

---

# **futura-tech-guide Documentation**

***Release 1***

**Marco Buccio**

**18 gen 2024**



<b>1</b>	<b>Installazione e configurazione</b>	<b>3</b>
1.1	Prerequisiti . . . . .	3
1.2	Installazione . . . . .	3



Il presente manuale raccoglie le informazioni necessarie all'installazione e alla configurazione dell'applicazione Futura.



---

## Installazione e configurazione

---

Il presente documento descrive i passi necessari all'installazione e alla configurazione dell'applicazione Futura per la creazione e gestione del PEI e del PDP.

### 1.1 Prerequisiti

Futura è una soluzione WEB costruita su stack LAMP (Linux Apache MySQL PHP). I componenti fondamentali del sistema e le versioni di riferimento sono le seguenti:

- Linux: selezionare la propria distribuzione di riferimento;
- Apache: predisporre il web server Apache (o in alternativa NGINX) configurando il virtual host per la gestione dell'applicazione;
- PHP: la versione di riferimento di PHP è la 8.1.x;
- MySQL: installare e configurare il server MySQL, versione di riferimento: 5.7.22;
- Server SMTP: L'applicazione invia numerose notifiche via email, sarà quindi necessario attivare un server SMTP;
- Molte sezioni dell'applicazione prevedono l'archiviazione di file. I file vengono archiviati in una folder del server. Quindi lo spazio disco del server dovrà essere dimensionato in modo opportuno;
- Certificato SSL per consentire l'accesso tramite protocollo cifrato HTTPS;

L'architettura del sistema e le risorse dovranno essere dimensionate in modo proporzionale al carico previsto.

### 1.2 Installazione

Per eseguire l'installazione del sistema sarà necessario compiere le seguenti operazioni:

## 1.2.1 Creazione del database

Creare un nuovo schema e un nuovo utente con permessi di lettura e scrittura sullo schema creato. Eseguire l'import dello schema fornito con la soluzione:

```
mysql -u [utente amministrativo] -p [schema] < dump.sql
```

Al termine dell'operazione si avrà il database contenente la struttura e i dati essenziali per il funzionamento.

## 1.2.2 Installazione dell'applicazione

Dopo aver creato il virtual host e assegnato i privilegi adeguati alle cartelle procedere inserendo i sorgenti nella cartella del virtual host (solitamente public\_html). Eseguire l'installazione delle librerie aggiuntive utilizzando composer:

```
composer install
```

## 1.2.3 Configurazione applicativa

I principali parametri di configurazione vengono gestiti dal file Env.php posto nella cartella env. Copiare Env.php.default in Env.php:

```
cp Env.php.default Env.php
```

I parametri di configurazione contenuti nel file Env.php definiscono il comportamento dell'applicazione. Di seguito si riepilogano gli elementi fondamentali:

```
Constants::$devMode = false;
```

Definisce se l'applicazione è in esecuzione in modalità sviluppatore. Se attivata si disabilita il sistema di login.

```
Constants::$db_user_name = ""; Constants::$db_password = ""; Constants::$db_host = "";  
Constants::$db_schema = "";
```

Dati di configurazione della connessione al database principale dell'applicazione.

```
Constants::$db_user_name_alfa = ""; Constants::$db_password_alfa = ""; Constants::$db_host_alfa  
= ""; Constants::$db_schema_alfa = "";
```

L'applicazione consente l'accesso ai dati storici dell'applicazione «alfa». Qui vanno configurati i parametri di accesso al db.

```
Constants::$db_user_name_esis = ""; Constants::$db_password_esis = ""; Constants::$db_host_esis  
= ""; Constants::$db_schema_esis = "";
```

Futura consente l'accesso ai dati storici dell'applicazione «esis». Questi i parametri di configurazione per l'accesso al db.

```
Constants::$encrypt_key = «»; Constants::$encrypt_iv = «»;
```

Chiave dell'algoritmo di cifratura e vettore di inizializzazione. I dati dell'applicazione verranno comunicati al database già in forma cifrata. La modifica di questi parametri o la perdita della chiave renderà irrecuperabili tutte le informazioni archiviate.

```
Constants::$APP_PROTOCOL = «https://»;
```

Protocollo di comunicazione tra client e server.

```
Constants::$APP_URL = «»;
```

Url dell'applicazione.

```
Constants::$APP_PRINT_NAME = «Futura»;
```

Nome dell'applicazione visualizzato nelle pagine e nelle mail.

```
Constants::$app_name = «»;
```

Permette di configurare un path relativo per l'applicazione. Nel caso la soluzione non fosse installata come virtual host indipendente ma fosse raggiungibile in un particolare path.

```
Constants::$ATTACHMENTS_DIR = «»;
```

Percorso di archiviazione dei files allegati. Predisporre tale folder con i permessi adeguati di lettura e scrittura da parte dell'utente che esegue l'applicazione.

```
Constants::$SMTP_FROM_NAME = «»; Constants::$SMTP_FROM_MAIL = «»; Constants::$SMTP_HOST = «»; Constants::$SMTP_PORT = 587; Constants::$SMTP_MODE = "tls"; Constants::$SMTP_USERNAME = «»; Constants::$SMTP_PASSWORD = «»;
```

Dati del server SMTP per consentire l'invio delle mail di notifica.

```
Constants::$maxSeenTimeDelay = 120; Constants::$onlineInterval = 60;
```

Periodicità di controllo dello stato di utetnte online per il monitoraggio degli accessi.

```
Constants::$defaultActionMethod = «_default»;
```

Parametro di configurazione relativo all'esecuzione delle classi di View del framework.

```
Constants::$useVueDevServer = false;
```

Consente di attivare l'utilizzo diretto del server di sviluppo di VueJS. Da disattivare in produzione.

## Office 365

```
Office365Constants::$CLIENT_ID = ""; Office365Constants::$CLIENT_SECRET = ""; Office365Constants::$REDIRECT_URI = ""; Office365Constants::$AUTHORITY_URL = ""; Office365Constants::$AUTHORIZE_ENDPOINT = ""; Office365Constants::$TOKEN_ENDPOINT = ""; Office365Constants::$RESOURCE_ID = ""; Office365Constants::$SCOPES= "";
```

Parametri di configurazione per l'integrazione del login tramite Office365.

## Recaptcha

Per garantire un miglior controllo delle form accessibili è stato introdotto il meccanismo di controllo recaptcha. Per utilizzare questa integrazione è necessario configurare gli appositi parametri:

```
RecaptchaConstants::$CLIENT_SECRET = ""; RecaptchaConstants::$SERVER_SECRET = "";
```

### 1.2.4 I task schedulati

L'applicazione fa uso di una serie di task schedulati per gestire delle code di attività. Tali task vengono attivati inserendo la seguente istruzione n crontab:

- - \* .
- user php [app\_folder]/Scheduler.php