REST API для запроса расписания и журнала

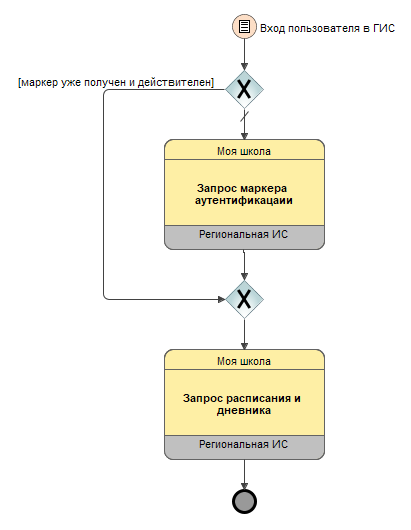
Общие сведения

## Назначение

REST API для запроса расписания и журнала (дневника ученика) предназначен для получения сведений из расписания и журнала образовательной организации из региональной информационной системы при входе пользователя ФГИС «Моя школа» в систему.

## Процесс взаимодействия

Процесс взаимодействия между ФГИС «Моя школа» и региональной информационной системой показан на рисунке в форме диаграммы хореографии на языке BPMN.



1. – Взаимодействия между ФГИС «Моя школа» и региональной информационной системой

При успешном входе пользователя ФГИС «Моя школа» запрашивает у региональной информационной системы маркер аутентификации, а затем обращается к региональной информационной системе с запросом расписания и дневника.

Запрос маркера аутентификации может быть пропущен, если срок его действия не истек. Он должен осуществляться по стандартному протоколу OAuth 2.0 с использованием параметра grant\_type = client\_credentials и в настоящем документе не описан.

# Ресурсы REST

## Общие сведения

Во взаимодействии используются ресурсы REST, перечисленные в таблице 1.

1. – Используемые ресурсы REST

| Имя | Наименование | Описание |
| --- | --- | --- |
| Основные ресурсы | | |
| Schedule | Расписание и дневник | Расписание уроков и дневник (для ученика) |
| Вспомогательные ресурсы | | |
| OperationOutcome | Результат операции | Результат операции с экземпляром ресурса REST |

## Запрос расписания и дневника

Запрос расписания и дневника осуществляется с помощью метода HTTP GET следующего вида:

GET [Базовый\_URL]/Schedule/<идентификатор учётной записи ЕСИА>{?start=<дата начала>&end=<дата конца>}

При успешной обработке возвращается код статуса HTTP 200 OK, а тело ответа содержит экземпляр ресурса Schedule. При ошибке возвращается соответствующий код статуса HTTP, а тело ответа содержит экземпляр ресурса OperationOutcome с детальными сведениями об ошибке.

Необязательные параметры start и end задают период расписания. Если указан один из них, должен быть указан другой, при этом дата начала расписания не может быть больше даты конца. По умолчанию содержание ресурса Schedule должно охватывать период учебы за три недели: предыдущую, текущую (на момент запроса) и следующую.

# Логическая модель ресурсов REST

## Общие сведения

Логическая модель ресурса REST представляется в виде диаграммы классов UML и иерархической таблицы, имеющей формат, описанный в таблице 2. В общем случае ресурс описан в виде нескольких классов, включая головной класс с именем ресурса и вспомогательные классы, связанные с головным и между собой отношениями направленной композиции. Имена вспомогательных классов уникальны только в пределах конкретного ресурса.

Независимо от числа вспомогательных классов каждый ресурс описывается одной иерархической таблицей, формат которой описан в таблице 2.

