Hiring Homework Problem:

Candidate applicant should create a simple Spring Boot Sample Restful application (spend less than a few hours on this or submitted a sample repo with better or similar capabilities – doesn't have to be the Family Pet app requirements defined below nor even java Spring, must have more than 60% of your user commits or be a major open source project). Can be any version of Spring, in fact any code like C#, python, etc, some we can discuss your coding over an in person interview – code review.

Homework App Summary:
A family pet tracker, that tracks pet name, pet birthdate, pet species, veterinarian's name, and list of doctor visits (date, reason), create a CRUD set of restful end-points for getting a pet (& medical history), a list of pets, and create, delete, & update functions.

Functional Should have:
1. the family pets & medical history is persisted to a db (nosql, sql, or super fast persistent /dev/null db)
2. the system should track family member responsible to take care (this could be a enum or static list)

NFR – Non Functional Requirements:
1. (tests) at least 1 unit test per pattern layer (ie: 1 persist, 1 service, & 1 for controller api)
2. Commit code to a github, gitlab, or some public shared db for checkout and code review before onsite interview date
3. Use gradle for dependency management and tasks (or mvn, ant-hill, whatever you want)
4. Create model objects, service Manager(s), and controller web-api classes as appropriate
5. Should be able to be checked out and gradlew :test and build app bootRun (or command line java –jar …) whatever, lang implementation

Bonus Points:
1) SWAGGER2 end-point definitions
2) Component tests with Embedded test DB (using embedded Mongo `de.flapdoodle.embed`)
4) Actuate enabled end-points
5) Spring Security enabled (Permissions)
6) Additional persistent table/collection for the Family and Family member db.
7) CI (travis, Jenkins, cloudbees, shipcode- saas, etc) deploy to cloud provider use free one, don't spend money.
8) Sample rest client tests for running end-points (via whatever approach you wish). +2 Feign Client
9) Container deployment manifests any any other coolness related to microservices, spring, cache

Don't spend a lot of time, at most, whatever applicant can get done in an evening, highly encouraged to use Spring Initializr bootstrap or customizing a spring sample app to fit the requirements. The objective here is to evaluate capabilities beyond typical "fizz-buzz" whiteboard style code snippets and look-discuss-review candidate's background to utilize professional software development tools and environments. Do as little or as much as candidate would like, this is uber micro-sprint.

Whatever is not done is fine, the objective here is to have a candidate created app (even a broken one that doesn't compile) that we can mutually discuss and review during on-site interview. Like at the end of a sprint

do the interview as an agile-scrum review, but instead of 40-80 hrs of work effort, we’re looking at something very simple less than 4 hours (max) to gauge each other work personalities and technology capabilities.