NodeJS

The implementation of the event loop in Nodejs allows for some quite interesting use cases in the development of applications, the first thing I heard about node, was, asynchronous, callbacks, and event loop, well, by developing my own projects and snippets, with the help of the Express framework, what you want the most in node, is, asynchronous logic, every function that can be async, should be async, always using await, along with the non-blocking version of every integrated function ,as long as your function is declared as async, also, the capabilities for templating and concatenating strings, allow for some clever dynamic site DOM manipulation. As I was developing, my first app in node , a file server for downloading and uploading files, to and from my computer, I was able to leverage the access to the filesystem of node, to simply, upload with multer ,download some static files, and use a simple for loop with the templates, to create a button for every file in the directory, before making it all async, but as you all know the biggest problem with node, is that, its JavaScript at the end of the day, where a parsed Json could be a system vulnerability , and weak typing, but this pales in comparison to the lack of native threads, the event loop might look flashy, but one CPU bound operation can bring an entire app down quite easily, and that's where web workers, and the map method comes in, of course, you didn't think I was going to concatenate strings like that with JS, no, I map the entire list of files to create html elements, and then join all of the strings with join, and then return the template to the client, to inject in a strategically placed div, this can be done, either in the server, or the client, which is basically ssr vs csr debate, but im not that good at frontend, so im not going to go in a big statement about it.

In conclusion, nodejs, is a very competent language, and very useful , for projects with any scope, in the general usability of the language npm, is a big plus for me, theres a lot of packages out there, iteration speed and startup time, is very good, though not as good as common lisp, but in practice its almost the same.