https://killeralex.github.io/cvrus/

Тестовое задание на должность «специалист по разработке технической документации»

Задача: необходимо описать интеграцию между 2 системами 1С и Mind&Machine.
Описание:
Из системы 1С в систему Mind&Machine передается информация где, какие сотрудники и когда работали. В системе 1С есть следующие модели:

1. *Физическое лицо*;
2. *Сотрудник* – физ. лицо с определенным табельным номером (может быть несколько у одного физ. лица);
3. *Кадровый перевод* – изменение в работе сотрудника (Пример: назначен на новую должность, назначен на работу в новом магазине, изменены другие параметры работы).

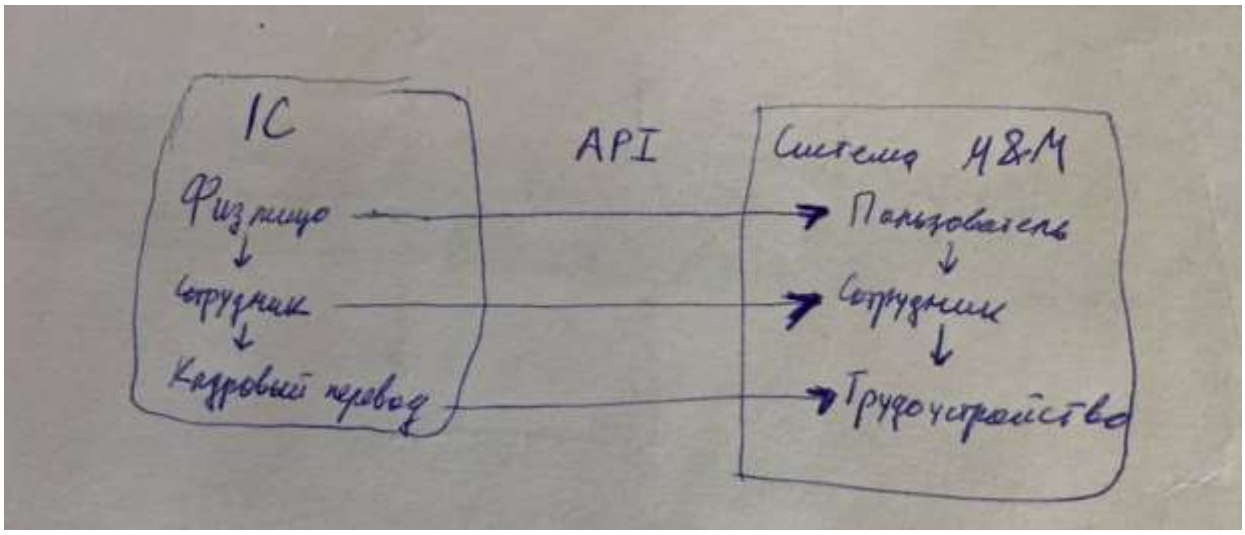
В системе Mind&Machine есть следующие модели:

1. *Пользователь системы*;
2. *Сотрудник* – физ. лицо с определенным табельным номером;
3. *Трудоустройство* – отображает где, кем и когда работал сотрудник (Например: с 20.01.2020 по 20.09.20 сотрудник А работал в магазине Б на должности В).

Для каждой модели есть следующие 3 API метода:

- Получить список элементов модели;
- Создать или изменить элемент модели;
- Удалить элемент модели.

Подготавливаемый документ нужен как для внутреннего пользования, так и для внешней организации.



Ожидаемый результат:

1. Описание интеграции на 2-3 страницы (300-600 слов);
2. Недостающую информацию необходимо придумать самостоятельно, либо указать, что по данному аспекту требуется уточнение;
3. Описание API методов приводить не нужно (будет достаточно указать примеры url);

Описание интеграции информационных систем на базе 1С и решения от Mind&Machine

1 Обоснование архитектуры

В данном документе представлено описание интеграции информационных систем (ИС) на базе 1С и на базе решения от Mind&Machine. На начальном этапе из первой ИС (1С) в систему Mind&Machine передается информация о том, где, какие сотрудники и когда работали. Таким образом, достаточно реализовать односторонний поток данных из ИС1 в ИС2.

Для того, чтобы сделать подход к интеграции двух систем более универсальным и однотипным, предлагается:

1) реализовать универсальные методы чтения и создания новых моделей для обеих систем;
2) на первом этапе ограничить информационное взаимодействие только двумя ИС, в т.ч. не создавать общие НСИ (словари) или другие ИС.

