Strider Technical Assessment Briefing

This briefing is to give you information to be as successful as possible in Strider Technical Assessment. The STA is a take-home coding test that you can do on your own computer at the time you find best.

Instructions

- Suggested time to complete the test is located in the test itself. The number of suggested hours ranges from 6-8hrs depending on the test. This does not include the time to set up your environment.
- Do not post this test anywhere on the public internet (including Github, Gitlab, etc.)
- We recommend using the programming language and frameworks you are most familiar with to solve the problem
- Make sure to include instructions in the README for the reviewer to be able to set up your project in his or her local environment.
- Reviewers will only see what is in your ZIP file, so make sure that any comments that you want to make about the project are left in the README.
- Write your name towards the top of your README; this helps us identify solutions after unzipping more easily
- The code you write should be as production-ready as possible given the time constraints.
- Use git during your development process and commit regularly and intelligently. Use it as you would use it in a production project (don’t just dump everything into one commit). We will read your commit messages and evaluate how you use git.

Submission

- Save your whole project including `/.git` as a ZIP file (please avoid other compression formats), with title in the format "[firstName]-[lastName]-[test-version]-[test-domain-type].zip" e.g., "John-Smith-2-0-web-front-end.zip"
- Do not include on your ZIP file managed packages' installed files (e.g node_modules, vendor, etc)
- Please include, if any, collections files to help testing your solution (e.g Postman, Insomnia)
- Upload your project to Dropbox or Google Drive and make the link public.
- Fill out the submission form: https://vmhvvm5cp60.typeform.com/to/leQMMtIl

What help you can get

- You may use the internet, Google, StackOverflow, libraries, etc to solve the project
- You cannot reference other people's solutions for the same or similar test
- You cannot get outside help from anyone
- You can ask questions about requirements, submission, etc to your Strider recruiter

Evaluation criteria

Below are lists of criteria we check in each technical assessment. Please note that not all criteria apply to every assessment.

▼ **All Asessements**

- **Fulfilling requirements**
  - Happy paths
  - Corner cases (validations, constraints, etc)
- **Architecture choices**
  - API endpoints structuring (methods, paths, parameters)
  - Design patterns (service layers, repositories, mediators, builders, etc)
  - Module/folder organization
  - Technologies (frameworks, libraries, database)
  - Scalability bottlenecks
- **Cleanliness of the code**
  - General pattern consistency (naming, organization)
  - Presence of clean code practices
  - Code smells (conditional statement nesting, hard-coded values, duplication, method complexity, inconsistent returns, etc)
  - Variables and methods naming (readability)
- **Use of best practices**
  - Project organization
  - Dependencies management
  - Framework's recommendations and built-in features usage
  - Configuration management
  - Version control (git commits tree, messages)
- **Quality of written content (code comments, README)**
  - Required sections in README according to project description
  - Clear instructions to run the project
  - Endpoint documentation
  - Clarity of written English
  - Depth of thought

- Back-end & Data Engineering only
  - **Efficiency of database queries**
    - ORM usage
    - Pagination strategy
    - SQL Join clauses
    - User input sanitization
- Back-end only
  - **Tests**
    - Coverage
    - Chosen types
    - Suite organization
    - Chosen scenarios and assertions
- Front-end only
  - **UX**
    - UX should be reasonably good
    - Colors, style, typography, etc are not important as long as the application is usable

Variations

To accommodate different types of developers, like front-end, back-end, and full-stack, we allow you to choose the variation of the test you want to take. The factors to consider when choosing which variation to take are:

- What you're best at. You'll want to choose the variation that will best show off your skills.

- What job you want next — do you want to be a mobile dev, front-end, full-stack, etc? Companies typically consider applicants for roles based on the test that they take. For example, if they are hiring for a front-end role, they will want to see front-end code in the form of a front-end or full-stack assessment.
    - If you want to be considered for more positions through Strider, you can take this test more than once with different variations. Just let your Strider recruiter know, and we can schedule both.

The available variations are:

- 💻 Web Front-end
- 💻 Web Back-end
- 📱 Mobile (coming soon)
- 📱 Data Engineering - Build an ETL pipeline, answer some questions with SQL, and explain how you would architect the system in a real production environment

❗ Environment setup ❗

We highly recommend that you set up an environment in your language and framework of choice (according to the variation you choose above) before your scheduled time for the assessment. That will ensure you can spend 100% of your assessment time on writing code for the project rather than wrangling your environment.

Scheduling & next steps

When you are ready to take the assessment, let your recruiter know the date you want to receive it and which variation you wish to take. **You'll have one week to turn it in.**