

# CHATBOT con MVC

Il progetto che abbiamo intenzione di perseguire consiste nello sviluppo di un'interfaccia strutturata come un'applicazione di messaggistica, che include la stessa intelligenza artificiale **Chat GPT** e che permetta all'utente di interfacciarsi e di effettuare una qualsiasi domanda per poi restituire il risultato più corretto possibile che possa aiutare l'utente.

## REQUISITI

L'applicazione sarà dotata di una interfaccia console e di un'interfaccia web, le quali si baseranno sul pattern **MVC** (Model View Controller), di conseguenza il progetto sarà suddiviso in tre principali parti:

La struttura del pattern architetturale Model-View-Controller è composta da tre parti principali:

- **Model:** il Model rappresenta i dati dell'applicazione. Essa è la parte dell'applicazione che gestisce la memorizzazione, l'accesso e l'aggiornamento dei dati. Il Model può essere costituito da una o più classi che rappresentano i dati e le operazioni sui dati.
- **View:** la View è la parte dell'applicazione che si occupa dell'interfaccia utente e della presentazione dei dati. Essa visualizza i dati contenuti nel Model all'utente, e fornisce un'interfaccia per l'input degli utenti. La View può essere costituita da una o più classi che rappresentano le diverse parti dell'interfaccia utente, come finestre, pulsanti, etichette, ecc.
- **Controller:** il Controller è la parte dell'applicazione che gestisce le interazioni dell'utente con l'applicazione, e si occupa di gestire le comunicazioni tra la View e il Model. Esso riceve gli input dell'utente dalla View e interagisce con il Model per recuperare e aggiornare i dati. Il Controller può essere costituito da una o più classi che gestiscono le diverse interazioni dell'utente con l'applicazione.

Il funzionamento del pattern architetturale MVC prevede che la View e il Controller lavorino insieme per fornire all'utente un'interfaccia utente per interagire con il Model. Quando l'utente esegue un'azione sull'interfaccia utente, il Controller riceve l'input dell'utente dalla View e gestisce l'interazione con il Model. Il Model può quindi eseguire le operazioni necessarie sui dati, e restituire i risultati al Controller. Il Controller a sua volta aggiorna la View con i dati restituiti dal Model, in modo che l'utente possa visualizzarli e interagire con essi.

Il linguaggio che abbiamo intenzione di utilizzare è **PHP**, in quanto il linguaggio è noto per la sua semplice integrabilità con l'**HTML** e per lo sviluppo di un'interfaccia grafica più visibilmente comprensibile e decorosa abbiamo implementato un framework nominato **Bootstrap**.

Per lo sviluppo di questo progetto servirà l'utilizzo di Apache cioè un web server open-source che consente agli utenti di ospitare e mantenere i propri siti su internet, dove dovrà essere installato PHP, versione 7.4, inoltre utilizzeremo le API sviluppate da OpenAI per l'implemento dell'intelligenza artificiale.

Le API saranno integrate tramite l'utilizzo di PHP, **JQuery** (tramite **Ajax**) e **Python** (per la comunicazione tramite cli).

## **VANTAGGI E SVANTAGGI DEL PATTERN**

Di seguito sono elencati i vantaggi e gli svantaggi del pattern architetturale MVC:

### **VANTAGGI**

- **Separazione dei compiti:** MVC separa in modo efficace la logica del software dal codice dell'interfaccia utente, il che rende più facile la manutenzione dell'applicazione e l'aggiornamento di parti specifiche del codice senza influire sul resto dell'applicazione.
- **Riusabilità del codice:** poiché il Model e il Controller sono separati dalla View, è possibile riutilizzare il codice del Model e del Controller per creare diverse interfacce utente.
- **Testabilità:** la separazione dei compiti e la modularità del codice rendono più facile testare l'applicazione in modo unitario, facilitando la correzione degli errori e il mantenimento del codice.
- **Scalabilità:** poiché l'applicazione è suddivisa in parti separate e indipendenti, è più facile gestire e scalare l'applicazione.