Описание моделей в обеих ИС представлено в табл. 1, 2.

Таблица 1 – Описание моделей в ИС на базе 1С

№	Модель	Описание
1	Физическое лицо	гражданин РФ или зарегистрированный гражданин иностранного государства с разрешением на пребывание на территории РФ (далее – гость), однозначно идентифицируемый в едином реестре граждан РФ и гостей по уникальному идентификатору текущая реализация – номер и серия паспорта
2	Сотрудник	физическое лицо с определенным табельным номером (может быть несколько у одного физ. лица); текущая реализация – физическое лицо может выполнять не более 3х ролей сотрудников, т.е. иметь не более 3х полей ждя указания табельного номера
3	Кадровый перевод	изменение в работе сотрудника (пример: назначен на новую должность, назначен на работу в новом магазине, изменены другие параметры работы) текущая реализация – передаётся текущий статус и история изменений в виде uid приказов по организации

Таблица 2 – Описание моделей в ИС на базе решения от Mind&Machine

№	Модель	Описание
1	Пользователь системы	соответствует п. 1 табл. 1
2	Сотрудник	соответствует п. 2 табл. 1
3	Трудоустройство	`hrstatus`, см. рис. 2

Таблица 3 – описание методов API

№	Метод	Описание
1	Получить список элементов	GET (доступ только по чтению)
2	Создать или изменить элемент	POST (создать новый элемент) PUT (редактировать существующий или создать новый элемент)
3	Удалить элемент	DELETE (удалить элемент) `DELETE /resource/1` (REST) `POST /deleteResource { "id": 1 }` (gRPC)

Допущение 1. API в общем случае не зависит от реализации ИС (монолит/MSA, одна или несколько независимых ИС).

Допущение 2. Рассмотренные ИС не содержат ссылки на общее хранилище персональных данных физических лиц или собственно такое хранилище. Поэтому передаваемые посредством API данные в определённом смысле избыточны. Имеется в виду, что вместо передачи, например, в JSON-файле структуры типа `individual: "Иванов Иван Иванович, паспорт 45 00 000001"` (см. рис. 2) достаточно было бы передавать только uid. То же касается данных о послужном списке сотрудника – можно было бы передавать id изменений в работе сотрудника из общего словаря, а также не передавать в ММ историю изменений.

Например (при реализации внешнего словаря НСИ в РСУБД):

uid (autoinc)	**Name**	**Вариант**
1	назначен на новую должность	1 и uid_должности
2	назначен на работу в новом магазине	2 и uid_магазина
3	изменены другие параметры работы	3 и номера параметров или новый список параметров

Примечание: uid, Name – поля в таблице НСИ в РСУБД

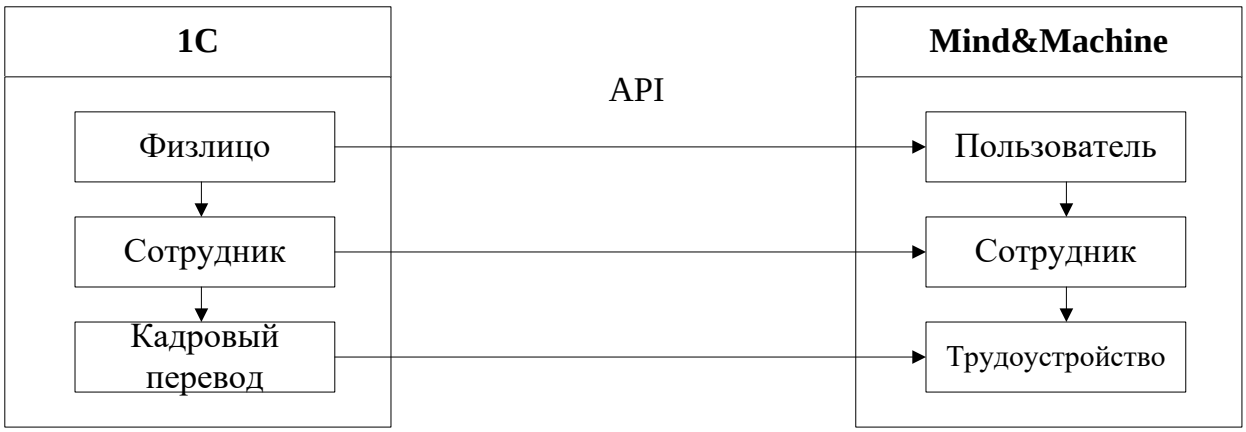


Рисунок 1 – Модели для интеграции ИС

2 Варианты реализации API

Примеры реализации REST API и gRPC API представлены на рисунках 2-6. На первом этапе предлагается реализовать REST API с сериализацией JSON. Конкретная реализация методов уточняется на этапе ЧТЗ. Рабочий вариант документируемого API приведён в приложении.