1. – Формат иерархической таблицы

| Графа | Содержание |
| --- | --- |
| Имя | Имя поля ресурса. Для вспомогательных классов указывается имя целевого конца композиции, а за ним имена полей вспомогательного класса, имеющие в качестве префикса имя целевого конца и символ точки, например, class.lesson.subject |
| Описание | Описание поля и сведения об ограничениях, наложенных на поле |
| Тип | Тип данных поля. Для поля, представляющего целевой конец композиции, в качестве имени типа данных используется BackboneElement |
| Кратность | Кратность: нижняя и верхняя граница допустимого числа повторений поля |

## Логическая модель ресурса Schedule (Расписание и дневник)

Логическая модель ресурса Schedule (Расписание и дневник) показана на рисунке 2 и приведена в таблице 3. Она охватывает в том числе случаи, когда учитель имеет несколько мест работы, а ученик (что маловероятно, но теоретически возможно) получает образование в нескольких образовательных организациях. Состав данных включает в себя не только расписание уроков, но еще дополнительные сведения о каждом уроке, включая тему и домашнее задание, а для ученика еще и сведения об отсутствии на уроке и полученных оценках. Это дает возможность учителю получить информацию о поурочном планировании, а ученику получить информацию, которая обычно содержится в дневнике.

Факт обучения по индивидуальной программе, при котором темы и домашние задания относятся не к классу, а к ученику, должен быть отражен в поле note (Примечание) в произвольной форме.



1. – Логическая модель ресурса Schedule
2. – Состав полей ресурса Schedule

| Имя | Описание | Тип данных | Кратность |
| --- | --- | --- | --- |
| Унаследованы от абстрактного класса Resource | | | |
| id | Идентификатор учётной записи ЕСИА ученика или учителя | id | 1..1 |
| meta | Метаданные экземпляра ресурса | Meta | 1..1 |
| meta.lastUpdated | Дата и время формирования содержания ресурса (включая часовой пояс) | instant | 1..1 |
| Собственные атрибуты | | | |
| school | Расписание в образовательной организации | BackboneElement | 0..\* |
| school.id | Идентификатор образовательной организации, присвоенный Рособрнадзором в сведениях о лицензии и государственной аккредитации | id | 1..1 |
| school.name | Наименование образовательной организации | string | 1..1 |
| school.scheduleUrl | Адрес URL страницы расписания данного лица. Если параметры начала и конца периода заданы, то на начало периода, если не заданы – на день запроса | uri | 1..1 |
| school.period | Период, охваченный расписанием в данной образовательной организации | Period | 1..1 |
| school.period.start | Дата начала периода | date | 1..1 |
| school.period.end | Дата конца периода | date | 1..1 |
| school.academYear | Академический год | code | 1..1 |
| school.class | Класс или группа | BackboneElement | 0..\* |
| school.class.class | Идентификатор класса, группы | code | 1..1 |
| school.class.grade | Параллель | code | 1..1 |
| school.class.lesson | Урок | BackboneElement | 1..\* |
| school.class.lesson.id | Глобально уникальный идентификатор урока (для привязки оценок и посещаемости к уроку) | id | 1..1 |
| school.class.lesson.subject | Предмет | string | 1..1 |
| school.class.lesson.insteadOf | Предмет и тема замененного урока | string | 0..1 |
| school.class.lesson.topic | Тема урока | string | 0..1 |
| school.class.lesson.homework | Домашнее задание | string | 0..1 |
| school.class.lesson.period | Начало и конец урока | Period | 1..1 |
| school.class.lesson.period.start | Дата и время начала урока | date | 1..1 |
| school.class.lesson.period.end | Дата и время конца урока | date | 1..1 |
| school.class.lesson.classroom | Место проведения урока (для дистанционного урока – место преподавателя) | code | 1..1 |
| school.class.lesson.distant | Признак дистанционного урока | boolean | 0..1 |
| school.class.lesson.note | Примечание | string | 0..1 |
| school.class.lesson.attendance | Отсутствие на уроке | BackboneElement | 0..1 |
| school.class.lesson.attendance.code | Код причины отсутствия. Допустимые значения: н (Отсутствие по неуважительной причине) | п (Отсутствие по уважительной причине) | б (Отсутствие по болезни) | о (Опоздание) | code | 1..1 |
| school.class.lesson.score | Оценка | BackboneElement | 0..\* |
| school.class.lesson.score.code | Вид работы, например, 'домашнее задание', 'контрольная', в соответствии с категориями работ, принятыми в данной образовательной организации | code | 0..1 |
| school.class.lesson.score.weight | Вес оценки в соответствии с категориями весов, принятыми в данной образовательной организации | code | 0..1 |
| school.class.lesson.score.score | Выставленная оценка в соответствии с системой оценок, принятой в данной образовательной организации | code | 1..1 |
| school.class.lesson.score.score2 | Вторая оценка в соответствии с системой оценок, принятой в данной образовательной организации | code | 0..1 |
| school.class.lesson.score.scoreSystem | Система оценок (см. таблицу 14). В случае иной системы должно быть указано ее наименование | code | 0..1 |

