Università
della
Svizzera
italiana

**Software
Institute**

SOFTWARE & DATA ENGINEERING MASTER THESIS PROPOSAL

Edoardo Riggio

API SCOUT: AN INFORMATION RETRIEVAL SYSTEM FOR OPENAPI SPECIFICATIONS

Abstract

The primary objective of this thesis is to create an information retrieval system enabling users to explore a compre hensive collection of OpenAPI specifications. The system will use historical versioned data - scraped from existing github repositories - as well as user-uploaded data.

This platform serves a dual purpose, catering to both academics and developers. In the case of academics, this platf orm can serve as a repository from which to conduct research, facilitating more in-depth studies on API specificatio ns and their evolution. On the other hand, developers can look up several different examples that could help them get started with their own OpenAPI documentation.

The main contributions of this thesis include two crucial aspects: data engineering and software engineering. In the first case, the thesis will delve in the process of refining raw data and finding ways to classify the API specifications. In the second case, the thesis will define the architectural design and the development of the service itself.

Contents

List of Figures

List of Tables

1 Introduction

In the ever-evolving landscape of software engineering, APIs play a pivotal role in facilitating the communication and interoperability between software services. Being software systems intrinsically complex, it is good practice to keep track of the interfaces that they expose. Moreover, especially in the cases where these interfaces have to be used by external developers, it is paramount to have good documentation for such interfaces.

In this thesis, we will work with a specific type of API: HTTP API. Such APIs are exposed by servers and accessible though HTTP requests. To properly document public HTTP APIs, Swagger proposed in 2011 the "Swagger Specification" (today known as "OpenAPI Specification"). As Swagger explains on its website, "[t]he OpenAPI Specification (OAS) defines a standard, language-agnostic interface to HTTP APIs which allows both humans and computers to discover and understand the capabilities of the service [...]." [1]

Every service with public endpoints, should expose – as per Swagger's specifications – a `openapi.json` or `openapi.yaml` file. This file can be one, or it can be the root of several other smaller files – at the discretion of the author. Moreover, each document that specifies the endpoints' structure must be written following a specific set of rules. Such rules are defined and explained on Swagger's "OpenAPI Specification" website.

1.1 Motivation

From our research, we noticed that there is a void in tools and services that can be helpful in the study of the evolution and current state of HTTP APIs. Although platforms such as SwaggerHub[1] and APIs.guru[2] already exist, they lack features that are essential for research.

For example, they lack a proper search system. In the previously mentioned platforms, indexing of specifications is superficial and most of the time only the title is matched with the query. With API Scout, we want to index API specifications based not only on their title, but based on all the tags containing natural language.

Another feature missing in the aforementioned platforms is a proper visualization system. By using tools to convert specifications into interactive trees, users can better understand the structure of the service's endpoints. Moreover, we can also plot some statistics about the selected API and how it compares to all the specifications present on our platform.

1.2 Objective

The goal of this thesis is to build a platform that both academics and developers can use to understand and study the intricacies of real-world HTTP APIs.

In the case of the developers, they can search and consult our historical and current collection of APIs to better understand how to structure one. Aided by the graphical tools, a developer can also understand how the endpoints should be structured to obtain a more cohesive structure.

[1] https://app.swaggerhub.com/search
[2] https://apis.guru/

2 Proposal

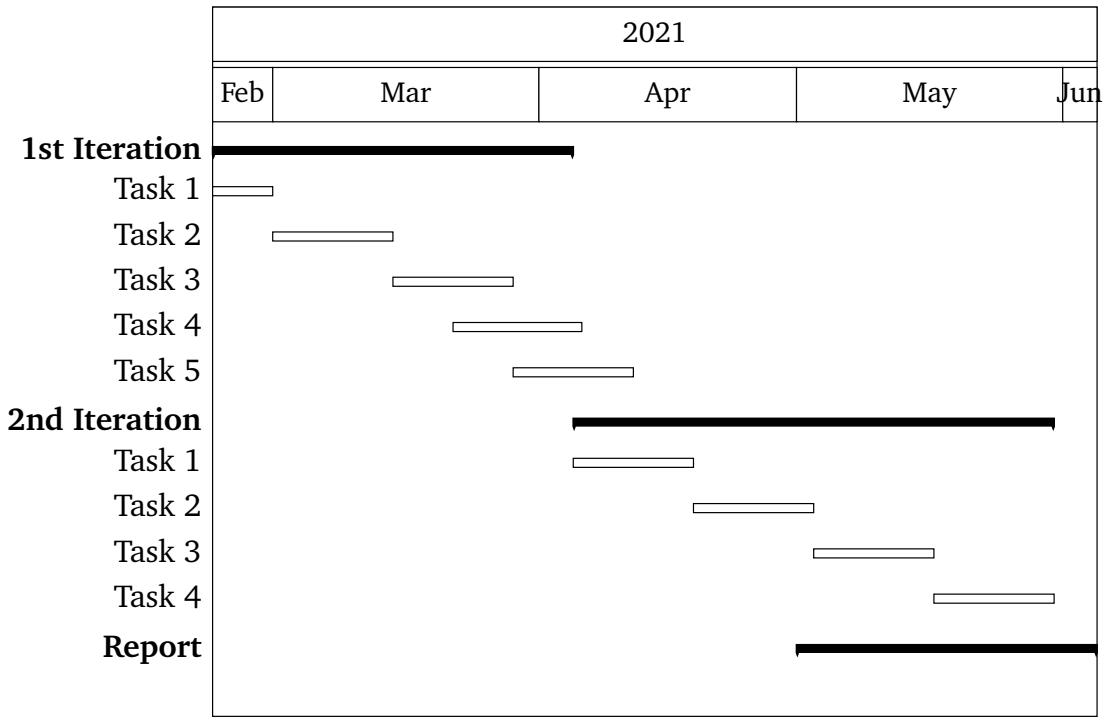
3 State of the Art

4 Preliminary Results - Feasibility Study

5 Work Plan for the Spring semester

This section describes a high level view of the work plan for the spring semester.

- Iteration 1: Data Analysis Phase
  - Task 1: Setup project (Pick tools/technologies to use)
  - Task 2: Design of metrics to locate changes in evolution
  - Task 3: Setup database of metrics
  - Task 4: Classification of changes
  - Task 5: Evaluate and improve system
- Iteration 2: Visualization Phase
  - Task 1: High level visualization of API
  - Task 2: Drilled down visualization of API
  - Task 3: Visualizing changes
  - Task 4: Evaluate and improve system
- Iteration 3: Report



References

[1] Swagger. Openapi specification.