

Программный интерфейс СДО AcademicNT

Взаимодействие с программным интерфейсом (API) СДО AcademicNT осуществляется по HTTP с помощью REST-like запросов.

Существуют два уровня API:

- public — доступ без ограничений,
- private — доступ по ключу сессии.

При обращении к функциям уровня private запросы должны содержать в cookie идентификатор сессии, по которому и определяется ключ сессии.

Перечень доступных функций и их спецификация приведены ниже.

Функции public API

1 Экспорт календаря

По ключу возвращается информация о календаре пользователя в СДО в формате iCalendar. Используется для экспорта и синхронизации данных календаря в сторонние приложения.

Формат запроса:

```
http://de.ifmo.ru/api/public/calendar/<key>
```

Параметры:

- key – ключ календаря.

Ключ может быть создан пользователем в СДО AcademicNT.

2 Вывод паспорта беджа

По ключу возвращается страница с описанием беджа.

Формат запроса:

```
http://de.ifmo.ru/api/public/badgeinfo/<key>
```

Параметры:

- key – ключ беджа.

Ключ доступен в СДО AcademicNT в описании достижения.

3 Вывод беджа

По ключу возвращается файл изображения беджа.

Формат запроса:

```
http://de.ifmo.ru/api/public/badge/<key>
```

Параметры:

- key – ключ беджа.

Ключ доступен в СДО AcademicNT в описании достижения.

4 Вывод файла

По типу и ключу возвращается файл.

Формат запроса:

```
http://de.ifmo.ru/api/public/file/<type>/<key>
```

Параметры:

- type – ключ файла,
- key – тип файла.

Используется для вывода вложений и прикрепленных к сообщениям файлов.

5 Вывод сообщения

По типу и ключу возвращается страница с записью из раздела сетевого общения.

Формат запроса:

```
http://de.ifmo.ru/api/public/meeting/<type>/<key>
```

Параметры:

- type – ключ записи,
- key – тип записи.

Используется для вывода вложений и прикрепленных к сообщениям файлов.

6 Регистрация пользователя на электронный курс

По идентификатору пользователя и идентификатору курса производится регистрация пользователя на курс и возвращает «1» в случае, если пользователь зарегистрирован и «-1» в противном случае.

Формат запроса:

```
http://de.ifmo.ru/api/public/getCourseAccess?  
pid=<pid>&courseid=<cid>
```

Параметры:

- pid – идентификатор пользователя,
- cid – идентификатор курса.

7 Вывод результата выполнения пользователем элементов электронного контроля

По идентификатору пользователя, идентификатору элемента и идентификатору курса выводится число, представляющее результат пользователя по элементу электронного контроля или «0», если таких данных не существует.

Формат запроса:

```
http://de.ifmo.ru/api/public/getMark?  
pid=<pid>&courseid=<cid>&unitid=<uid>
```

Параметры:

- pid – идентификатор пользователя,
- cid – идентификатор курса,
- uid – идентификатор элемента.

8 Вывод результата выполнения пользователем элементов электронного контроля

Метод, возвращающий по ID курса, ID сценария тестирования и ID пользователя массив объектов, содержащий все результаты пользователя по данному сценарию тестирования.

Формат запроса:

```
http://de.ifmo.ru/api/public/courseAttempts?  
pid=<pid>&courseid=<cid>&unitid=<uid>
```

Параметры:

- pid – идентификатор пользователя,
- cid – идентификатор курса,
- uid – идентификатор элемента.

Метод возвращает объект в формате JSON со следующими значениями:

- course_id – ID курса,
- unit_id – ID сценария,
- attempts – массив с объектами,
- start – дата входа в сценарий тестирования в формате ISO 8601 YYYYMMDDThhmmss±hhmm,
- end – дата завершения сценария тестирования в ISO 8601 YYYYMMDDThhmmss±hhmm, может быть null,
- result – количество баллов пользователя за данный сценарий тестирования, может быть null.

Пример возвращаемого ответа:

```
{  
  "course_id": 1,  
  "unit_id": 1,  
  "attempts": [  
    {  
      "start": "20150809T183142+0300",  
      "end": "20150809T183142+0300",  
      "result": 0  
    },  
    {  
      "start": "20150809T183142+0300",  
      "end": null,  
      "result": null,  
    }  
  ]  
}
```

9 Вывод информации о курсе

Метод, возвращающий по ID курса и ID сценария тестирования объект, содержащий текущие настройки данного сценария тестирования.

Формат запроса:

```
http://de.ifmo.ru/api/public/courseInfo?  
courseid=<cid>&unitid=<uid>
```

Параметры:

- `pid` – идентификатор пользователя,
- `cid` – идентификатор курса.

Ответ представлен в нотации JSON со следующими значениями:

- `course_id` – идентификатор курса,
- `unit_id` – идентификатор сценария,
- `attempts` – количество попыток на данный сценарий тестирования,
- `limit` – ограничение по времени в минутах на данный сценарий тестирования.

Пример возвращаемого ответа:

```
{  
  "course_id": 1,  
  "unit_id": 1,  
  "attempts" : 3,  
  "limit" : 30  
}
```

10 Вход с помощью внешней службы единого входа

Метод, позволяющий войти в систему AcademicNT с помощью аккаунта внешней службы единого входа при условии, что он привязан к существующей учетной записи системы AcademicNT.

Предназначен для использования в виде параметра `redirect_url` при запросах к службам единого входа.

Формат запроса:

```
https://de.ifmo.ru/api/public/oauth/login/<sso_name>
```

Параметры:

- `sso_name` – идентификатор службы единого входа, зарезервированный в AcademicNT.

