

Time Detective Database

Mike Colbert

Database Technology:

After the data was normalized, it appears this application would function best with PostgreSQL. The data structure has numerous relationships. PostgreSQL allows for JOINS which makes managing the data in these relationships easier. In MongoDB, the related documents would be nested making it more difficult to work with the data.

There will be many users tracking many tasks and I expect the database will be busy. PostgreSQL is a modern, fast, reliable, open-source relational database management system that can handle the traffic and row changes. It is a great choice for transactional data such the type we will be inserting frequently.

Data Structure:

The **Roles** table contains the different application roles utilized by the application for authorizations. Roles can easily be added or updated with this database structure. All new accounts will be added as a *user* by default. If a user requires *administrator* permissions, an existing administrator will utilize the *User Manager* to promote the *user* to *administrator*.

The **Users** table will contain user details for each account. Email will be used for identification and must be unique. Passwords will be hashed using bcrypt with a salt. The hash value will be stored in the password database field.

Tasks table contains information about the tasks users want to track. Date_created will be auto populated when the row is written to the database. This will allow the user to know how long they have been tracking this task.

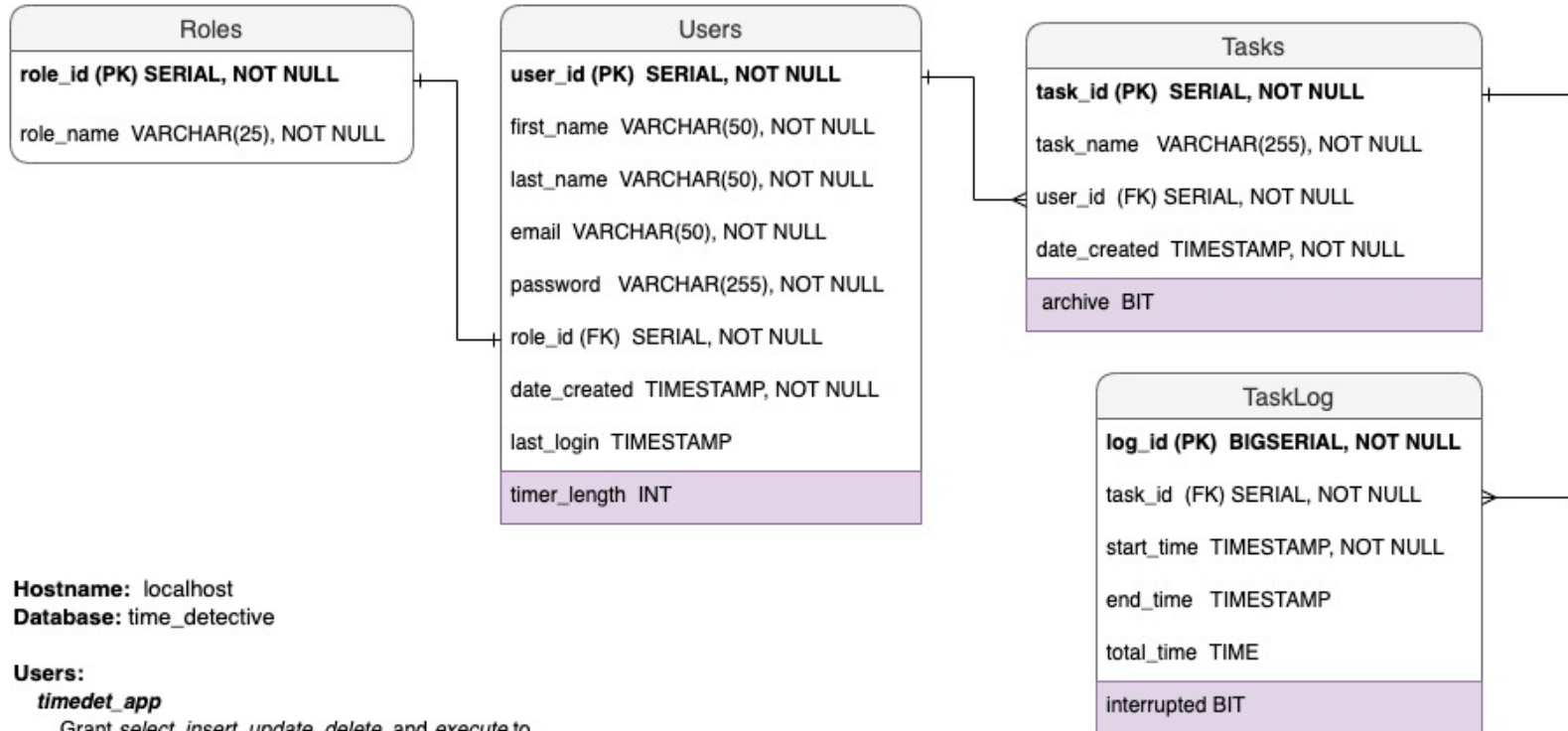
The **TaskLog** table is an associative table connecting Users with Tasks. This table will be key in limiting an individual user's application view to seeing only their tasks. This table will also be key to calculating how

much time was spent on a task. The TaskLog table is populated from the start and stop timer buttons and through a log entry screen in case you forget to start a timer. Users are allowed to delete and edit individual log entries to give the best opportunity for accurate data.

Users:

Two users will be created. The *timedet_app* account will be used by the API server to connect to the *TimeDetective* database on PostgreSQL. This *_app* account will have INSERT, DELETE, LIST, UPDATE, and EXECUTE permission on all tables in the database. EXECUTE permissions are necessary to run stored procedures.

The second user will be *timedet_admin*. This account will have all permissions to the *TimeDetective* database, including all of the tables. This will allow for the *TimeDetective* application management to be kept separate with no chance of crossing over to another database.



Hostname: localhost
Database: time_detective

Users:

timedet_app

Grant *select, insert, update, delete, and execute* to all tables in the time_detective database.

timedet_admin

Grant *all* permissions in the time_detective database

No need to implement until stretch goals are developed.