라라벨 API 서비스 아키텍처 구성 경험기

안정수
@findstar

목차

5. Response
    - presentation
        - 결과의 일부를 hidng or calculating
        - scheme 변경에 영향을 받지 않는다.
        - API Spec 의 일관성을 유지하기 위해서

6. 기타
    - 인증에는 JWT + Multi Auth 를 활용(JWT 에는 custom 한 field 정보를 추가할 수 있어 이를 통해서 db query 없이도 authorizing 가능)
    - app/exception/handler 의 render/report 잘 활용하면 유용합니다.(report 는 sentry.io가 좋더라.)
    - form request 의 failedValidation 메소드를 override 하면 원하는 형태로 validation fail 처리 가능
    - 기타 참고한 사항들.

Route 정의

- 모든 Route 는 Controller 에 바인딩
- Route:cache 활용을 위해서 closure 사용 X
- Controller Method Injection 을 활용한다.

FormRequest 의 활용

- 매뉴얼

Form Request 가 처리해주는 것.

- request validation

```
public function rules()
{
    return [
        'title' => 'required|unique:posts|max:255',
        'body' => 'required',
    ];
}
```

- authorizing

```
public function authorize()
{
    $commentId = $this->route('comment');

    return Comment::where('id', $commentId)
                ->where('user_id', Auth::id())->exists();
}
```

- make DTO : 내부에서 사용할 전용 DTO 객체 생성

```
public function getProductDTO()
{
    return new ProductDTO(
        $this->get('productName'),
        $this->get('productOwner'),
        $this->get('productType', Type::ProductA),
        $this->get('productQuantity', 1)
    );
}
```

Service 객체

- Method Injectoin 을 통해서 Controller 에 주입
- 비지니스 로직을 처리
- 내부에서 바로 Eloquent ORM 을 통해서 Model 객체 조회 및 사용
- NonRepository Pattern

네이밍

- 단순한 목적 단위의 네이밍 X (ex UsersHandler, UserManager..)
- 목적 + 액션 단위의 네이밍 (ex UserRetriever, UserModifier, UserCreator...)

Response(Presentation)

- JsonMapper, Fratal 를 활용한 Json 타입의 변환.

JWT

- Json Web Token
- 토큰 안에 정보를 담을 수 있어서 좋음/토큰의 사이즈 늘어남
- https://github.com/tymondesigns/jwt-auth 이게 제일 쓸만함, 필요한건 다 있었음

참고

- 조대협님의 JWT
- Outsider 님의 JWT