

# Assignment 3: API specs and Review Presentation

Work in your group to write a description for a simple API. *You do not need to write the API itself.* Then, as individuals, use OpenAPI and Swagger to write interactive documentation for the API your group described. Then, as individuals, create a short video of your API documentation where you explain each line of your OAS document to a future COMP 3040 student.

---

## *Assignment Goals:*

### *Concepts:*

- Use GitHub to share and modify files within a group
- Understand that a API documentation—the informal description followed by a (very formal) specification (OAS) file—are the first steps in the API product lifecycle
- Learn about interactive API documentation tools such as Swagger and/or Spotlight
- Understand the need for specification documents written in OpenAPI (OAS)
- Acquaint yourself with JSON and YAML syntax (within OAS)

### *Skills:*

- *Work in a group to understand APIs and to think of a small and simple API*
  - *Work as an individual to learn and apply the OAS 3.0 to create a specification document for a not-yet-written API*
  - *Create a YAML file (to store your OAS/specification file)*
  - *Use screen recording software to explain a new concept to a given audience*
-

## *Part 1: Work as a group to write a description for a simple and short API about a topic related to Manitoba*

Overview: Work as a group to come up with an idea for a simple API. Use Github and Markdown to collectively author a description of the API to a general audience.

### General Requirements:

- The topic of your API *must be related to something about our province of Manitoba*
- Describe the API in a Markdown file. Format and describe the API as though you are proposing it to a management team
- The documentation headings must include:
  - API description
  - List of endpoints with parameters (see below for requirements)
  - Description of resources—formatted as JSON
  - Sample request with Sample response
- Use GitHub to share and revise this API description. Each group member should have at least one commit.
- This file will be quite short. I expect around 200-300 words.

### Specific Requirements:

Your API should describe:

- Resources: one or two resources
- Endpoints and Parameters: Describe 1-3 endpoints. If you have one endpoint, include at least 3 parameters. If you have two endpoints, include at least 2 parameters. If you have 3 endpoints, include at least 1 parameter.
  - Note: If you want to keep it simple, these endpoints could all be GET methods.

### Schedule

- Thursday, November 19: We will devote class time to working in groups to write Part 1
- Tuesday, November 24: Submit Part 1. I will give you feedback by Wednesday, Nov 25 at 11:59PM

### Submission Requirements:

Save your general description in a .md file using this format: GroupXX\_v[x].A3.md where XX is your group number. Also, in the text box of the assignment you will paste 1) a link to the Github repo that shows the history of this file, AND 2) the GitHub user names of each group member. . Remember that each group member should make a commit.

## *Part 2: Work as an individual to create an OAS specification file for the API your group described.*

Overview: Transform your group's general API description into an OAS specification document for a technical audience, to begin the API product development process.

### General Requirements:

1. Use OpenAPI (OAS) version 3.0
  - o Hint: Consider using the Simple API Template. Or you can write it without a template.
2. Use the Swagger Editor, or, more likely, SwaggerHub. SwaggerHub (which contains the Editor) has a free version that lets you save your files in SwaggerHub.

### Details:

- Use `openapi 3.0`.
- Make up some server names.
  - o Since you do not have to write the actual API your sample calls will not work in the documentation.
- INFO root object:
  - o Specify the `version`, and `title` of your imaginary API.
  - o `description`:
    - Describe your API. You may copy and paste sentences from your group's general description.
  - o You do not need to specify a license.
- Tags root object: Add at least one tag.
- Paths root object:
  - o Describe 1-3 endpoints. For each endpoint, describe these properties: `Summary`, `Description`, `Responses`.  
NOTE: `Summary` and `Descriptions`: You do not need to use CommonMark. You also do not need to write in complete sentences if you think a briefer description is better.
  - o Parameters: If you have one endpoint, include at least 3 parameters. If you have two endpoints, include at least 2 parameters. If you have 3 endpoints, include at least 1 parameter.
    - Include `name`, `in`, `description`, `required`, `schema` and `example` for each parameter.
  - o There is a considerable learning curve with OAS, especially the path object. You might find this site helpful:  
[https://idratherbewriting.com/learnapidoc/pubapis\\_openapi\\_step4\\_paths\\_object.html](https://idratherbewriting.com/learnapidoc/pubapis_openapi_step4_paths_object.html)

- Components:
  - Organize your responses using components.
    - Note: Although components are not very practical in such a short OAS, they are invaluable for most OAS, and a significant improvement of OAS 3.0.
    - More on Responses:
      - For a '200' response (the expected response): reference the array to be returned as a components/schemas object. For example:

```
responses:
  '200':
    description: Add a description.
    content:
      application/json:
        schema:
          type: array
          items:
            $ref: '#/components/schemas/Country'
```

- Modify the "400" response so it makes sense for the given method.

