

Progetto di Basi di Dati - Manuale Utente

Michele Lazzeri | 822879

1 Installazione

- Installare PostgreSQL ≥ 9.1
 - Installare l'ultima versione stabile di Apache
 - Installare PHP ≥ 5.4
 - Abilitare i moduli PHP di Apache
 - Abilitare i moduli psql di PHP in php.ini
 - Inserire il contenuto della cartella www in una cartella accessibile da Apache
 - Creare un database nuovo tramite psql
 - Inserire il dump (src/dump.sql) in tale database
 - Modificare il contenuto del file www/lib/conf.php inserendo i dati relativi al database
 - Accedere tramite browser al file www/info.php (se i file sono stati copiati in /var/www basta accedere all'indirizzo `http://localhost/info.php`)
 - Cercare la voce 'include_path' (nel mio caso /usr/share/php)
 - Inserire in tale cartella il contenuto della cartella lib
 - Modificare la password in src/check.sh e rendere tale script eseguibile
 - Creare un nuovo cronjob per tutti i giorni alle 0:01 in cui viene avviata la funzione fix_cron . Per farlo:
 - Da terminale

```
crontab -e
```
 - In fondo al file aggiungere:

```
1 0 * * * src/check.sh >/dev/null 2>&1
```
- Ovviamente al posto di 'src/check.sh' va inserito il percorso corretto di check.sh

2 Note

- Per facilitare il testing è stata inserita un form in alto a sinistra nelle pagine web che permette di impostare una data diversa dall'attuale. In questo modo è possibile modificare la data senza dovere cambiare quella di sistema. Tale form ha un solo difetto: essendoci dei controlli ad ogni inserimento di spese per non andare in negativo sul conto, nel momento in cui si inserisce una spesa s1 in un tempo x, quindi si torna ad una data precedente tramite questo form, si inserisce un'altra spesa s2 e si ritorna alla data x, è possibile che quest'ultima operazione non vada a buon fine, non permettendo l'inserimento della spesa s1 in quanto il conto andrebbe in negativo. Per ovviare a tale problema, basta evitare di impostare una data anteriore. Tale comportamento non è comunque vietato dall'applicazione, essendo *esclusivamente* un form di testing. Nell'applicazione reale tale form non dovrebbe esistere, il rinnovo dei conti di credito e dei bilanci avviene tramite cronjob.
- Il database è popolato da 50 utenti con username 1, 2, 3,... e password 'password'
- Quasi tutti gli utenti hanno conti di deposito, entrate e spese
- Solo gli utenti con username 1 e 2 hanno dei conti di credito e bilanci