

INSTAGRAM CLONE BACKEND TEST

We are building an Instagram clone where users can post photos.

Backend Requirements:

- Basic Auth logic, using JWT.
 - o Sign up needs to encrypt the password before sending it to the DB.
 - o Sign In should check the encrypted password and return a token.
- Users can create Posts *
 - o Description
 - o Comments
 - o We are **NOT** going to upload any kind of file / Image.
- Users can like a post *
 - o Keep in mind, only one like per user is allowed.
- Users can Unlike a post *
 - o Keep in mind, this only works when that post was already liked. TIP: You can use the same endpoint for doing Like/Unlike behaviors.
- Users can comment on a Post *
 - o Posts should have a limit of 100 characters.
 - o If a user sends more than 100, it should be trimmed to 100.
- Users can follow another user *
- Users can un follow another user *
 - o Keep in mind, this only works when a user already follow that account. TIP: You can use the same endpoint for Follow/Un follow.
- User needs an endpoint to retrieve their user information *
 - o NickName
 - o CreationDate
 - o Followers

Observations:

* These endpoints are protected. They are only accessible with the Authorization token that correspond to the proper user. This Token needs to be sent with the Authorization header in the request.

Some stuffs that you should keep in mind:

- Express and NodeJS are required in this test.
- TypeScript should be a primordial syntax to use. (you can have some functions or helpers in pure JS)
- Classes and inheritance are really worth it. Try to create proper patterns for objects.
- MongoDB is required. You can use Mongoose if you want.

Nice to have (but not required):

- Docker deployment
 - o Add DockerFile docker-compose and DockerIgnore files to the project. So it can be deployed in a local machine.
- Code documentation
 - o Add basic documentation to each class / method.
- Create a Postman Documenter with the endpoints listed before.