## Логическая модель ресурса OperationOutcome (Результат операции)

Ресурс OperationOutcome (результат операции) служит для описания ошибок, предупреждений и информационных сообщений о результате предпринятой операции. Диаграмма классов UML ресурса OperationOutcome показана на рисунке 3, состав элементов приведен в таблице 4.



1. – Ресурс OperationOutcome (Результат операции)
2. – Состав элементов ресурса OperationOutcome

| Имя | Описание | Тип данных | Кратность |
| --- | --- | --- | --- |
| Унаследованы от абстрактного класса Resource | | | |
| id | Глобально уникальный идентификатор экземпляра ресурса | id | 1..1 |
| meta | Метаданные экземпляра ресурса | Meta | 1..1 |
| meta.lastUpdated | Дата и время формирования содержания ресурса (включая часовой пояс) | instant | 1..1 |
| Собственные атрибуты | | | |
| issue | Ошибка, предупреждение или информационное сообщение, создаваемое в результате действия системы | BackboneElement | 1..\* |
| issue.severity | Серьезность отклонения от успешной обработки (см. таблицу 14) | code | 1..1 |
| issue.code | Тип отклонения от нормальной обработки (см. таблицу 14). Система, создающая экземпляр ресурса OperationOutcome, должна выбрать наиболее подходящий код отклонения и может предусмотреть дополнительный код ошибки в элементе details | code | 1..1 |
| issue.details | Дополнительные сведения об ошибке, например, ее описание или код, присвоенный системой | string | 0..1 |

# Представление экземпляра ресурса в формате JSON

## Общие сведения

Представление экземпляра ресурса в формате JSON основано на спецификации STD 90 (RFC 8259) (<https://www.rfc-editor.org/rfc/pdfrfc/rfc8259.txt.pdf>). Экземпляр ресурса представляется как объект JSON, у которого первое свойство имеет имя "resourceType", а значением служит имя типа ресурса.

Имена свойств чувствительны к регистру. Имя свойства должно быть уникальным в экземпляре ресурса.

Объект JSON не должен быть пустым. Если в экземпляре ресурса присутствует элемент, он должен иметь хотя бы одно свойство из числа определенных его типом. Свойство, имеющее пустое значение, не должно включаться в экземпляр ресурса.

Элемент логической модели, у которого максимальная кратность превышает 1 (например, 0..\*) представляется в JSON как типизированный массив. Имя массива должно быть в единственном числе. Такой элемент представляется как массив даже в том случае, если в конкретном экземпляре ресурса он не повторяется.

Элемент, имеющий в модели тип данных boolean, представляется как JSON boolean, а элементы всех других простых типов представляются как JSON string, соответствующие шаблонам, описанным для этих типов данных.

Элементы модели, имеющие комплексные типы данных (например, Period или BackboneElement) представляются как объекты JSON, содержащие компоненты соответствующих типов данных.

## Представление типов данных, используемых в модели

### Тип данных boolean

Булевский тип данных (true/false). Общие сведения о типе данных boolean приведены в таблице 5.

1. – Общие сведения о типе данных boolean

|  |  |
| --- | --- |
| Имя | Описание |
| boolean | Значение true | false  В JSON –- boolean (true или false).  Regex: true|false |

### Тип данных code

Тип данных code предназначен для представления перечислимых значений (контролируемых строк) и является специализацией строкового типа string. Общие сведения о типе данных code приведены в таблице 6.