## Submission Requirements:

Export file as an UNRESOLVED YAML file and name it using this naming structure:

[lastName\_firstName]\_A3\_v[x].yaml

*Part 3: Work as an individual to create a short (2-4 minute) video file that explains your OAS file to future COMP 3040 students.*

### General Requirements:

1. Use screen recording software to create a 2-4 minute video. Aim for around 3 minutes.
2. You may use voice-over or on-screen captions.
3. Audience? Next year's COMP 3040 students.
4. Goals? Help future COMP 3040 students understand OAS. A secondary goal is to demonstrate to the graders and myself that you wrote the OAS file yourself and you understand OAS.
5. Use ABC format.
6. Note: There are no grades for video quality. I understand that fine-tuning a video can be very time-consuming and you are busy. Use the video to show me what you've learned.

### Details:

Include this content:

- Abstract:
  - Begin by describing the purpose of the video—to help others learn OAS by providing an annotated example of documentation for a simple API.
  - Describe the API and its purpose.
- Body:
  - Describe each line of the OAS file. If you have similar sections (like multiple get methods), you can describe one in detail and then describe how the remaining methods differ. Don't spend much time on the easy-to-understand objects, like info and servers.
  - As you describe the OAS file, annotate your description with details of your own personal learning journey; describe what you found challenging and how you figured out your problems.
- Conclusion:
  - Offer general advice and resources on how to learn OAS.
  - Credit your group members—who helped you come up with the API idea—in either the Abstract or Conclusion.

### Submission Requirements:

Save the file as an mp4 file using this naming structure: [lastName\_firstName]\_A3\_v[x].mp4

## *Schedule and Deadlines*

Since this is a complex assignment, we will use a scaffolded approach of working sessions and interim deadlines, as follow:

1. *Thursday, November 19*: We will devote class time to working in groups to write Part 1.
2. *Tuesday, November 24*: Submit Part 1. I will give you feedback by Wednesday, Nov 25 at 11:59PM.
3. *Tuesday, December 1*: We will devote the entire class on Tuesday, December 1 to troubleshooting (in groups) problems and questions with OAS files. To succeed in this assignment, you will need to have a mostly-complete OAS file at the start of this class. You can use this class to fix errors and help other group members get a working OAS file.
4. Dec 1-December 8: Complete Part 3 (an individual video explanation of your OAS file).
5. *Tuesday, December 8 (11:59pm)*: All three parts of the assignment are due.

## *Assignment 3 Grading Guide*

### *Part 1: Group General Description of API (2)*

Fulfills these requirements; If not deduct in .5 increments, with maximum deductions as shown.

- Each group member made a commit to file (1)
- Content:(1)
  - API clearly described
  - Suitable for a general audience
  - Endpoints, parameters and resources clearly described

### *Part 2: Individual OAS file (4)*

Fulfills these requirements; If not deduct in .5 increments, with maximum deductions as shown.

- Free from errors (1)
- FOUR endpoints and/or parameters.
  - Endpoints require: Summary, Description, Responses (1)
  - Parameters require: name, in, description, required, schema and example (1)
- Uses components (1)
- 200 and 400 return types successfully described (1)
- Displays as expected in Swagger Preview (1)
- Syntax is easy-to-understand and any errors do not detract from meaning or purpose (1)

### *Part 3: Individual video tutorial (4)*

Achieves these goals. If not deduct in .5 – 1, with maximum deductions as shown:

- Shows student understands OAS (2)
- Well-organized and uses ABC format (1)
  - –1/2 if over 4 minutes long
- Includes personal learning reflections and advice (1)

### *Additional Deductions:*

- File names not using standard or types specified (-1)
- Submission between 1 and 24 hours late (-1) ; between 24-48 hours late (-2);
  - note: after 48 hours the assignment folder will not accept assignments.
- Video cannot be viewed in VLC (-1). Student will be given option to re-record presentation and submit.

*Grade:    /10*