**Developer requirements:** as a student, implementation capabilities are limited

* Low-to-no-cost implementation
* Framework
* Web server to host web application
* Database to hold data
* Programming languages to implement backend

**Technology stacks considered:**

1. **LAMP:** 
   1. Linux (OS)
   2. Apache (server)
   3. MySQL (database)
   4. PHP (programming language)

**Pros:** Open source, low-cost, highly documented, flexible, and easy to use

**Cons:** Limited scalability, MySQL is limited with large databases

1. **MERN:**
   1. MongoDB (database)
   2. Express.js (web framework for Node.js)
   3. React (front-end framework)
   4. Node.js (server)

**Pros:** Open source, scalable (huge library of modules), cloud-compatible, uses one programming language (JavaScript)

**Cons:** JavaScript can be difficult

**Considerations:**

1. Using LAMP would require me to setup a Linux machine, which would add a layer of complexity to the project (not necessarily complex, but rather steps I would not have to take otherwise).
2. MySQL would require relational database creation including entity-relationship model diagrams, which take a bit of time.
3. I have coursework with LAMP and understand the full stack.
4. I already understand a bit of JavaScript including Express, React, and Node frameworks.
5. MERN stack is scalable and uses one programming language.
6. I have used MERN technologies more recently than LAMP technologies.

**Conclusion:** Considering the drawbacks of using the LAMP stack, despite my prior knowledge, I plan to build the web application using the MERN stack. I have coursework in both LAMP and MERN and would like to further my knowledge of MERN with this project.

**References:**

[How to choose the right tech stack for web development (educative.io)](https://www.educative.io/blog/choose-a-web-development-tech-stack)