una descrizione dettagliata dell'architettura implementativa e di deployment, illustrando le tecnologie adottate e le motivazioni alla base delle scelte progettuali.

Nel contesto dell'architettura implementativa, il documento analizza nel dettaglio i moduli principali del sistema, i design pattern utilizzati. Saranno inclusi diagrammi delle classi, e una spiegazione dettagliata dei design pattern utilizzati e delle motivazioni di queste scelte.

Gli obiettivi di questo documento sono: motivare le decisioni architetturali, fungere da guida per lo sviluppo della piattaforma, e garantire la piena tracciabilità e copertura dei requisiti definiti nel documento di *Analisi dei Requisiti v1.0.0*.

In sintesi, il documento intende essere un punto di riferimento essenziale per tutti gli attori coinvolti nel ciclo di vita del progetto, offrendo una visione chiara e strutturata delle fondamenta tecniche che sorreggono NearYou e delle logiche che ne determinano il funzionamento.

## 1.2 Glossario

Con l'intento di evitare ambiguità interpretative del linguaggio utilizzato, viene fornito un Glossario che si occupa di esplicitare il significato dei termini che riguardano il contesto del Progetto<sub>G</sub>. I termini presenti nel glossario sono contrassegnati con una *G* a pedice : Termine<sub>G</sub>.

I termini composti, oltre alla *G* a pedice, saranno uniti da un "-" come segue: termine-composto<sub>G</sub>.

Le definizioni sono presenti nell'apposito documento *Glossario v1.0.0.pdf*.

## 1.3 Riferimenti

### 1.3.1 Riferimenti normativi

- Regolamento del Progetto<sub>G</sub> didattico  
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/PD1.pdf>  
(Consultato: 2025-02-10).
- Capitolato<sub>G</sub> C4 - NearYou - Smart custom advertising platform  
<https://www.math.unipd.it/~tullio/IS-1/2024/Progetto/C4p.pdf>  
(Consultato: 2025-02-10).
- *Norme di Progetto v1.0.0*

### 1.3.2 Riferimenti informativi

- *Glossario v1.0.0*
- *Analisi dei Requisiti v1.0.0*
- Analisi-dei-Requisiti<sub>G</sub> - SWE 2024-25  
<https://www.math.unipd.it/~tullio/IS-1/2024/Dispense/T05.pdf>  
(Consultato: 2025-02-10).
- Dependency Injection - SWE 2024-25
- Design Pattern Creazionali - SWE 2024-25
- Design Pattern Strutturali - SWE 2024-25
- Software Architecture Patterns - SWE 2024-25

- Verbali Interni
- Verbali Esterni

## 2 Tecnologie

Questa sezione descrive strumenti e tecnologie impiegate nello sviluppo del software del progetto nearYou. Descriveremo quindi a seguire le tecnologie usate per lo sviluppo del software, le librerie e i framework utilizzati, e le motivazioni di queste scelte.

### 2.1 Linguaggi

#### 2.1.1 Python

Trattasi di un linguaggio di programmazione ad alto livello, interpretato e orientato agli oggetti, noto per la sua sintassi chiara e leggibile. Python dispone anche di una vasta libreria standard ed è caratterizzato dalla grande quantità di framework disponibili.

##### 2.1.1.1 Versione

[DA INSERIRE]

##### 2.1.1.2 Documentazione

La documentazione può essere trovata qui: <https://docs.python.org/> (consultato: 2025-03-02)

##### 2.1.1.3 Utilizzo operato nel progetto

- Creazione dei sensori e simulazione degli spostamenti;
- Generazione dei punti di interesse oggetto del messaggio pubblicitario;
- Interazione con il database per la permanenza dei dati;
- Interazione con l'LLM mediante API;
- Logica di selezione dei punti di interesse rilevanti per l'utente;
- Testing.

##### 2.1.1.4 Librerie e framework

- Clickhouse Connect
  - . Versione:
  - . Documentazione: <https://clickhouse.com/docs/integrations/python> (consultato: 2025-03-02)
  - . Descrizione: ClickHouse Connect è una libreria Python che permette di connettersi a database ClickHouse, eseguire query SQL in modo veloce ed efficiente.
- Pyflink
  - . Versione:
  - . Documentazione: [https://pyflink.readthedocs.io/en/main/getting\\_started/index.html](https://pyflink.readthedocs.io/en/main/getting_started/index.html) (consultato: 2025-03-02)
  - . Descrizione: PyFlink è l'API Python di Apache Flink, che permette di scrivere e gestire applicazioni per l'elaborazione di flussi e batch di dati distribuiti in tempo reale, sfruttando la potenza e la scalabilità di Flink direttamente con Python.
- Langchain

- . Versione:
- . Documentazione: <https://python.langchain.com/docs/introduction/> (consultato: 2025-03-02)
- . Descrizione: LangChain è una libreria Python che semplifica la creazione di applicazioni basate su modelli linguistici, consentendo di orchestrare prompt, gestire la memoria della conversazione e integrare fonti di dati esterne.
- Groq
  - . Versione:
  - . Documentazione: <https://console.groq.com/docs/overview> (consultato: 2025-03-02)
  - . Descrizione: Groq è una libreria Python che permette di interagire con modelli di linguaggio tramite API, facilitando la generazione di testo, risposte conversazionali e completamenti.
- Confluent Kafka
  - . Versione: 2.8.0
  - . Documentazione: <https://docs.confluent.io/kafka/overview.html> (consultato: 2025-03-02)
  - . Descrizione: Confluent Kafka è una libreria Python che semplifica l'interazione con Apache Kafka, permettendo di produrre, consumare e gestire stream di dati in tempo reale, sfruttando la piattaforma Confluent.
- Geopy
  - . Versione:
  - . Documentazione: <https://geopy.readthedocs.io/en/stable/index.html> (consultato: 2025-03-02)
  - . Descrizione: Geopy è una libreria Python che facilita la geocodifica, il calcolo delle distanze e l'interazione con servizi di localizzazione.
- OSMnx
  - . Versione:
  - . Documentazione: <https://osmnx.readthedocs.io/en/stable/> (consultato: 2025-03-02)
  - . Descrizione: OSMnx è una libreria Python per scaricare, analizzare e visualizzare reti stradali e dati geografici da OpenStreetMap, utile per creare grafi, calcolare percorsi e studiare l'urbanistica.
- Faker
  - . Versione:
  - . Documentazione: <https://faker.readthedocs.io/en/master/> (consultato: 2025-03-02)
  - . Descrizione: Faker è una libreria Python che genera dati falsi realistici, come nomi, indirizzi, email o testi casuali, utile per test, mock di database o prototipazione.

## 2.2 Strumenti e servizi

### 2.2.1 Docker

Per lo sviluppo, il testing e il rilascio dell'applicativo sono stati usati dei container Docker, come suggerito dalla proponente, al fine di garantire ambienti di sviluppo e di test isolati, consistenti e riproducibili.

## 3 API

## 4 Architettura di sistema

## 5 Architettura di deployment

## 6 Stato dei requisiti funzionali