**Структура спецификации**

Описываемое требование

1. Контракты. Контракт представляет собой подробное описание каждого поля запроса или ответа. Структура контракта:
   1. Type - тип поля
   2. Field - название поля в json
   3. Description - описание поля
   4. req - обязательно ли json должен содержать данное поле
2. Описание запросов, необходимых для обеспечения требования. Запросы должны обеспечивать полный CRUD операций (создание, чтение, изменение, удаление) и удовлетворять принципам REST архитектуры (например, <https://www.infoq.com/articles/rest-introduction>).
3. Для выполнения операций использовать следующие методы http:

|  |  |
| --- | --- |
| **Method** | **Description** |
| GET | Получение ресурса(ов) |
| POST | Создание ресурса |
| PUT | Обновление полей ресурса |
| DELETE | Удаление ресурса |

1. В случае ошибки должен возвращаться следующий контракт

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Field** | **Description** | **Req.** |
| string | message | Сообщение ошибки. | yes |
| int | code | Уникальный код ошибки. | yes |
| string | field | Название поля объекта запроса, вызвавшего ошибку. | no |

1. В конце каждого запроса должно быть приведено описание возможных ошибок в следующем формате:
   1. Http - http код ошибки
   2. Code - локальный код ошибки
   3. Description - описание ошибки
2. Не использовать http код 500 для стандартных ошибок. Если пользователь видит 500 код, значит в системе произошла критическая ошибка, которую надо исправить.

**Пример:**

1. **Управление исполнителями**
   1. Контракты
      1. Subgroup

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Field** | **Description** | **req** |
| long | id | номер подгруппы | true |
| long[] | studentIds | список id студентов подгруппы | true |

* + 1. CreateSubgroup

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Field** | **Description** | **req** |
| long[] | studentIds | список id студентов подгруппы | true |

* 1. Создание подгруппы

POST http://host\_name:port\_number/subgroups

Контракт запроса: CreateSubgroup

Контракт ответа: Subgroup

Пример запроса

{

"studentIds": [

1,

2

]

}

Пример ответа:

{

"id": 1,

"studentIds": [

1,

2

]

}

Возможные ошибки:

|  |  |  |
| --- | --- | --- |
| **Http** | **Code** | **Description** |
| 401 | 2 | Метод требует авторизации (токен) |
| 403 | 3 | Невозможно расшифровать/распарсить токен |
| 401 | 4 | Токен валидный, но в базе отсутствует пользователь, которому он соответствует |
| 400 | 5 | Неизвестное поле запроса |
| 400 | 6 | Невалидное поле запроса (например, нулевой или отрицательный id) |
| 404 | 7 | Не найдена сущность в базе (в данном случае студент с одним из studentIds) |

* 1. Изменение списка студентов подгруппы

PUT http://host\_name:port\_number/subgroups/{subgroupId}

Контракт запроса: CreateSubgroup

Контракт ответа: Subgroup

Пример запроса

{

"studentIds": [

1,

3

]

}

Пример ответа:

{

"id": 1,

"studentIds": [

1,

3

]

}

|  |  |  |
| --- | --- | --- |
| **Http** | **Code** | **Description** |
| 401 | 2 | Метод требует авторизации (токен) |
| 403 | 3 | Невозможно расшифровать/распарсить токен |
| 401 | 4 | Токен валидный, но в базе отсутствует пользователь, которому он соответствует |
| 400 | 5 | Неизвестное поле запроса |
| 400 | 6 | Невалидное поле запроса (например, нулевой или отрицательный id) |
| 404 | 7 | Не найдена сущность в базе (в данном случае студент с одним из studentIds или подгруппа с таким subgroupId) |

* 1. Удаление подгруппы

DELETE http://host\_name:port\_number/subgroups/{subgroupId}

Контракт запроса: void

Контракт ответа: void

|  |  |  |
| --- | --- | --- |
| **Http** | **Code** | **Description** |
| 401 | 2 | Метод требует авторизации (токен) |
| 403 | 3 | Невозможно расшифровать/распарсить токен |
| 401 | 4 | Токен валидный, но в базе отсутствует пользователь, которому он соответствует |
| 400 | 5 | Неизвестное поле запроса |
| 404 | 7 | Не найдена сущность в базе (в данном случае подгруппа с таким subgroupId) |

* 1. Получение списка студентов, состоящих в подгруппе

GET http://host\_name:port\_number/subgroups/{subgroupId}

Контракт запроса: void

Контракт ответа: Subgroup

Пример ответа:

{

"id": 1324,

"studentIds": [

1,

3

]

}

|  |  |  |
| --- | --- | --- |
| **Http** | **Code** | **Description** |
| 401 | 2 | Метод требует авторизации (токен) |
| 403 | 3 | Невозможно расшифровать/распарсить токен |
| 401 | 4 | Токен валидный, но в базе отсутствует пользователь, которому он соответствует |
| 400 | 5 | Неизвестное поле запроса |
| 404 | 7 | Не найдена сущность в базе (в данном случае подгруппа с таким subgroupId) |

* 1. Получение списка всех подгрупп, в которые входит студент

GET http://host\_name:4242/subgroups/student/{studentId}

Контракт запроса: void

Контракт ответа: Subgroup[]

Пример ответа:

{

"subgroups": [

{

"id": 1324,

"studentIds": [

1,

3

]

},

{

"id": 1325,

"studentIds": [

1,

2

]

}

]

}

|  |  |  |
| --- | --- | --- |
| **Http** | **Code** | **Description** |
| 401 | 2 | Метод требует авторизации (токен) |
| 403 | 3 | Невозможно расшифровать/распарсить токен |
| 401 | 4 | Токен валидный, но в базе отсутствует пользователь, которому он соответствует |
| 400 | 5 | Неизвестное поле запроса |
| 404 | 7 | Не найдена сущность в базе (в данном случае студент с таким studentId) |