

MilliPW - Guida all'uso

Installazione automatizzata

Il progetto include alcuni script che automatizzano e velocizzano l'installazione.

Prerequisiti

- Python 3.11 deve essere installato e presente nella PATH di sistema
- PostgreSQL 15 o una versione successiva deve essere installato e deve essere in esecuzione. È necessario creare un database vuoto.

Istruzioni

1. Scaricare il progetto sul proprio computer. Se si scarica come archivio, estrarre l'archivio in una cartella.
2. Eseguire lo script. Su windows, è sufficiente eseguire il file `setup_and_run.bat`.
3. Inserire i dettagli del database creato nei prerequisiti. Lo script chiede passo passo cosa inserire.
 - Host: di default è `localhost`
 - Port: di default è `5432`
 - Utente: di default è `postgres`
 - Nome del database
 - Password del database
4. Attendere che l'installazione termini. Al termine dell'installazione, non chiudere lo script.
5. Se lo script è stato chiuso al termine dell'installazione, eseguire lo script `run.bat`. Se lo script precedente è ancora in esecuzione, saltare questo passaggio.
6. Aprire un browser a scelta e recarsi a `http://127.0.0.1:8000/`.

Installazione manuale

Prerequisiti

- Python 3.10 o successivo deve essere installato.
- Deve esistere un database PostgreSQL locale o remoto.
- (Opzionale) `git`

Setup Iniziale

1. Clona la repository o scarica i file. Inserisci i file scaricati in una cartella.
2. Crea e attiva un ambiente virtuale. Ciò isola le dipendenze del progetto. Per fare ciò, avviare un terminale e digitare:

```
# Su macOS o Linux
python3 -m venv venv
source venv/bin/activate

# Su Windows
```

```
python -m venv venv
.\venv\Scripts\activate
```

3. Installa le dipendenze:

```
pip install -r requirements.txt
```

Configurazione del database

1. Creare un file `.env`. Copiare il file `.env.example` in un nuovo file chiamato `.env`.
2. Modificare il file `.env`. Aprire il file `.env` e inserire la stringa di connessione di PostgreSQL per intero in `DATABASE_URL`.
 - Per un DB locale, il risultato sarà simile a:
`DATABASE_URL="postgres://utente:password@localhost:5432/nome_db"` dove bisogna sostituire il proprio nome utente, la propria password e il nome del database.
3. Aggiornare la `SECRET_KEY`. Per sicurezza, è consigliato aprire `millipw_django/settings.py` e modificare `SECRET_KEY` in una stringa unica e casuale.

Migrazione del database e caricamento dei dati

1. Iniziare la migrazione. Il comando seguente crea le tabelle del database seguendo i modelli di Django.

```
python manage.py makemigrations hospital
python manage.py migrate
```

2. Caricare i dati iniziali. Digitare

```
python manage.py seed_db
```

Eseguire l'applicazione

1. Avviare il server. Digitare

```
python manage.py runserver
```

2. Visualizzare l'applicazione. Aprire un browser e navigare a `http://127.0.0.1:8000/`. Verrà visualizzata l'homepage dell'applicazione.