

Alejandro Lebron

Quiz 2 Answers:

1.1:

When I first started building this project manager application, I didn't realize how much server-side PHP would be necessary for basic functionality. The first major feature is user authentication, which requires the server to securely validate usernames and passwords against a database while hashing and salting passwords, so they're never stored in plain text. The second critical feature is database management and data persistence, where PHP handles SQL queries to store projects and users while enforcing business rules like requiring three members per project. The third essential feature is input validation and security enforcement, where server-side PHP validates every piece of user input before it enters the database to prevent SQL injection attacks and other vulnerabilities. What surprised me most was realizing that the browser alone cannot handle these tasks securely, it needs a trusted server to manage everything behind the scenes.

1.2:

The first action I took was implementing proper password hashing and salting, where each user gets a unique salt and their password is hashed using SHA-256 before being stored in the database. This means that even if someone accesses the database, they cannot read passwords or use precomputed hash tables to crack them, making brute force attacks a lot harder. The second critical security action was using prepared statements with parameterized queries in PHP to prevent SQL injection attacks, where user input is treated as data values rather than executable SQL code. By using `mysqli_prepare()` and `bind_param()`, any malicious SQL commands that someone tries to inject through form fields have automatically escaped and treated as regular text. I was really surprised to learn how easily attackers could manipulate databases without these protections, but with prepared statements and proper validation, our application is way more secure.