1. – Общие сведения о типе данных code

| Имя | Описание |
| --- | --- |
| code | Указывает, что значение value берется из множества контролируемых строк, где-либо определенных.  Технически тип code ограничен строкой, в которой не меньше одного символа и отсутствуют ведущие и концевые пробельные элементы, и в содержании которой могут быть только единичные пробелы.  В JSON – string.  Regex: [^\s]+(\s[^\s]+)\* |

### Тип данных dateTime

Тип данных dateTime используется для представления дат и времени в соответствии со стандартом ISO 8601. Общие сведения о типе данных dateTime приведены в таблице 7.

1. – Общие сведения о типе данных dateTime

| Имя | Описание |
| --- | --- |
| dateTime | Дата, дата и время или часть даты (например, только год или год + месяц), используемые при коммуникации с человеком.  Формат YYYY, YYYY-MM, YYYY-MM-DD или YYYY-MM-DDThh:mm:ss+zz:zz, например, 2018, 1973-06, 1905-08-23, 2015-02-07T13:28:17-05:00 или 2017-01-01T00:00:00.000Z.  Если часы и минуты указаны, то часовой пояс ДОЛЖЕН быть указан. Секунды могут быть указаны в соответствии с этой схемой, но могут быть заполнены нулями и могут игнорироваться получателем. Даты ДОЛЖНЫ быть валидными. Время "24:00" не разрешено. Дополнительные високосные секунды разрешены.  В JSON string – объединение xs:dateTime, xs:date, xs:gYearMonth, xs:gYear.  Regex: ([0-9]([0-9]([0-9][1-9]|[1-9]0)|[1-9]00)|[1-9]000)(-(0[1-9]|1[0-2])(-(0[1-9]|[1-2][0-9]|3[0-1])(T([01][0-9]|2[0-3]):[0-5][0-9]:([0-5][0-9]|60)(\.[0-9]+)?(Z|(\+|-)((0[0-9]|1[0-3]):[0-5][0-9]|14:00)))?)?)? |

### Тип данных id

Тип данных id предназначен для представления идентификаторов. Общие сведения о типе данных id приведены в таблице 8.

1. – Общие сведения о типе данных id

| Имя | Описание |
| --- | --- |
| id | Любое сочетание символов в кодировке ASCII в нижнем или верхнем регистре ('A'...'Z', и 'a'...'z'), цифры ('0'...'9'), знаки '-' и '.', длина которого не превышает 64 символа. (Им может быть целое число, идентификатор UUID без префикса или любой шаблон, удовлетворяющий этим ограничениям.)  В JSON – string.  Regex: [A-Za-z0-9\-\.]{1,64} |

### Тип данных instant

Тип данных instant предназначен для представления штампа даты и времени с точностью до секунды или более высокой. Общие сведения о типе данных instant приведены в таблице 9.

1. – Общие сведения о типе данных instant

| Имя | Описание |
| --- | --- |
| instant | Тип instant представляет собой время в формате YYYY-MM-DDThh:mm:ss.sss+zz:zz (например, 2015-02-07T13:28:17.239+02:00 или 2017-01-01T00:00:00Z). Время ДОЛЖНО быть указано с точность до секунд или выше и ДОЛЖНО указывать часовой пояс.  В JSON string – xs:dateTime.  Regex: ([0-9]([0-9]([0-9][1-9]|[1-9]0)|[1-9]00)|[1-9]000)-(0[1-9]|1[0-2])-(0[1-9]|[1-2][0-9]|3[0-1])T([01][0-9]|2[0-3]):[0-5][0-9]:([0-5][0-9]|60)(\.[0-9]+)?(Z|(\+|-)((0[0-9]|1[0-3]):[0-5][0-9]|14:00)) |

### Тип данных string

Тип данных string представляет собой строковые данные в кодировке Unicode, которые могут быть записаны несколькими строками (то есть могут содержать символы возврата строки и перевода каретки). Общие сведения о типе данных string приведены в таблице 10.