Здесь и далее записи вида (1), (2) являются примерами URL с точностью до имени ресурса[1].

GET /v1/uid – *вывести всех физических лиц* (1)
curl -v http://localhost:80/uid (2)

```
{
individual: "Иванов Иван Иванович, паспорт 45 00 000001",
employee1: "1480000001",
employee2: "none",
employee3: "none",
hrstatus: "назначен на новую должность, 10.01.2022, приказ ММ
HR/10012022",
hrtranslation: "ММ HR/10012022;ММ HR/01042021;EXT вх/159 01052019"
}
```

Рисунок 2 – Пример записи в формате JSON о сотруднике в ИС на базе 1С

```
{
uid: "4500000001",
firstname: "Иван",
secname:"Иванович",
lastname:"Иванов",
employee: [
"1480000001",
"none",
"none"
]
hrstatus: "назначен на новую должность, 10.01.2022, приказ ММ
HR/10012022",
hrtranslation: "ММ HR/10012022;ММ HR/01042021;EXT вх/159 01052019"
}
```

Рисунок 3 – Вариант записи о сотруднике в ИС на базе 1С

```
{
individual: "Иванов Иван Иванович, паспорт 45 00 000001",
employee1: "1480000001",
employee2: "none",
employee3: "none",
hrstatus: "назначен на новую должность, 10.01.2022, приказ ММ
HR/10012022"
}
```

Рисунок 4 – Вариант записи о пользователе в ИС на базе решения от Mind&Machine

[1] https://coderoad.ru/1537140/%D0%9A%D0%B0%D0%BA%D0%B8%D0%BC-%D0%B4%D0%BE%D0%BB%D0%B6%D0%B5%D0%BD-%D0%B1%D1%8B%D1%82%D1%8C-%D1%81%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82-%D0%B4%D0%BB%D1%8F-URL-%D0%B0%D0%B4%D1%80%D0%B5%D1%81%D0%BE%D0%B2-ReSTful

GET /v1/uid
GET /v1/uid/:id
GET /v1/uid/:id/`firstname`
GET /v1/uid/:id/`secname`
GET /v1/uid/:id/`lastname`
GET /v1/uid/:id/`employee`
GET /v1/uid/:id/`hrstatus`
GET /v1/uid/:id/`hrtranslation`

Рисунок 5 – Вариант реализации REST API (1C)

//Контекстная информация в конфигурационном файле

```
{
string uid = 1;
string first_name = 2;
string sec_name = 3;
string last_name = 4;
string employee1 = 5;
string employee2 = 6;
string employee3 = 7;
string hrstatus = 8;
}
```

//Сообщения в файлах *.proto (пример)

```
syntax = "proto3";
message Person {
  uint64 id = 1;
  string email = 2;
  bool is_active = 3;

 enum PhoneType {
    MOBILE = 0;
    HOME = 1;
    WORK = 2;
  }   message PhoneNumber {
    string number = 1;
    PhoneType type = 2;
   }

  repeated PhoneNumber phones = 4;}
```

Рисунок 6 – Вариант реализации gRPC API (RPC-JSON или Protobuf)

СПИСОК СОКРАЩЕНИЙ

ИС	–	информационная система
НСИ	–	нормативно-справочная информация
РСУБД	–	реляционная система управления базами данных
API	–	Application Programming Interface (программный интерфейс приложения)
gRPC	–	open-source Remote Procedure Call framework (каркас RPC производства Google)
JSON	–	JavaScript Object Notation (формат сериализации)
MSA	–	Microservices Architecture (микросервисная архитектура)
REST	–	Representational State Transfer (передача репрезентативного состояния)
uid	–	уникальный идентификатор
WFM	–	Workforce Management (управление рабочим временем персонала)

Приложение

Примеры обращения к конечным точкам в методах API
(с точностью до имени ресурса v1)

