

Technical Manual

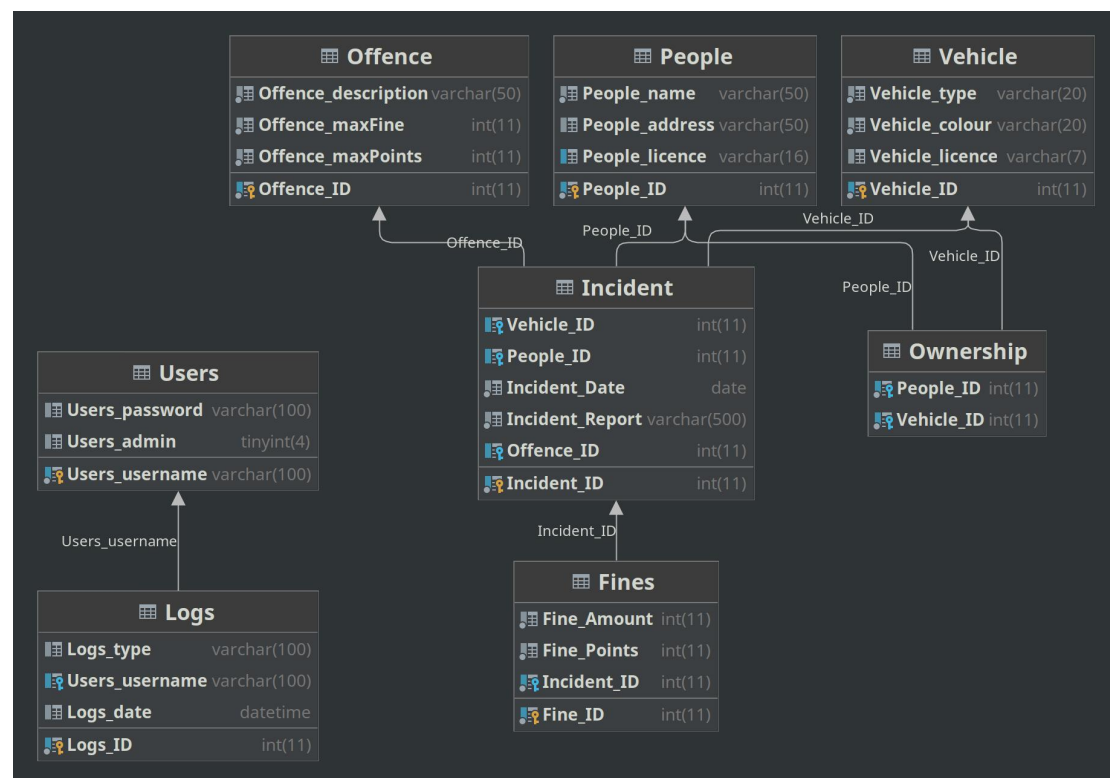
Structure

The code is separated into 4 main parts, HTML, JS, CSS, PHP. JS and CSS files are stored in respective directories., PHP and HTML are stored in the main installations directory. The HTML files only contain the structure of the pages. The CSS files contain the styling for the page. The JS Files contain everything needed to connect the frontend to the PHP and database. The PHP contains all the code necessary to execute the queries on the database.

Running the code

Assuming Apache, MySQL and PHP are installed the files can be just placed in the root directory and run.

Database design



This is the current design of the database which shows the tables with it's columns and how they link together with foreign keys. Apart from the standard design which was given for the task 2 extra tables were added, Users and Logs which are used for storing the login information and the audit logs of the site.

Code design

PHP

For the PHP a config.php was created to store the database connection string and is used across every page. Every PHP file checks to see if the user exists and the admin related pages check if the user is admin and logged in as well, to make sure that no unauthorized users are making changes to the database without access. All the data sent via PHP is sent as JSON data so that it can be easily manipulated on the frontend and then can easily extended in the future.

JS

For the JS each page has it's own JS except for checkUser.js which is used in all the pages. The checkUser.js file is added to all pages to run an AJAX request which then checks if the user is logged in or not and if not redirect them to the login page. If the user is logged in show the user logged in on screen and show the admin relevant links if the user is admin. All the other individual JS files run an AJAX request to get the any data from the PHP files or to show an alert box with any messages that are required to show.

CSS

A template.css was created to have common styles used across all files, for example, button styles. The template also includes a root style which has all the css variables, which currently only store the page colours so that they can be used globally across all css files and then can be changed when necessary. A nav.css was created to have the navigation styles. Any other pages that require individual styles such as the the reports page have a separate CSS file. Each individual CSS file imports nav.css and the nav.css imports in the template.css