

Limitation

None.

Compiling the API

No issues.

Solution

Requirements

1. Create a client
 - a. All fields are required
 - b. Send an email and sync documents after a client is created (using the mock repositories provided)
2. Edit a client
 - a. All fields are required
 - b. If the email has changed, send an email and sync documents after a client is updated (using the mock repositories provided)
3. Search for a client
 - a. Searching in the "search field" should filter the list of clients by their first or last name
 - b. Example: John Stevens and Steven Smith should both show up if a user searches "steven"
 - c. Example: John Stevens should show up if a user searches "john"

Actual Work

- Updated program.cs to show new controllers in the api when running.
- Updated launchSettings.json to auto launch swagger
- Finished item 1 and 2 by adding a scaffolded controller
 - Updated the scaffolded controller to accept form body
- Finished item 3
 - Added a field "Name" in GET/api/Clients

Extras

- Quality and best practices
 - Do not put business logic in the controllers.
 - You can put them in another class like a service.
 - For this exercise I just used the controller to also have the business logic.
 - Do not put logic in repositories.
 - You can put them in another class like a service.
 - The repositories should only be doing data manipulation for this exercise.
 - Using message queuing to replace email sending and syncing the documents.
- Can this submission's code architecture easily scale to a codebase with 20 developers?
 - Yes it can with proper pull requests. But the architecture can be better by doing segregation of codes and putting them in separate projects.
- How can you ensure data integrity in case of failures?
 -
- How can you ensure the API behaves as you intend it to?
 -
- How can you improve the performance of this?
 - Feel free to mock any external infrastructure as in-memory if you need.

Additional (Not in requirements and extras)

- Included Delete in Api since this was with the scaffolded controller.

Notes

I have finished this in around 3-4 hours. I initially was trying to do the UI part first as I know I will be able to finish this API exercise quickly. For the UI since it is in react and I have no actual experience in it, I will still try to do my best to deliver what I can.