



Amicus - Technical Interview Prep

Hello! Thank you for taking the time to interview for the Staff Software Engineer role at Amicus. We're excited to have you move forward to the next step of our process. The purpose of this challenge is to evaluate your ability to solve problems with code and get an idea for how you think about solving technical problems.

We know your time is important, so **please do not spend more than 2 hours working on this!** We feel this is enough time to gauge your ability without taking up too much of your day.

Prompt:

At Amicus, one of the many technical challenges we solve involves the collection, aggregation and classifying of documents. Documents for a particular legal case come in a variety of formats and orientations, and can range from 100s to 100,000s of files. One problem that stems from the amount of information we receive is the inability to classify a document correctly. When our system fails to process a document correctly, we need to flag the document as needing review and output that information to our staff to be reconciled.

- You will receive an email from Replit - if you don't and it isn't in your Spam folder, please let us know so that we can resend the invite
- Click Join Replit Multiplayer and create an account (this is a free account)
- If you are experiencing any access issues, please try to logout/ login again and then re-try the link. If this does not work, please contact us ASAP
- This assignment will need to be completed in Python
- Using the file documents.csv under Files in the upper left corner, complete the following tasks below.

Tasks:

1. Create the data models needed to store and process the information in documents.csv (this can be done as classes or objects, no db needed)
2. Create API endpoints to expose data
 - a. An endpoint to get all documents that need HITL (status == NEEDS_REVIEW)
 - b. An endpoint to save changes once the document is reviewed
 - c. An endpoint that exposes duplicate entries in documents.csv
3. Please use an http framework to mock out the API calls. We've intentionally omitted request/ response criteria to test design of the APIs

4. Document your work! Please include reasoning behind the use of third-party libraries (if any), technical tradeoffs, etc.
5. At the end of the assessment, take 5 to 10 min to details any improvements or changes you would make to the code

Please send us an email when you are finished so that we can review your work.

Feel free to reach out to Rachel at rachel@amicuslaw.ai with any questions. We're looking forward to seeing your work!