

Forum:[Hack Forums](#)

Thread: [What's best for interactive websites?](#)

Kuso - July 17, 2017, 11:11 am

So I am planning on creating a website with interactive features like leveling system, rewards, achievements, Marketplace, interactive profiles and activities, etc.

Say I were to build a site with these type of features, what would be a suitable back-end, front-end?

I was thinking about just building the MVP and seeing what it looks like in Ruby Rails for now, PostGress, and I could possible add Javascript/JQuery for some of the interactive features. But I was sitting here thinking that this would probably be better off as a MEAN (Mongo, Express, Angular 1, Node.JS) stack application?

Node.js can deal with those loads better, especially with the large package returns this might have. However, I keep hearing it is more ideal if I am building a site and not a app then I should stick with Ruby Rails.

I would have done React MERN but I am still getting the hang of that and redux. Its kind of tricky to pick up, but I could possibly see.

I'm thinking Rails could be efficient though, and relational databases I'm fine with.

Shroomies_temp - July 18, 2017, 3:06 pm

If its meant to be an operational website. Go with technologies you are confident with (which i think appears to be RoR).

All the features you want should be doable in any programming language and the slight speed boosts you might get from picking one stack over the other are not worth security/bug prone in the end.

notluther - July 18, 2017, 6:16 pm

You say you're thinking about building up an MVP for this, so I would recommend just using whatever tools you're comfortable with. You don't need to worry about anything other than functionality, that's sort of the point of an MVP.

If you find that the tools you're comfortable with are sufficient for the project, even better.

Kuso - July 19, 2017, 6:22 pm

Yeah going with something I am comfortable with is understandable. I am actually asking in the sense which is more suitable per say between RoR vs Node.JS for backend. Was thinking about Node because it can handle heavy data loads better than RoR. If the site I intend on building is going be heavy in JS functionalities for the interactive features and or animations, it would perhaps be a choice for node.

But I couldn't really decide seeing RoR is recommended for websites in general. I recently just picked up both and can learn how to make a full CRUD with each, but I have not yet understand to the full the extent of the data utilizations.

For example: <https://vss-blog.herokuapp.com/>

That's my first website blog app with simple CRUD for posting, replies, and sign in and sign out VIA devise user models.

But when I added panning animations from the top of the page, the app starts to have a rather noticeable lag. Its not my MacBook either, it's latest one so it fresh and should be running it fine. Not a hardware question. I am pretty sure it is a framework issue.

Elite Hacker 69 - July 19, 2017, 9:54 pm

I would do this in php/js and for data like leveling up and stuff just store it in the database then work with objects and save as I go

KiwiDesire - July 21, 2017, 6:42 am

Frameworks like Rails and Django are quite easy to get into and easy to maintain. Especially Django with its admin GUI.

I've heard that nodejs is good for creating smaller responsive sites.

Shroomies_temp - July 22, 2017, 8:03 am

(Jul 19, 2017 06:22 PM)\$tupid Young Wrote: [\[CITE\]](#) Yeah going with something I am comfortable with is understandable. I am actually asking in the sense which is more suitable per say between RoR vs Node.JS for backend. Was thinking about Node because it can handle heavy data loads better than RoR. If the site I intend on building is going be heavy in JS functionalities for the interactive features and or animations, it would perhaps be a choice for node.

But I couldn't really decide seeing RoR is recommended for websites in general. I recently just picked up both and can learn how to make a full CRUD with each, but I have not yet understand to the full the extent of the data utilizations.

For example: <https://vss-blog.herokuapp.com/>

That's my first website blog app with simple CRUD for posting, replies, and sign in and sign out VIA devise user models.

But when I added panning animations from the top of the page, the app starts to have a rather noticeable lag. Its not my MacBook either, it's latest one so it fresh and should be running it fine. Not a hardware question. I am pretty sure it is a framework issue.

Just go with the language you are most comfortable with. People tend to overlook "language speed", but it shouldnt be your priority, unless you are looking to have several thousands of users online at once. And by that time you'll probably be:

- a) rewriting the whole thing, cause you realise how many architectural mistakes you have done
- b) buying a machine that will actually run that no matter what language you are using

Quote:But when I added panning animations from the top of the page, the app starts to have a rather noticeable lag.

If its laggy because of animation then backend has NOTHING to do with it. It just means that the graphics card handling browser sucks (or the page is done shitty, also happens).

SUPERTIME - July 22, 2017, 1:00 pm

With today's frameworks you cannot go wrong, as long as they support the features you need. There are many flavors out there.

Kuso - July 24, 2017, 9:48 am

(Jul 22, 2017 08:03 AM)Shroomies_temp Wrote: [\[CITE\]](#) Just go with the language you are most comfortable with. People tend to overlook "language speed", but it shouldn't be your priority, unless you are looking to have several thousands of users online at once. And by that time you'll probably be:

- a) rewriting the whole thing, cause you realise how many architectural mistakes you have done
- b) buying a machine that will actually run that no matter what language you are using

If it's laggy because of animation then backend has NOTHING to do with it. It just means that the graphics card handling browser sucks (or the page is done shitty, also happens).

Ok.

Also the graphics card is 1080, chrome, and the code page was actually tested on code pen which runs fine