## SVANTAGGI

- **Complessità:** la suddivisione dell'applicazione in parti separate può portare a una maggiore complessità nel codice, poiché è necessario gestire le comunicazioni tra le diverse parti dell'applicazione.
- **Sovraccarico di codice:** l'utilizzo del pattern MVC può portare a un aumento della quantità di codice necessaria per implementare l'applicazione.
- **Inefficienza:** l'utilizzo del pattern MVC può portare a una riduzione delle prestazioni dell'applicazione, poiché la separazione dei compiti può richiedere una maggiore quantità di elaborazione rispetto a un'architettura più semplice.
- **Difficoltà di apprendimento:** il pattern MVC richiede una conoscenza avanzata dell'architettura dell'applicazione e delle interazioni tra le diverse parti dell'applicazione, il che può rendere difficile l'apprendimento per i nuovi sviluppatori.

## STRUTTURA

Il progetto è così strutturato, la cartella principale del progetto, chiamata "**ChatBot**", è inserita dentro **htdocs**. Al suo interno sarà presente la cartella "**app**" che conterrà le cartelle **Model**, **View** e **Controller**. Queste sottocartelle rappresentano i tre componenti principali del pattern MVC.

All'interno della cartella "**controller**" saranno presenti i file che rappresentano i controller della nostra applicazione. I controller sono responsabili di gestire le richieste degli utenti e di coordinare la logica dell'applicazione.

All'interno della cartella "**view**", avremo diverse sottocartelle che rappresentano le diverse viste dell'applicazione. Ad esempio, avremo una sottocartella chiamata "**web**" e una chiamata "**cli**". All'interno di queste sottocartelle, avremo file come "**index.php**" che rappresentano la vista principale di quella sezione dell'applicazione. La cartella "**template**" conterrà file comuni a tutte le viste, come ad esempio "**header.php**" e "**footer.php**". Questi file saranno inclusi in ogni pagina della nostra applicazione.

Nella cartella "**model**", avremo diversi file che rappresentano i modelli dell'applicazione.

La cartella "**public**" conterrà tutti i file accessibili al pubblico. Questo include file **CSS**, file **JavaScript** e il file "**index.php**" che rappresenta la pagina principale dell'applicazione.

La cartella "**vendor**" conterrà le **librerie di terze parti** che utilizziamo nella nostra applicazione, come ad esempio il framework **Open-AI** e **CensorWord**. Infine, avremo anche il file ".htaccess" che contiene le regole per configurare il server web.

## **DESCRIZIONE**

Il progetto utilizza le stesse API utilizzate da Chat GPT, in modo da creare un'applicazione che offre ai suoi utenti la possibilità di chattare con un sistema automatizzato di risposta che utilizza tecnologia di intelligenza artificiale e machine learning.

La principale funzione della applicazione è quella di fornire un **servizio di assistenza** che possa rispondere alle domande degli utenti con un alto livello di precisione e comprensione del linguaggio umano. La soluzione è basata sulle API fornite da OpenAI, che permettono di integrare il software di intelligenza artificiale all'interno della applicazione.

Inoltre, l'applicazione deve avere la capacità di analizzare i messaggi degli utenti e capire le loro intenzioni, in modo da fornire risposte pertinenti e utili.

Dovrebbe anche permettere di personalizzare i profili degli utenti, in modo che gli utenti possano accedere alle conversazioni precedenti e fare riferimento ai loro dati personali per ottenere una migliore assistenza.

L'applicazione dovrebbe anche offrire la possibilità di utilizzare le API di pagamento di OpenAI, in modo che gli utenti possano acquistare una maggiore quantità di credito o di funzioni avanzate. Ciò consentirà agli utenti professionisti di utilizzare l'applicazione per scopi commerciali, come l'automazione di una ChatBot aziendale, ottenendo un AI sempre aggiornato e performante.

