SQL injection refers to when an injection attack where an attacker can execute malicious SQL statements that can thus control a web application's database. Hence, by accessing a website’s database without permission, the attacker can modify or even erase important data about the website. This, however, is not a problem, since we are using the Laravel framework.

Laravel’s object-related mapping uses parameter binding to avoid SQL injections. Parameter binding protects from SQL injections by ensuring that attackers can’t bypass the query data, which could modify the query’s intent. Let’s consider an example, if in the instance form field we are asked to supply an email-address, but instead we write: ‘abc@example.com’ or 1=1. 1=1 syntax is a simple logic that always evaluates to true. When this is used with “or”, all records will be returned from the table form. Let’s look at another example. If, for instance, we supply an email address again and we write: drop table users, like this: ‘[abc@example.com](mailto:abc@example.com)’ ; drop table users; , there would be a dangerous error in the database. If the account is responsible for executing the application queries and has the “drop” privilege, it could destroy and erase all data from the users table. However, Laravel's parameter binding can remove these problems. When it is used, the whole input will be quoted, thus the 1=1 or drop table users will not be executed.

Another attack could be Cross-Site Scripting. Laravel will automatically escape any HTML entities passed along with a new variable, using {{}}. What happens is that if, for example, an attacker passes the following string: My list <script>alert("spam!")</script>. If this is saved in the database without any filters, then an alert display window will show up without the admin’s consent. However, since Laravel provides an escape tag, using {{}}, then it would render a string like this: My list &lt;script&gt;alert("spam !")&lt;/script&gt, preventing the website from displaying an alert window.