

# Point Backend Engineering Challenge

**Objective:** Build a minimal twitter-like GraphQL API. In addition to the requirements listed below the other criteria which need to be completed to successfully accomplish the challenge are to make the code readable, reusable and extendible.

**Time Estimate:** The requirements of this coding challenge should take about 3-4 hours to complete.

**Submission:** Please publish your source code and ReadMe on a public repository on Github or Gitlab. Feel free to e-mail [kenan@getpoint.io](mailto:kenan@getpoint.io) with any questions.

## Requirements:

- There needs to be basic authentication using an e-mail and/or phone number and password. This can be some out of the box solution like Passport.js (Node/Express), Firebase or Auth0, etc.
- The API needs to leverage GraphQL (<https://graphql.org/>)
- The data does **NOT** need to be stored in a database and can just be key/value data stored in memory, redis, memcached, etc.
- The API needs to support the following functionality:
  - Creating a new user with e-mail and/or phone number and password
  - Logging in as an existing user with the same credentials provided when creating a user
  - Posting a message linked to the user
  - Retrieving all the messages that a user posts
- A README.md file (markdown) describing the functionality of the API along with the pros and cons of the approach, the routes for each of the endpoints, and the steps to run the project locally along with any dependencies which need to be installed prior to running the project.

## Bonus:

- Database storage (ex: No SQL, MySQL, PostgreSQL)
- Multimedia support (uploading a picture or video)
- Editing/Deleting a message
- Adding user relationships (i.e. friendships, follow/follower, etc)
- Socket notification a client can subscribe to be alerted of a new tweet being added
- Anything else that you can think of!