View Models

To better understand creating and updating a subscriber, we need to take a closer look at view models. As I said in the introduction, these classes contain data that the frontend needs. The only place they used in the backend is the controller:

```php
namespace App\Http\Web\Controllers\Subscriber;

use Inertia\Response;

class SubscriberController
{
    public function create(): Response
    {
        return Inertia::render('Subscriber/Form', [
            'model' => new UpsertSubscriberViewModel(),
        ]);
    }

    public function edit(Subscriber $subscriber): Response
    {
        return Inertia::render('Subscriber/Form', [
            'model' => new UpsertSubscriberViewModel($subscriber),
        ]);
    }
}
```

If you remember from the good old Blade days, `create` and `edit` renders the create and edit page for a subscriber. But instead of rendering a Blade view, they are returning an `Inertia\Response` instance. This is the only difference when using inertia. The `Subscriber/Form` is the path of a Vue component that contains the subscriber form. The