

Engineering Project - Backend Api

Build a Simple Messenger Api

We'd like you to build a very simple messenger api that would enable a web app to build a simple messenger application.

High Level Requirements

Build an api that supports a web-application to enable two users to send short text messages to each other, like Facebook Messages app or Google Chat.

Ignore Authorization, Authentication and Registration. Assume this is a global api that can be leveraged by anyone, security is not a concern.

1. A short text message can be sent from one user (the sender) to another (the recipient).
2. Recent messages can be requested for a recipient from a specific sender - with a limit of 100 messages or all messages in last 30 days.
3. Recent messages can be requested from all senders - with a limit of 100 messages or all messages in last 30 days.
4. Document your api like you would be presenting to a web team for use.
5. Show us how you would test your api.
6. Ensure we can start / invoke your api.

Other Considerations

Here are some additional considerations to help you decide what and how much to build:

1. We're only expecting a couple hours to a half day of effort, though you're free to spend more if inspired.
2. Any functionality beyond the core features above is strictly optional but we'd be lying to say we're not impressed by something really cool you sneak in, whether it's from existing apps or your own idea entirely
3. If you get stuck and don't feel like you can finish this in a short amount of time (or at all), that's fine; just build what you can and supplement it with a detailed written description/design on where things ended up, where you would go next with more time, and be ready to talk about it
4. At any point if you'd like help or feedback with something you're stuck on feel free to reach out to me by email (pretend I'm another experienced engineer on your team but not on the same project); how much help you want

is up to you, and while asking questions is not viewed negatively we love seeing how much you can accomplish by yourself even if it's not ideal since we all do a lot of self-directed learning at a startup!

`## Ground Rules`

We'd also like to set some ground rules to keep this as fair as possible:

This should be original code rather than something you've written previously, borrowed from a friend, or found on the internet, though incorporating open source or third-party code is fine as long as it's clearly distinguishable from original code.

`## Evaluation`

Final code project in a ready to build/run state along with instructions, notes, and any related documentation should be checked into a public GitHub project with the link emailed to me for our team to review

We may or may not use in-office interview time to discuss your solution, so assume that we won't and that what you've included may be the only explanation seen by multiple technical people on our team

Creativity, clarity, design decisions, understanding the challenges of the problem, and being able to extrapolate further if/when we discuss it are more important than the volume of code written