Inoltre, l'applicazione dovrebbe fornire funzionalità avanzate di gestione dei dati, come l'archiviazione e la protezione dei messaggi degli utenti e delle preferenze personali. Dovrebbe anche essere in grado di gestire un grande numero di utenti contemporaneamente e garantire la stabilità e la sicurezza dell'applicazione.

Infine, l'applicazione dovrebbe avere un'interfaccia utente intuitiva e facile da usare. Dovrebbe essere accessibile da qualsiasi dispositivo, come PC, smartphone o tablet, e avere funzionalità avanzate come la ricerca di messaggi e la possibilità di modificare le impostazioni dell'account.

In sintesi, il progetto scolastico informatico che riproduce Chat GPT con le sue API è un'applicazione di assistenza virtuale avanzata che utilizza tecnologia di intelligenza artificiale per fornire supporto di alta qualità agli utenti. L'interfaccia utente intuitiva e le funzionalità avanzate di gestione dei dati rendono l'applicazione performante e intuitiva. L'utilizzo delle API amplia ulteriormente le potenzialità della soluzione, rendendo la Chat AI sempre aggiornata e performante.

Il progetto sarà raggiungibile attraverso il nome di dominio: [www.chatbot.it](http://www.chatbot.it).  
Una volta effettuato l'accesso all'utente sarà richiesto di accettare **l'informativa del GDPR**, e ciò sarà ovviamente obbligatorio per l'utilizzo dell'applicazione, la quale conterrà un'unica pagina dove l'utente potrà interagire con l'intelligenza artificiale.

Inoltre il nostro ChatBot sarà in grado di rilevare le parole non appropriate grazie all'inserimento di un apposito **filtro** che segnalerà quando la domanda contiene un termine volgare.

## **INFORMATIVA GDPR**

Il **GDPR (General Data Protection Regulation)** è un regolamento dell'Unione Europea che riguarda la protezione dei dati personali dei cittadini europei. È stato adottato il 27 aprile 2016 e in vigore dal 25 maggio 2018.

L'obiettivo del GDPR è di proteggere la privacy dei cittadini europei stabilendo regole chiare per il trattamento dei dati personali da parte di aziende e organizzazioni. Il regolamento si applica a tutte le aziende e le organizzazioni che raccolgono, elaborano o conservano dati personali di cittadini europei, indipendentemente dal fatto che siano o meno residenti dell'Unione Europea.

Il GDPR stabilisce una serie di diritti dei cittadini europei in relazione ai loro dati personali, tra cui il diritto di accedere ai propri dati, di chiederne la cancellazione, di chiederne la portabilità, di opporsi al loro trattamento e di essere informati in modo chiaro e trasparente sulle modalità di trattamento dei dati.

Inoltre, il regolamento impone alle aziende e alle organizzazioni di adottare misure tecniche e organizzative adeguate per garantire la sicurezza dei dati personali, prevenire la perdita, l'accesso non autorizzato, la divulgazione, la modifica o la distruzione dei dati personali. Le aziende e le organizzazioni devono inoltre notificare tempestivamente le violazioni dei dati personali alle autorità competenti e ai cittadini interessati.

Il GDPR prevede sanzioni significative per le aziende e le organizzazioni che non rispettano le norme in materia di protezione dei dati personali, tra cui multe fino al 4% del fatturato annuo globale dell'azienda o 20 milioni di euro, a seconda di quale cifra è più elevata.

# MANUALE DI ISTRUZIONI

Questa ChatBot è stata progettata per essere semplice ed efficiente, senza richiedere alcuna registrazione o gestione degli utenti.

Di seguito troverete una serie di istruzioni su come utilizzare la ChatBot.

## **Accedere alla Chat**

Per utilizzare la Chat, è necessario accedere ad essa. Puoi farlo visitando il sito web o l'interfaccia console che ospita il ChatBot.