1. – Общие сведения о типе данных string

| Имя | Описание |
| --- | --- |
| string | Последовательность символов Unicode.  В JSON – string.  Строка string ДОЛЖНА не превышать 1MB (1024\*1024 символов и не содержать символы Unicode с кодом меньше 32, за исключением u0009 (горизонтальная табуляция), u0010 (возврат каретки) и u0013 (перевод строки). Ведущие и концевые пробельные символы разрешены, но ДОЛЖНЫ быть удалены, если используется форма XML.  Примечание – Это означает, что строка, состоящая только из пробельных элементов, должна быть превращена в пустую строку, но это будет трактоваться как недопустимое значение элемента. Поэтому строки должны содержать не пробельные символы.  Regex: [ \r\n\t\S]+ |

### Тип данных uri

Тип данных uri используется для представления унифицированных идентификаторов ресурсов. Общие сведения о типе данных uri приведены в таблице 11.

1. – Общие сведения о типе данных uri

| Имя | Описание |
| --- | --- |
| uri | Ссылка Uniform Resource Identifier Reference (RFC 3986).  Примечание – URI чувствительны к регистру. Для UUID (urn:uuid:53fefa32-fcbb-4ff8-8a92-55ee120877b7) используйте нижний регистр.  В строке JSON string – URI  Regex: \S\* (Это выражение разрешает очень многое, но URI должны быть валидными.) |

### Тип данных Period

Тип данных Period предназначен для представления периода времени или (неопределенного) момента времени внутри периода. Общие сведения о типе данных Period приведены в таблице 12, а состав элементов – в таблице 13.

1. – Общие сведения о типе данных Period

|  |  |
| --- | --- |
| Имя | Описание |
| Period | Период времени, определенный датой/временем начала и конца. Период задает диапазон времени. По контексту использования определяется, имеется ли в виду весь диапазон или какой-то один момент времени внутри этого диапазона |

1. – Состав элементов типа данных Period

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Описание | Тип | Кратность |
| start | Дата и время начала периода (включительно) | dateTime | [0..1] |
| end | Дата и время конца периода (включительно) | dateTime | [0..1] |

Терминологические ресурсы

В сведениях о расписаниях и журналах используются системы кодирования (справочники и классификаторы), перечисленные в таблице 14.

1. – Используемые системы кодирования

| Поле | Наименование | Источник | Имя системы кодирования |
| --- | --- | --- | --- |
| Schedule.academYear | Академический год | Перечисление в формате <год начала>/<год конца> | CodeSystem-Edu-EduGroupAcademYear-1 |
| Schedule.school.class.lesson.score.scoreSystem | Система оценок | Перечисление: 5 | 10 | 12 | 100 | А (Американская) | ЗЧ (Зачет/незачет) | CodeSystem-MySchool-ScoreSystemType-1 |
| OperationOutcome.issue.severity | Серьезность отклонения от успешной обработки | Перечисление: fatal (Фатальная ошибка) | error (Ошибка) | warning (Предупреждение) | information (Информация) | CodeSystem-Edu-EventIssueSeverity-1 |
| OperationOutcome.issue.code | Тип отклонения от нормальной обработки | Перечисление: invalid (Несоответствие спецификации) | structure (Ошибочная структура) | required (Элемент обязателен) | value (Ошибочное значение элемента) | invariant (Нарушено ограничение) | security (Ошибка доступа) | login (Требуется аутентификация) | unknown (Неизвестный принципал) | expired (Сеанс закончен) | forbidden (Доступ запрещен) | suppressed (Частичная информация) | processing (Ошибка обработки) | not-supported (Не поддерживается) | duplicate (Дубликат) | multiple-matches (Несколько совпадений) | not-found (Не найден) | retired (Прекращено) | too-long (Слишком длинное) | code-invalid (Ошибочный код) | extension (Недопустимое расширение) | too-costly (Слишком затратное) | business-rule (Нарушено бизнес правило) | conflict (Конфликт версий) | transient (Преходящая ошибка) | lock-error (Экземпляр блокирован) | no-store (Недостаточно места) | exception (Возникло исключение) | timeout (Таймаут) | incomplete (Неполные результаты) | throttled (Система на обслуживании) | informational (Информационное) | CodeSystem-Edu-EventIssueCode-1 |

Примеры

## Расписание учителя

В таблице 15 приведен пример расписания учителя физики, преподающего в Заковряшинской СОШ (идентификатор 03b68eb8-169f-9a50-ba4e-e9b02a7897f0 в Сводном реестре лицензий Рособрнадзора) и имеющего идентификатор учётной записи ЕСИА 1563978. В качестве сервиса конференцсвязи в школе используется Discord (<https://discord.com/>).

