

Yes, I believe the GitHub Gist API is RESTful because it is stateless. All of the requests that we submitted were atomic operations, we gave all of the necessary info and there was no reliance on previous session data. The basic CRUD operations are supported, so we are able to create, read, update, and delete gists in GitHub (as atomic requests). Authorization was required, but the client is responsible for submitting a bearer token with each request, keeping the requests as stateless.

The way I organized the tests was more of an integration test rather than individual unit tests because some of the tests relied on previous tests. For example, the delete test (#9) relied on the gist created in test #3. These tests were not fully self-contained/stateless but this particular usage of stateful tests does not affect the 'RESTfulness' of the GitHub API.