## **Iniziare la conversazione**

Poi si può inserire la propria domanda all'interno dell'apposita textbox, premendo poi il tasto INVIO per inviarla.

## **Fare domande**

La ChatBot è stata progettata per rispondere alle domande degli utenti. Quando fai una domanda, la ChatBot cercherà di fornirti una risposta. Assicurati di formulare le domande in modo chiaro e preciso.

## **Ricevere risposte**

La ChatBot fornirà le risposte alle tue domande nella chat. Cerca di leggere attentamente le risposte per capire se sono utili e soddisfacenti. In caso contrario, ripeti la domanda o prova a chiedere ulteriori informazioni alla ChatBot.

## **Limitazioni**

La ChatBot senza gestione utente può rispondere solo alle domande che sono state preimpostate. Non è in grado di fornire informazioni personalizzate o rispondere a domande che richiedono un'analisi più approfondita. Inoltre, ricorda che la ChatBot è in grado di rispondere solo a domande sull'argomento per cui è stata progettata.

## **Scorciatoie da tastiera**

- Tasto **'Enter'** per inviare la domanda;
- Combinazione **'Shift + Enter'** per andare a capo;
- Combinazione **'Ctrl + K'** per pulire la chat.

## IMPLEMENTAZIONI FUTURE

Per migliorare l'esperienza con il nostro ChatBot si potrebbe implementare la gestione dell'utenza tramite un database **MySQL**, specificando email e password e crittografando i dati dell'utente tramite hash **sha-256**.

Grazie a questa implementazione, l'utente avrà a disposizione lo storico delle proprie domande e delle risposte del bot.

Inoltre si potrebbe implementare un grafico che visualizzi la gestione dell'utilizzo del ChatBot.

Un grafico che visualizzi la gestione dell'utilizzo di un ChatBot potrebbe includere diverse metriche per misurare la performance e l'efficacia del ChatBot. Ecco alcuni esempi:

- **Tasso di completamento delle conversazioni:** questo indicatore misura la percentuale di conversazioni che vengono completate con successo dal ChatBot, ovvero quelle in cui l'utente ha ottenuto una risposta soddisfacente alle proprie domande o richieste.
- **Tempo di risposta medio:** questo indicatore misura il tempo medio di risposta del ChatBot alle domande degli utenti. Un tempo di risposta rapido è un indicatore di un ChatBot efficiente.
- **Numero di conversazioni simultanee:** questo indicatore misura il numero di conversazioni che il ChatBot può gestire contemporaneamente. Un alto numero di conversazioni simultanee è un indicatore di un ChatBot scalabile e in grado di gestire un alto volume di utenti.
- **Feedback degli utenti:** questo indicatore misura il feedback degli utenti sul ChatBot, come ad esempio le recensioni o i punteggi che gli utenti gli attribuiscono. Questo indicatore può essere utilizzato per valutare la soddisfazione degli utenti e identificare eventuali aree di miglioramento.
- **Numero di conversazioni non completate:** questo indicatore misura il numero di conversazioni in cui l'utente non ha ottenuto una risposta soddisfacente dal ChatBot e ha abbandonato la conversazione. Questo indicatore può essere utilizzato per identificare eventuali problemi o aree di miglioramento nella risposta del ChatBot.

- **Numero di conversazioni trasferite a un operatore umano:** questo indicatore misura il numero di conversazioni che il ChatBot non è in grado di gestire e che vengono trasferite a un operatore umano per la gestione. Questo indicatore può essere utilizzato per valutare la capacità del ChatBot di rispondere alle domande degli utenti e identificare eventuali aree di miglioramento.

Queste metriche possono essere rappresentate in un grafico a linee o a barre, in cui l'asse x rappresenta il tempo e l'asse y rappresenta il valore dell'indicatore.

Il grafico può essere utilizzato per monitorare l'andamento delle metriche nel tempo e identificare eventuali trend o variazioni.

## DIAGRAMMA DI GANTT