Для упрощения в качестве периода расписания вместо трех недель выбраны два дня. Урок в классе 9а проводится дистанционно из компьютерного класса (кабинет 27). Уроки в классах 8б и 8д проводятся в кабинете физики (кабинет 36).

1. – Пример расписания учителя

| Класс | Дата и время начала урока | Дата и время конца урока | Предмет | Тема | Домашнее задание |
| --- | --- | --- | --- | --- | --- |
| 8б | 2021-12-20T09:30:00+04:00 | 2021-12-20T10:15:00+04:00 | Физика | Закон Ома | Перышкин упр.29 1, 2, 5 |
| 8д | 2021-12-20T10:30:00+04:00 | 2021-12-20T11:15:00+04:00 | Физика | Закон Ома | Перышкин упр.29 1, 2, 5 |
| 9а | 2021-12-21T09:30:00+04:00 | 2021-12-21T10:15:00+04:00 | Физика | Магнитное поле | Перышкин упр.31 1, 3 |

Представление расписания учителя в формате JSON:

{

"resourceType": "Schedule",

"id": "1563978",

"meta": {

"lastUpdated": "2021-12-20T17:30:11+04:00"

},

"school": [

{

"id": "03b68eb8-169f-9a50-ba4e-e9b02a7897f0",

"name": "Заковряшинская СОШ",

"scheduleUrl": "https://dnevnik.reg22.ru/75612?2021-12-20",

"period": {

"start": "2021-12-20",

"end": "2021-12-21"

},

"academYear": "2021/22",

"class": [

{

"class": "8б",

"grade": "8",

"lesson": [

{

"id": "8149f194-ae1c-475c-a135-6ff84488d2af",

"subject": "Физика",

"topic": "Закон Ома",

"homework": "Перышкин упр.29 1, 2, 5",

"period": {

"start": "2021-12-20T09:30:00+04:00",

"end": "2021-12-20T10:15:00+04:00"

},

"classroom": "36"

}

]

},

{

"class": "8д",

"grade": "8",

"lesson": [

{

"id": "01f15d63-88b2-4892-9e69-2c20f600954b",

"subject": "Физика",

"topic": "Закон Ома",

"homework": "Перышкин упр.29 1, 2, 5",

"period": {

"start": "2021-12-20T10:30:00+04:00",

"end": "2021-12-20T11:15:00+04:00"

},

"classroom": "36"

}

]

},

{

"class": "9а",

"grade": "9",

"lesson": [

{

"id": "9bdab85e-c7bd-4555-9997-6a32ca13a13f",

"subject": "Физика",

"topic": "Магнитное поле",

"homework": "Перышкин упр.31 1, 3",

"period": {

"start": "2021-12-21T09:30:00+04:00",

"end": "2021-12-21T10:15:00+04:00"

},

"classroom": "27",

"distant": true

}

]

}

],

"distantUrl": "https://discord.com/"

}

]

}

## Расписание и дневник ученика

В таблице 16 приведен пример расписания ученика класса 8бАлгебра Заковряшинской СОШ (идентификатор 03b68eb8-169f-9a50-ba4e-e9b02a7897f0 в Сводном реестре лицензий Рособрнадзора), имеющего идентификатор учётной записи ЕСИА 5612003. В качестве сервиса конференцсвязи в школе используется Discord (<https://discord.com/>).

Для упрощения в качестве периода расписания вместо трех недель выбран один день, в котором были четыре урока: физика, алгебра, информатика, литература. Урок информатики проводится вместо урока географии. На первом уроке ученик отсутствовал по неуважительной причине, на второй опоздал. На третьем уроке получил оценку 5 по 5-балльной системе за ответ у доски, на четвертом уроке было сочинение на тему «Мой край», за которое были выставлены оценки 4/5 по 5-балльной системе.

