



# Code7Crusaders

Software Development Team

Analisi per la scelta delle tecnologie backend

## **Membri del Team:**

Enrico Cotti Cottini, Gabriele Di Pietro, Tommaso Diviesti  
Francesco Lapenna, Matthew Pan, Eddy Pinarello, Filippo Rizzolo

**Data:** 12 Febbraio 2025

# Indice

<b>1</b>	<b>Obiettivo</b>	<b>2</b>
<b>2</b>	<b>Analisi tecnologie backend scelte</b>	<b>2</b>
2.1	Flask . . . . .	2
2.2	Langchain . . . . .	2
<b>3</b>	<b>Possibili alternative da valutare</b>	<b>2</b>
3.1	FastAPI . . . . .	2
3.2	Django REST Framework . . . . .	2
3.3	LlamaIndex . . . . .	2
3.4	Haystack . . . . .	2
<b>4</b>	<b>Confronto</b>	<b>2</b>
<b>5</b>	<b>Conclusioni</b>	<b>2</b>

# 1 Obiettivo

Questo documento si pone l'obiettivo di confrontare le diverse tecnologie backend in modo da avere un'idea precisa e prendere una decisione sicura per la scelta di esse e per la loro integrazione all'interno del nostro progetto. Il confronto considera aspetti tecnici, caratteristiche, vantaggi e svantaggi dei vari framework discussi. Le nostre scelte finali sono l'utilizzo di Flask e di Langchain per diverse motivazioni chiarite nei paragrafi successivi.

## 2 Analisi tecnologie backend scelte

### 2.1 Flask

### 2.2 Langchain

## 3 Possibili alternative da valutare

### 3.1 FastAPI

### 3.2 Django REST Framework

### 3.3 LlamaIndex

### 3.4 Haystack

## 4 Confronto

## 5 Conclusioni

**Flask** è stato scelto per la sua leggerezza e semplicità nel creare API RESTful. Essendo un micro-framework, permette di sviluppare rapidamente un backend senza imporre dei vincoli rigidi. Inoltre, la sua ampia documentazione e la sua flessibilità lo rendono ideale per prototipi e progetti in evoluzione. Per quanto riguarda **Langchain**, invece, si tratta di una libreria progettata per facilitare l'integrazione dei *Large Language Models (LLMs)* nei sistemi Software. Esso permette di gestire conversazioni, memoria contestuale e connettori a database vettoriali, rendendo l'interazione con i modelli più strutturata e personalizzabile.