1С

GET /v1/uid – *вывести всех физических лиц*
GET /v1/uid/:id – *вывести всю информацию о конкретном физическом лице*
GET /v1/uid/:id/`firstname` – *здесь и далее вывести частный элемент информации о конкретном физическом лице (*)*
GET /v1/uid/:id/`secname` *(*)*
GET /v1/uid/:id/`lastname` *(*)*
GET /v1/uid/:id/`employee` *(*)*
GET /v1/uid/:id/`hrstatus` *(*)*
GET /v1/uid/:id/`hrtranslation` *(*)*
…
DELETE /v1/uid
DELETE /v1/uid/:id

POST

M&M

GET /v1/uid
GET /v1/uid/:id
GET /v1/uid/:id/`firstname`
GET /v1/uid/:id/`secname`
GET /v1/uid/:id/`lastname`
GET /v1/uid/:id/`employee`
GET /v1/uid/:id/`hrstatus`

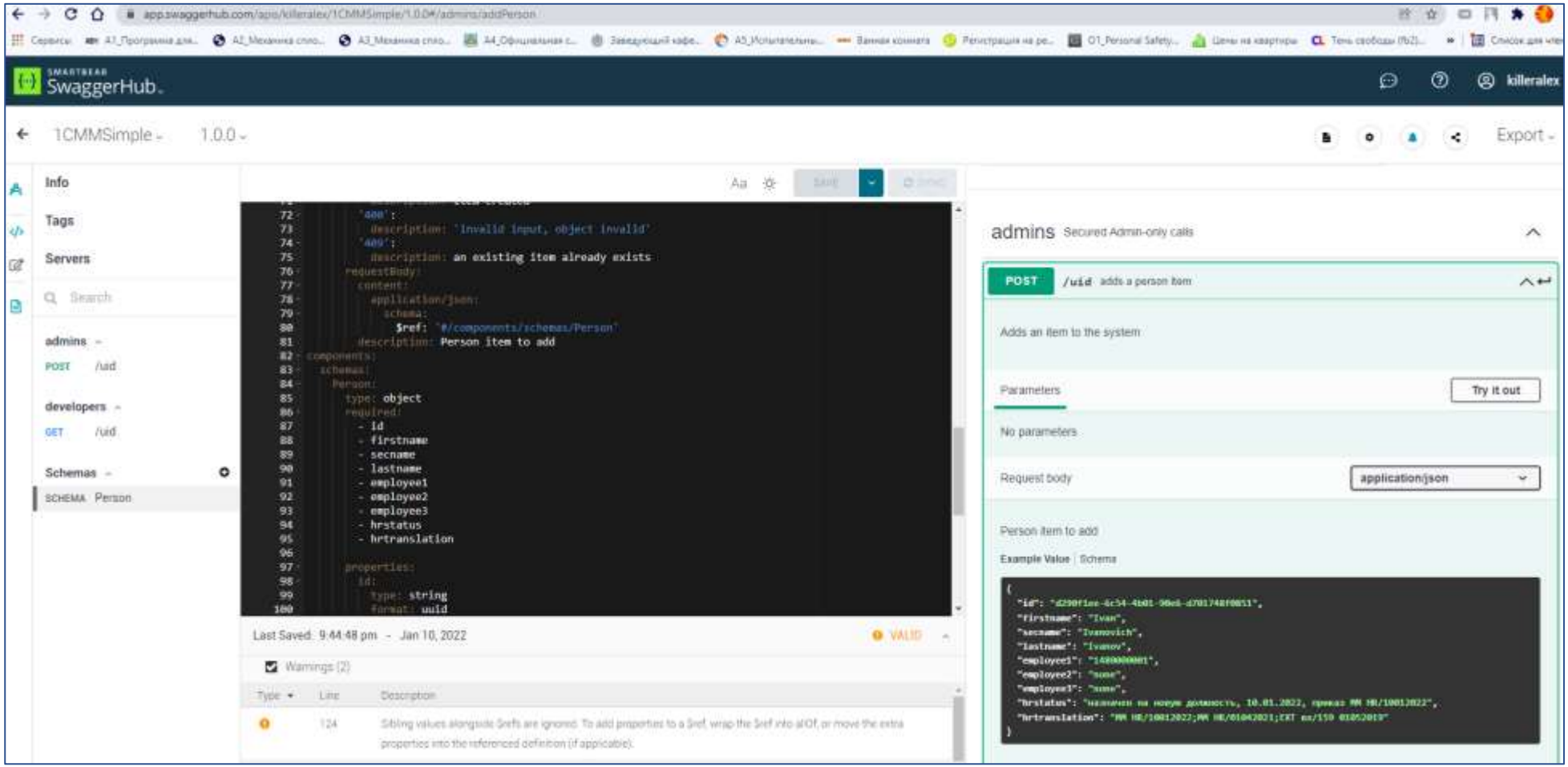


Рисунок П1 – Пример описания API в SwaggerHub (https://app.swaggerhub.com/apis/killeralex/1CMMSimple/1.0.0#/Person)

