NodeJS Frameworks

ExpressJS:

ExpressJS is the most known framework to be used in node due to its simple nature and being aligned with the single-threaded nature of JS. Express contains several features such as:

- Using several features from Node JS and making it faster.
- Using Async programming to schedule multiple operations to be done independently
- Having the ability to be tested in any area of code.

Pros:

- Allows you to use JS for both back-end and front-end
- Can handle a huge number of requests
- Open source
- Easy to learn

Cons:

- Express depends on callback functions due to the single threaded nature of JS which can be hard to learn.
- Express depends mostly on middlewares
- Express needs to be organized due to it being pattern dependant

---

KoaJS:

KoaJS is one of the more dominant frameworks and it is mostly used for creating APIs while being slightly similar to Express which makes it a flexible framework same as Express but more lightweight.

Pros:

- Lightweight compared to other frameworks
- Customizable middleware to personalize the user experience
- Has the support for generators which cleans up the code from all the callbacks so no callback hell

Cons:

- Community is small so it's hard to find a lot of documentation and examples for
- It isn’t compatible with express style middleware
- The generators used by Koa to clean the code are not compatible with other frameworks.

MeteorJS:

MeteorJS is specific to and uses only Javascript where it’s the most comfortable to develop a code for different systems whether that's for mobile devices (Android & iOS) or desktop devices.

Pros:

- Cross-platform which helps with implementing everything for android and ios in one place.
- Open-source with community documentation
- Can be integrated with any framework easily
- Easy to learn

Cons:

- The framework is dependant on several pre-built packages which can cause conflict in certain projects
- Although it can create mobile web applications easily it isn’t as effective as creating them for native desktop
- Using express is easier in creating web applications

LoopbackJS:

Loopback is known to offer better compatibility with any framework as it can be integrated with different APIs. It is also very flexible, having the ability to connect with several devices and services.

Pros:

- Good documentation which facilitates working on it
- Can connect with several browsers, databases and services
- Good community support
- Can be integrated with any framework
- Fully functional rest API

Cons:

- Steep learning curve
- Not recommended for huge long term projects
- Has a good community but it isn’t as big as express
- You need to write a lot of SQL queries and convert them yourself in the models

---

NestJS:

NestJS is mostly used for building scalable applications due to its flexibility with its libraries. This framework is also built using typescript so it sticks to the rules of TS which makes it useful for backend work.

Pros:

- Easy to use external libraries
- Easy to learn
- Compatible with Angular
- Can simplify complex GUIs

Cons:

- Tough to debug
- Can be tricky to use because simple things such as adding routes isn't so straightforward in Nest
- Can feel outdated due to the similarity of it to .Net Core

References:

- **https://www.simform.com/blog/best-nodejs-frameworks/**

- https://apiko.com/blog/express-mobile-app-development/
- https://cubettech.com/resources/blog/introduction-to-koa-javascript-koa-js-pros-and-cons/
- https://www.rlogical.com/blog/why-use-meteorjs/
- https://www.voidcanvas.com/loopback-pros-and-cons
- https://www.vairix.com/tech-blog/should-you-use-nestjs