What do you think about web application security and concerns about confidentiality? How do you see this personally affecting you as the developer, as well as users and potentially global partners?

Professor and classmates,

Data leaks are likely the biggest concern for most people when it comes to web application security. I used to think that cloud Technology made web applications less secure in this respect but my previous course taught me that this is not necessarily the case. Most of the time, securing data is the responsibility of both the cloud platform and the developer in varying degrees depending on the type of platform. This means that when I was blaming the cloud platforms for poor security, I was blaming the developers in part. I realize now that security will always fall on the developer in varying degrees no matter what type of web application it is. What I’m trying to say is that web security is our own responsibility. Even if the platform is mostly or even entirely responsible for security, we should do our research to ensure that the platform has the security that we require. However, if the platform promises security that it can’t provide, that is when the blame falls on them.

Hacking is something else entirely as there will likely never be a foolproof plan to prevent hacking. Hacking strategies are constantly evolving so there’s no way to prevent any hacking methods that haven’t been invented yet. The best we could do is prevent hacking methods that we know already exist. On the bright side, this could still prevent a good amount of hackers from accessing the application. This is still much better than no protection at all.

Class, the University has updated the discussion questions from TECHNICAL to non-TECHNICAL. Below is the previous TECHNICAL question. You may answer either or both questions, thank you.

BELOW WAS THE ORIGINAL TECHNICAL QUESTION:

Compare and contrast the routing system used in Express and the Laravel PHP framework. From a coding perspective, provide three code examples of how routes are similar and three code examples of how routes differ between the two technologies.

Professor Estey,

Express and Laravel are two popular frameworks with a lot of differences, but I’ll just focus on the routing systems.

Express uses a GET request to display a page. Express’ routing system is powerful and involves two other requests as well. The POST request creates new data and the DELETE request deletes data. The following is an example of how Express defines a route;

app.get('/about', (req, res) => {

res.send('About Page');

}); (Mikoliuk, n.d.)

Laravel also uses a GET request to define a route to a page. Their routing system is simple, allowing the user to define routing using a PHP or plain text file. Laravel also supports Route parameters which allows for the creation of dynamic URL’s. The following is an example of how Laravel defines a route;

Route::get('/', function () {

return 'Welcome to Laravel!';

});

References:

Mikoliuk, E. Express vs. Laravel: A Detailed Comparison for Web Developers. (n.d.) MarketSpash <https://marketsplash.com/tutorials/express/express-vs-laravel/>

Davidson, T. Laravel vs. Express- Which Framework is More Powerful?. (Dec 11, 2022) CleanCommit <https://cleancommit.io/blog/laravel-vs-express-which-framework-is-more-powerful/>

Scott,

Searching for vulnerabilities is an incredibly useful way to incorporate AI. I knew there were programs that did that for developers, but I never considered AI. Hackers don’t have to be the only ones using AI though. An advanced enough AI would possibly be able to predict future hacking methods as well which would be much more secure than what most developers could do. If this sort of technology is used by developers, it could prevent a lot of attacks.