1. – Пример расписания ученика

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Дата и время начала урока | Дата и время конца урока | Предмет | Тема | Домашнее задание |
| 2021-12-20T09:30:00+04:00 | 2021-12-20T10:15:00+04:00 | Физика | Закон Ома | Перышкин упр. 29 1, 2, 5 |
| 2021-12-20T10:30:00+04:00 | 2021-12-20T11:15:00+04:00 | Алгебра | Рациональные дроби | Макарычев упр. 24, 26 |
| 2021-12-20T11:30:00+04:00 | 2021-12-20T12:15:00+04:00 | Информатика | Информационные процессы | Семакин стр. 21 2, 3, 5 |
| 2021-12-20T12:30:00+04:00 | 2021-12-20T13:15:00+04:00 | Литература | Сочинение на тему «Мой край» | ­ |

Представление расписания ученика в формате JSON:

{

"resourceType": "Schedule",

"id": "5612003",

"meta": {

"lastUpdated": "2020-12-20T16:11:13+04:00"

},

"school": [

{

"id": "03b68eb8-169f-9a50-ba4e-e9b02a7897f0",

"name": "Заковряшинская СОШ",

"scheduleUrl": "https://dnevnik.reg22.ru/33478?2021-12-20",

"period": {

"start": "2021-12-20",

"end": "2021-12-20"

},

"academYear": "2021/22",

"class": [

{

"class": "8б",

"grade": "8",

"lesson": [

{

"id": "8149f194-ae1c-475c-a135-6ff84488d2af",

"subject": "Физика",

"topic": "Закон Ома",

"homework": "Перышкин упр.29 1, 2, 5",

"period": {

"start": "2021-12-20T09:30:00+04:00",

"end": "2021-12-20T10:15:00+04:00"

},

"classroom": "36",

"attendance": {

"code": "н"

}

}

]

},

{

"class": "8б",

"grade": "8",

"lesson": [

{

"id": "01f15d63-88b2-4892-9e69-2c20f600954b",

"subject": "Алгебра",

"topic": "Рациональные дроби",

"homework": "Макарычев упр. 24, 26",

"period": {

"start": "2021-12-20T10:30:00+04:00",

"end": "2021-12-20T11:15:00+04:00"

},

"classroom": "42",

"attendance": {

"code": "о"

}

}

]

},

{

"class": "8б",

"grade": "8",

"lesson": [

{

"id": "9bdab85e-c7bd-4555-9997-6a32ca13a13f",

"subject": "Информатика",

"insteadOf": "География: Ближний Восток",

"topic": "Информационные процессы",

"homework": "Семакин стр. 21 2, 3, 5",

"period": {

"start": "2021-12-20T11:30:00+04:00",

"end": "2021-12-20T12:15:00+04:00"

},

"classroom": "42",

"score": [

{

"code": "Ответ у доски",

"score": "5",

"scoreSystem": "5"

}

]

}

]

},

{

"class": "8б",

"grade": "8",

"lesson": [

{

"id": "5b6de220-6e6a-4bfe-9b14-f71111069278",

"subject": "Литература",

"topic": "Сочинение на тему «Мой край»",

"period": {

"start": "2021-12-20T12:30:00+04:00",

"end": "2021-12-20T13:15:00+04:00"

},

"classroom": "42",

"score": [

{

"code": "Сочинение",

"weight": "1,8",

"score": "4",

"score2": "5",

"scoreSystem": "5"

}

]

}

]

}

],

"distantUrl": "https://discord.com/"

}

]

}

## Ошибка запроса

Если идентификатор учётной записи ЕСИА, указанный в запросе расписания учителя, не найден, то должен быть возвращен код ответа HTTP 404 и следующий экземпляр ресурса OperationOutcome:

{

"resourceType": "OperationOutcome",

"id": "25f1a94d-96e1-4731-8b2e-8b3d73520adc",

"issue": [

{

"severity": "error",

"code": "not-found",

"details": "Учётная запись ЕСИА 1563978 не найдена"

}

]

}