API for Point of Contact for Collaboration Framework

Introduction

Software Company 1701 Delta (aka SC 1701-D) has experienced tremendous growth, doubling over the past several years. This growth has made collaboration more difficult due to the disconnect between different teams and organizations, within the company. The challenge of finding support from another team is beginning to impact productivity, increasing risk of releasing features on time. A mechanism is needed to solve the problem of finding a person of contact for a product.

Example Use Case:

Veena, and employee of SC 1701-D, is working on a feature that identifies potential security vulnerabilities of an application. There is another team within SC 1701-D who are in the process of creating an API that will run security scans and return a result containing the security vulnerabilities, however Veena works in Ashville, SC office and the team who create the security scan API is located in the Lucerne, CH office. She has never collaborated with anyone on the security scan team and is having a difficult time tracking a point-of-contact.

Task 1: Design and document the data model and API that will query a point-of-contact for a given product

During the Design stage of the application life cycle, a blueprint is needed to map out key parts of an application and determine how all the pieces will come together to create a product. Documentation is necessary to communicate how the individual parts work.

Deliverables:

- Create a high-level data model for all tables that will be used by the API.
- Create a document that will describe what the API is designed to do and it's usage. Keep in mind your audience will be other engineers.

Task 2: Create a proof of concept for the API

The proof of concept for the API can be written in any programming language. The API should handle happy path as well as basic error handling.

Given the use case described earlier, Veena should be able to query for a point of contact by product name or repository name. The requirements for the API are:

- Valid inputs used to query:
  - Product name
  - Repository name
- Given the use case described earlier, Veena should be able to query for a point of contact by product name or github repo.
- The minimal return value should be an object containing:
  - First name
  - Last name
  - Email
  - Chat user name
  - Location
  - Title or role
- Error handling

Deliverables:

- Link to functional proof of concept
- Access to code with instructions on how to install, build, run, etc… the API