Documentation

OpenAPI and Swagger

Exercises

- Remove all optionals from user and transaction endpoints
- Remove all try catches from users and transactions controllers

What Is OpenAPI?

OpenAPI Specification (formerly Swagger Specification) is an API description format for REST APIs. An OpenAPI file allows you to describe your entire API, including:

- Available endpoints (`/users`) and operations on each endpoint (`GET /users`, `POST /users`)
- Operation parameters Input and output for each operation
- Authentication methods
- Contact information, license, terms of use and other information.

API specifications can be written in YAML or JSON.

http://editor.swagger.io/

What Is Swagger?

Swagger is a set of open-source tools built around the OpenAPI Specification that can help you design, build, document and consume REST APIs. The major Swagger tools include:

- Swagger Editor – browser-based editor where you can write OpenAPI specs.
- Swagger UI – renders OpenAPI specs as interactive API documentation.
- Swagger Codegen – generates server stubs and client libraries from an OpenAPI spec.

Maven Springfox Swagger

```
<dependency>
        <groupId>io.springfox</groupId>
        <artifactId>springfox-swagger2</artifactId>
        <version>2.9.2</version>
</dependency>
```

Enable springfox swagger with default configuration

```
@SpringBootApplication
@EnableSwagger2
public class MoneyTransferAppApplication {

   public static void main(String[] args) {SpringApplication.run(MoneyTransferAppApplication.class, args);}
   @Bean
   public Docket api() {
       return new Docket(DocumentationType.SWAGGER_2)
               .select()
               .apis(RequestHandlerSelectors.any())
               .paths(PathSelectors.any())
               .build();
   }
}
```

Verification

http://localhost:8080/v2/api-docs

Swagger UI

Swagger UI is a built-in solution which makes user interaction with the Swagger-generated API documentation much easier.

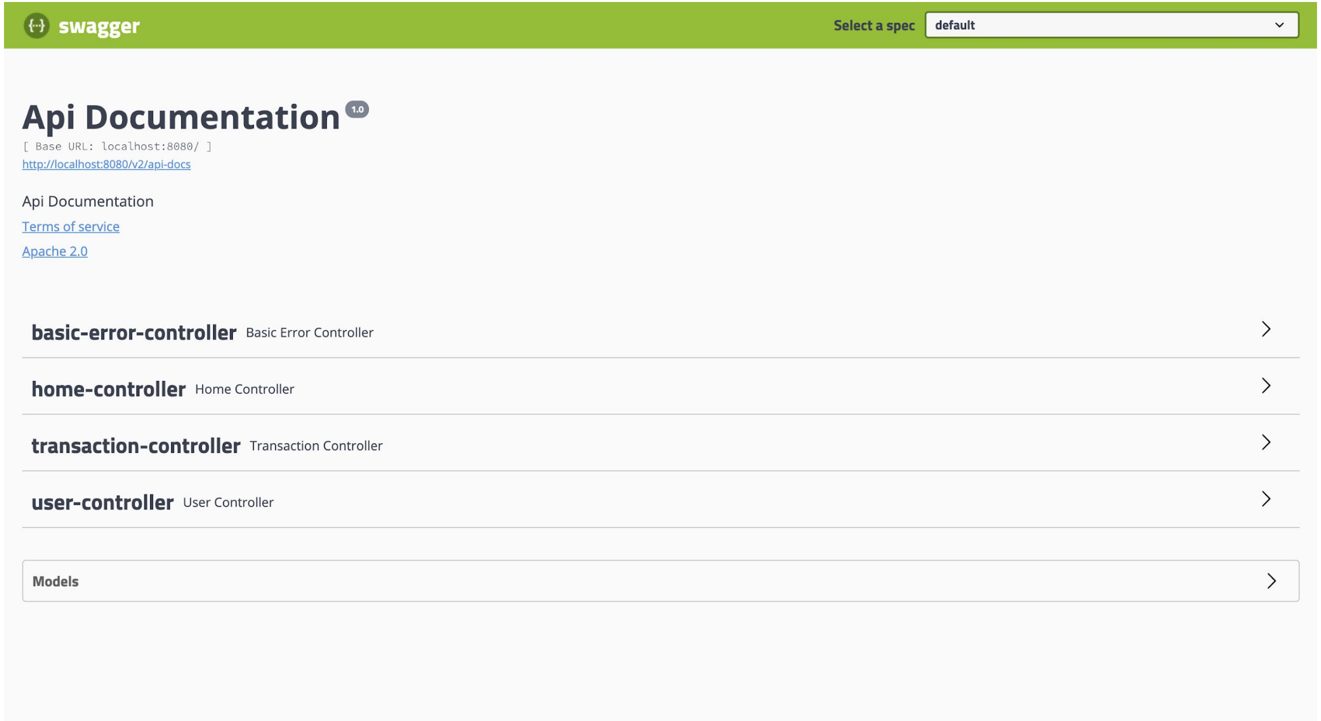
Enabling Springfox's Swagger UI

```xml
<dependency>
  <groupId>io.springfox</groupId>
  <artifactId>springfox-swagger-ui</artifactId>
  <version>2.9.2</version>
</dependency>
```

