



## Implementation Candidate Coding Challenge

### Objective:

We created this test to give candidates an opportunity to work with Emma's public API and some common customer requests. We have provided some documentation links, public API keys, and a list of tasks along with a few hints to help you get started.

You will be troubleshooting and updating a custom form. As a bonus you can provide some insight into your design eye and

### Documentation:

[Emma API Documentation](#)

[Emma Postman Collection](#)

[Tailwind Documentation](#)

### Notes about the public API:

You will mostly be focussed on the [Members API](#) for this task. The Postman Collection should have everything you need for testing and troubleshooting API calls. To use the Postman Collection use the API Keys and Basic Authentication to make calls.

### Before you start:

- The email address receiving this message should have received a user invite to Emma's SFTP server, Exavault. The sender email address is email@exavault.com. This email has the subject **ExaVault File Sharing Account**. That email will include the SFTP host information & username for your SFTP account. It will also have a link for you to set your password. After you set your password make sure you save it and please log into your SFTP account.
- On the SFTP server will be a file **key.txt** which will contain your API keys.
- Fork and clone the form repository [Form Test](#) and do your work there.
- Add your API keys from Exavault to the **/form-test/api/e2ma-keys.php** file.

## Saving and submitting your work:

- **MAKE SURE YOU DO NOT PUSH THE `/form-test/api/e2ma-keys.php` TO YOUR REMOTE REPOSITORY.** The `.gitignore` file should be configured to handle this.
- Push each update completing the 4 requests below in their own commit.
- In the end you will submit the URL to your github repository containing your work, so **COMMIT MESSAGES AND ORGANIZATION ARE IMPORTANT.**
- Add screenshots for each request update to the `/screenshots` directory proving your work. This is a commonly used communication method with customers.

## Requests:

### 1. My form is not updating member fields in my account!

- You can use the [POST /#account\\_id/members/add](#) endpoint to create a new member. This will return the **member\_id** and **email** you need to test the form.
- You can also get that information using the [GET /#account\\_id/members](#) endpoint.
- To troubleshoot to see if the member is updating you can use the [GET /#account\\_id/members/#member\\_id](#) endpoint.

### 2. I need to update subscription options on my form!

- You will need to update the field [PUT /#account\\_id/fields/#field\\_id](#) and update the form with the new field option.
- Make the update option your **First Name & Last Name**.
- Confirm the updated option updates on a member record.

### 3. I need to add a new field to my account & form!

- You can use the [POST /#account\\_id/fields](#) endpoint to create a new field. This returns a **field\_id** and [GET /#account\\_id/fields/#field\\_id](#) gives details.
- You can reference other fields in the form to understand the format fields.
- Make the new field your **First Name & Last Name** with shortcut **<first\_name>-<last\_name>**.
- To troubleshoot you should make sure the field exists and a member is able to update that field value on their member record.
  - i. [GET /#account\\_id/fields/#field\\_id](#)
  - ii. [GET /#account\\_id/members/#member\\_id](#)

#### 4. The opt-out button on my form is not working properly!

- There are several endpoints in [Members API](#) that will allow you to set a member status to opt-out. Reference the form and the API documentation to troubleshoot this button on the form.

#### 5. BONUS : Design the form after your favorite website

- Provide the URL to the site you are duplicating the styles from.
- Consider using TailWindCSS (included in the form) to do your styling.
- Reformat code if you have any opinions on how it can better work.