Emily Foglia

Professor Jeffrey Seaman

CMPS 361

22 October 2021

Session Control

As defined by Author Ethan Brown, sessions are a more convenient way to maintain state than cookies alone. Sessions are implemented by storing something on the client so that the server can identify each unique client's requests. A cookie can be used for this purpose. Alternatives exist, such as decorated URLs, but these methods are messy, difficult, and inefficient. HTML5 provides a new method called "local storage," which functions similarly to a cookie with a larger storage space. The "express-session" middleware is a solution that offers sessions, which is available for Express.js applications.

Sessions have several uses that can improve the user experience on your website. Sessions are useful for saving user preferences that are applied across multiple pages. It can also be used to provide user authentication information. A session allows a user to log in once rather than on every new page. When comparing cookies to sessions, sessions offer much more functionality than cookies. Sessions are generally server-side and allow for much more functionality than a small, local cookie.

As previously justified, I do believe that sessions can make your website better. It all depends on the functionality needed for your website to provide the best user experience. For example, a read-only website with a few JavaScript buttons does not need to bother with sessions. A cookie to track viewed pages is sufficient in this scenario. However, a website like the New York Times needs to implement sessions. If a user had to log in on every new article to

prove they have a subscription, the user experience would be much worse. Like most things with programming and technology, the choice to implement sessions is very much unique to the application.

References

Brown, E. (2020). *Web development with Node and Express: Leveraging the JavaScript stack* (2nd ed.). O'Reilly Media.