Verification

http://localhost:8080/swagger-ui.html

Swagger UI View



Api Documentation 1.0

[Base URL: localhost:8080/]

http://localhost:8080/v2/api-docs

Api Documentation

Terms of service

Apache 2.0

basic-error-controller Basic Error Controller

home-controller Home Controller

transaction-controller Transaction Controller

user-controller User Controller

Models

Exploring Swagger Documentation

```
@RestController
public class CustomController {

   @RequestMapping(value = "/custom", method =
RequestMethod.POST)
   public String custom() {
      return "custom";
   }
}
```

Advanced Configuration

The Docket bean of the application can be configured to give more control over the API documentation generation process.

Advanced Config: RequestHandlerSelectors

RequestHandlerSelectors allows using the **any** or **none** predicates, but can also be used to filter the API according to the **base package**, **class annotation**, and **method annotations**.

Advanced Config: RequestHandlerSelectors

```
@Bean
public Docket api() {
  return new Docket(DocumentationType.SWAGGER_2)
          .select()
          .apis(RequestHandlerSelectors.withClassAnnotation(Api.class))
          .paths(PathSelectors.any())
          .build();
}
------------------------------------------------------------------------------------------

@Api
public class UserController {
```

Advanced Config: @ApiOperation

```
@ApiOperation(value = "find user by id")
@GetMapping("/{id}")
public ResponseEntity<User> findOne(@PathVariable Long id) {
   User user = userRepository.findById(id).get();
   return new ResponseEntity<>(user, HttpStatus.OK);
}
```

Advanced Config: RequestHandlerSelectors.basePackage

```
@Bean
public Docket api() {
   return new Docket(DocumentationType.SWAGGER_2)
           .select()
           .apis(RequestHandlerSelectors.
basePackage("com.nursultanturdaliev.moneytransferapp.controller"))
           .paths(PathSelectors.any())
           .build();
}
```

Advanced Config: RequestHandlerSelectors.withMethodAnnotation

```
@Bean
public Docket api() {
  return new Docket(DocumentationType.SWAGGER_2)
        .select()

.apis(RequestHandlerSelectors.withMethodAnnotation(ApiOperation.class))
        .paths(PathSelectors.any())
        .build();
}
```

PathSelectors

PathSelectors provides additional filtering with predicates which scan the request paths of your application. You can use **any(), none(), regex(),** or **ant()**.

Custom Information ApiInfo

```
private ApiInfo apiInfo() {
   return new ApiInfo(
           "MONEY TRANSFER APP REST API",
           "This is the main rest api documentation.",
           "API V1.0.0",
           "Terms of service",
           new Contact("Nursultan Turdaliev", "www.kapusta.com",
"contact@kapusta.com"),
           "GPL 3.0", "https://www.gnu.org/licenses/gpl-3.0.en.html",
Collections.emptyList());
}
```

Custom Methods Response Messages

Swagger allows **globally overriding response messages of HTTP** methods through Docket’s **globalResponseMessage()** method.

Custom Methods Response Messages

Swagger allows **globally overriding response messages of HTTP** methods through Docket’s **globalResponseMessage()** method.

Globally Overriding Response Messages of HTTP Methods

```
@Bean
public Docket api() {
  return new Docket(DocumentationType.SWAGGER_2)
        .globalResponseMessage(RequestMethod.GET,
        newArrayList(new ResponseMessageBuilder()
                  .code(200)
                  .message("successful request")
                  .build(),
             new ResponseMessageBuilder()
                  .code(404)
                  .message("resource not found")
                  .build()))
```

Exercises

- Using PathSelector enable API documentation for all endpoints starting with /api/
- Change response messages of all POST, DELETE messages

References

- https://spring.io/guides/gs/testing-restdocs/
- https://swagger.io/
- https://www.baeldung.com/swagger-2-documentation-for-spring-rest-api
- https://www.vojtechruzicka.com/documenting-spring-boot-rest-api-swagger-springfox/
- https://dzone.com/articles/how-to-automatically-document-api-endpoints-via-sw
- https://swagger.io