

Diario di lavoro

Luogo:	Canobbio
Data:	12.09.2019

Lavori svolti

Ho finito di disegnare la struttura di base del database. Il quale è composto di 6 tabelle:

- users (per gli utenti)
- groups (per i gruppi)
- groups_users (tabella ponte fra gruppi e utenti)
- permissions (per i permessi)
- permissions_groups (tabella ponte fra permessi e gruppi)
- generator (tabella che identifica il generatore, scritta in maniera che sia facilmente espandibile la piattaforma)

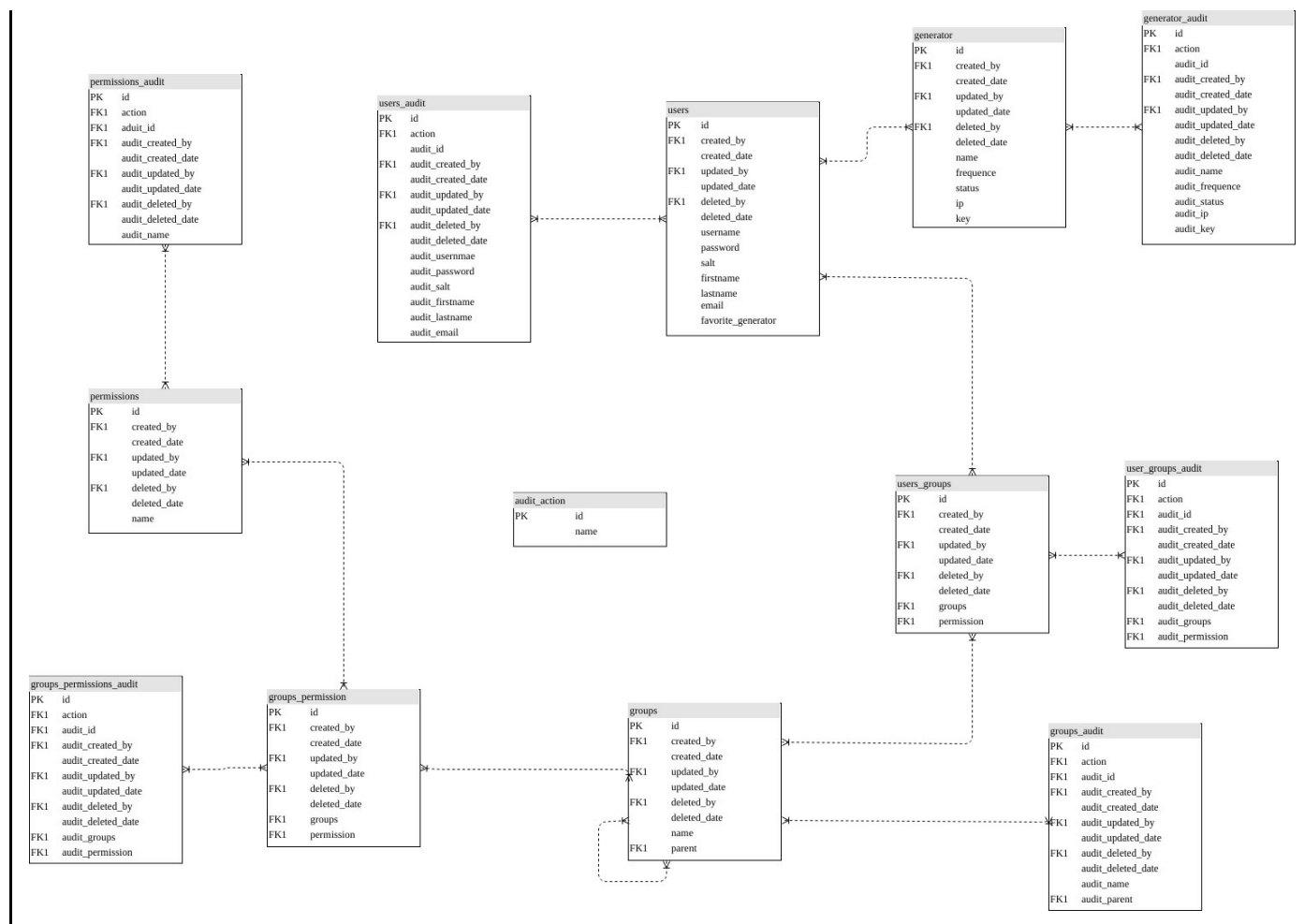
Siccome è richiesta la gestione degli utenti, sarà di base e con poche funzionalità, ho pensato di sviluppare il progetto, con gli audit. Cioè eseguire il log delle azioni sul database. Per eseguire questo mi sono informato su quale metodologia è più appropriata.

Qui descriverò le varie tipologie di log:

1. Eseguire il log di ogni tabella su una tabella di log unica, quindi inserire ogni statement che viene eseguito sulla banca dati in una tabella, l'utente che lo esegue, la data e l'ora.
2. Eseguire il log su una tabella dedicata per ogni tabella, scrivendo simultaneamente su entrambe le tabelle ogni modifiche, sulla tabella di log, verrà ogni volta aggiornato un dato, mai modificato o eliminato.
3. Eseguire i log nell'applicativo. (Probabilmente verrà scartato perchè i dati rimangono più esposti e nel caso vi fossero più applicativi che aggricano su una singola banca dati, i log non sarebbero più autentici.
4. Utilizzare un software esterno di log. (Probabilmente scartato perchè dovrei mettermi ad imparare come funziona, ed uscirei dallo scopo del progetto)

Alla fine ho deciso di utilizzare la prima opzione, perchè un pochino più laboriosa per quanto riguarda la creazione del database. Mentre è più semplice interagire, e creare anche delle interfacce per l'analisi dei log. Oppure per mostrare uno storico.

Quindi ogni tabella del database avrà anche una tabella `audit`. In ogni tabella vi sarà un id, la data e l'autore di ogni modifica (created, updated, deleted), ogni tabella avrà anche i suoi trigger di audit (uno per ogni azione: insert, update, delete). E le tabelle di audit avranno anche una foreign key sul dato attuale.



Problemi riscontrati e soluzioni adottate

-

Punto della situazione rispetto alla pianificazione

Sono un avanti con la con la progettazione, siccome oggi ho finito di progettare il database.

Programma di massima per la prossima giornata di lavoro

Domani dovrò iniziare a progettare le RestAPI per la piattaforma WEB.