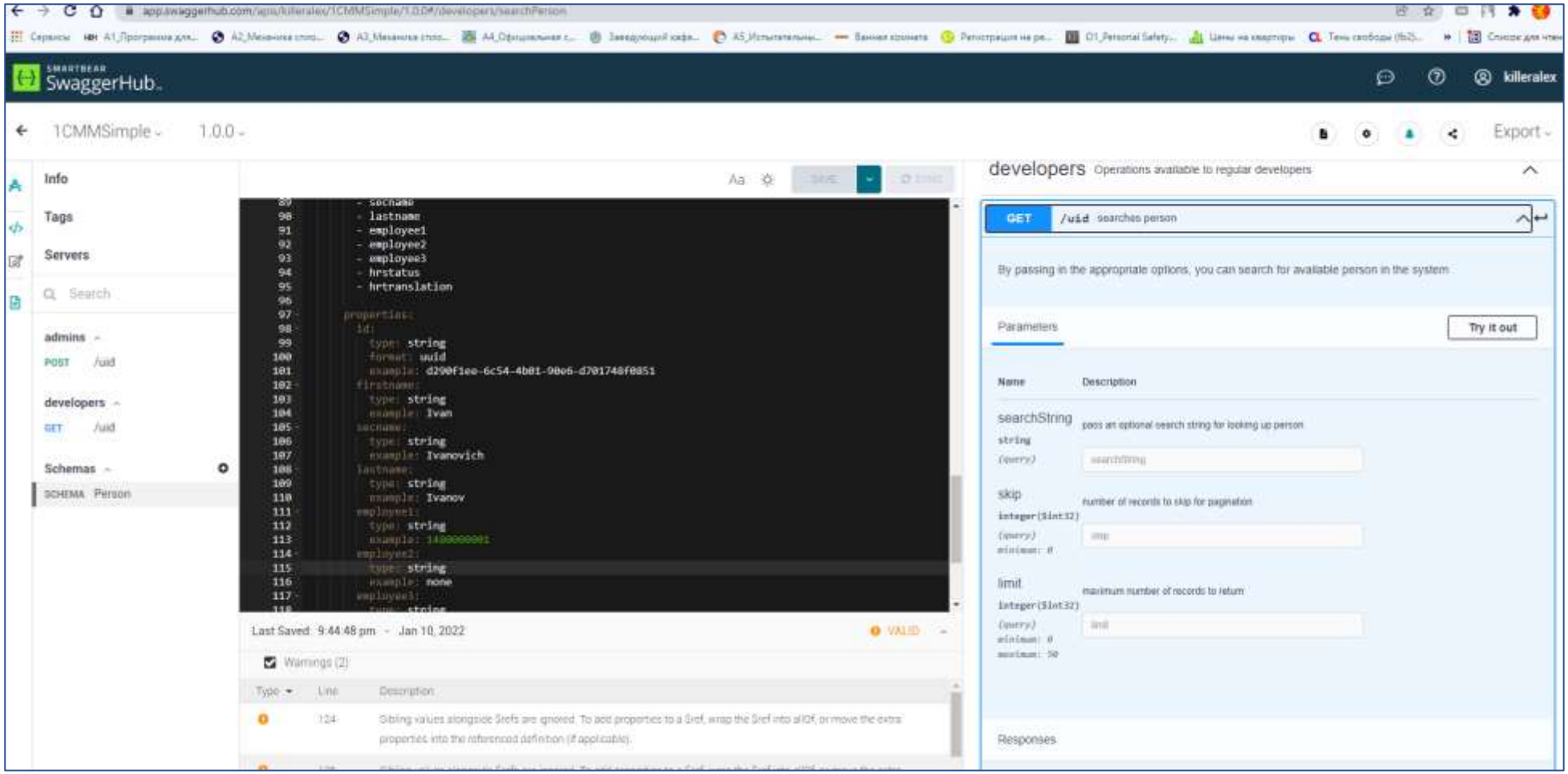


Рисунок П2 – Пример описания API в SwaggerHub (GET)