Результатом выполнения является осуществление входа в систему при успехе, сообщение о необходимости привязать внешний аккаунт к существующей учетной записи AcademicNT или зарегистрировать новую учетную запись AcademicNT на его основе, если внешний аккаунт не зарегистрирован в AcademicNT, сообщение об ошибке в случае ошибки.

11 Регистрация с помощью внешней службы единого входа

Метод, позволяющий зарегистрировать новую учетную запись в системе AcademicNT с помощью аккаунта внешней службы единого входа при условии, что он еще не привязан к существующей учетной записи системы AcademicNT.

Предназначен для использования в виде параметра `redirect_url` при запросах к службам единого входа.

Формат запроса:

```
https://de.ifmo.ru/api/public/oauth/register/<sso_name>
```

Параметры:

- `sso_name` – идентификатор службы единого входа, зарезервированный в AcademicNT.

Результатом выполнения является создание новой учетной записи и осуществление входа в систему при успехе, сообщение об ошибке в случае ошибки.

Функции private API

1 Сводка электронного журнала

По запросу возвращается сводка данных электронного журнала в формате json. Сводка содержит данные о результатах текущего и рубежного контроля по всем учебным годам.

Формат запроса:

<http://de.ifmo.ru/api/private/eregister>

Параметры: отсутствуют.

В таблице ниже приведено описание объектов json:

Таблица 1. Описание объектов сводки электронного журнала в формате json

Имя объекта	Значение
<code>years</code>	Массив записей учебных лет. Может быть пустым.
<code>years/[i]/group</code>	Номер группы.
<code>years/[i]/studyyear</code>	Учебный год.
<code>years/[i]/subjects</code>	Массив учебных предметов. Может быть пустым.
<code>years/[i]/subjects/[j]/name</code>	Название предмета.
<code>years/[i]/subjects/[j]/semester</code>	Семестр.
<code>years/[i]/subjects/[j]/marks</code>	Массив результатов рубежного контроля. Может быть пустым.
<code>years/[i]/subjects/[j]/marks/[k]/mark</code>	Оценка.
<code>years/[i]/subjects/[j]/marks/[k]/markdate</code>	Дата выставления оценки.
<code>years/[i]/subjects/[j]/marks/[k]/worktype</code>	Тип рубежного контроля.
<code>years/[i]/subjects/[j]/points</code>	Массив данных текущего контроля.
<code>years/[i]/subjects/[j]/points/[k]/variable</code>	Название переменной текущего контроля.

years/[i]/subjects/[j]/points/[k]/max	Возможный максимум баллов.
years/[i]/subjects/[j]/points/[k]/limit	Пороговое значение.
years/[i]/subjects/[j]/points/[k]/value	Набранное количество баллов.

Пример возвращаемого ответа:

```
{
  "years": [{
    "group": "0000",
    "studyyear": "2012/2013",
    "subjects": [{
      "name": "Предмет",
      "semester": "1",
      "marks": [{
        "mark": "зачет",
        "markdate": "01.01.2013",
        "worktype": "Зачет"
      }],
      "points": [{
        "variable": "Семестр 1",
        "max": "100", "limit": "60", "value": "80"
      }]
    }]
  }]
}
```

2 Вывод файла

По типу и ключу возвращается файл, к которому имеет доступ текущий пользователь.

Формат запроса:

```
http://de.ifmo.ru/api/private/file/<тип>/<ключ>
```

Используется для прикрепленных к письмам файлов.

3 Вывод протокола изменений в электронном журнале

Метод, возвращающий массив, содержащий протокол изменений в электронном журнале для текущего пользователя за указанное количество дней.

Формат запроса:

```
http://de.ifmo.ru/api/private/_eregisterlog?days=<days>
```

Параметры:

- days – количество дней, целое число, не больше 365, не обязателен, значение по умолчанию – 1.

Ответ представлен в нотации JSON со следующими значениями:

- subject – название дисциплины,
- var – переменная курса, содержит поля:
 - name – название переменной,

- min – минимальное допустимое значение,
- max – максимальное допустимое значение,
- threshold – пороговое значение,
- value – текущее значение в электронном журнале,
- date – дата обновления значения,
- sign – ФИО выполнившего изменение значения.

Пример:

```
[
  {
    "subject": "Основы программирования",
    "var": {
      "name": "Текущее тестирование",
      "min": "0",
      "max": "6",
      "threshold": "3"
    },
    "value": "6",
    "date": "12.12.2015",
    "sign": "Иванов И.И."
  },
  {
    "subject": "Основы программирования",
    "var": {
      "name": "Лабораторные работы",
      "min": "0",
      "max": "6",
      "threshold": "3"
    },
    "value": "6",
    "date": "12.12.2015",
    "sign": "Иванов И.И."
  }
]
```

4 Вывод входящих сообщений

Метод, возвращающий массив, содержащий все или непрочитанные входящие сообщения за указанное количество дней для текущего пользователя.

Формат запроса:

```
http://de.ifmo.ru/api/private/mail?days=<days>&unreadOnly=<unread>
```

Параметры:

- days – количество дней, целое число, не больше 365, не обязателен, значение по умолчанию – 1,
- unread – признак того, что требуется вернуть все или только непрочитанные сообщения, возможные значения: “true”/”false”, не обязателен, значение по умолчанию – “false”.

Ответ возвращается в нотации JSON со следующими значениями:

- id – идентификатор письма,
- sender – ФИО отправителя,
- topic – тема письма,
- text – текст письма,
- date – дата отправления,
- isRead – признак того, что сообщение уже было прочитано,
- isFileAttached – прикреплен ли файл к письму,
- fileName – название файла.

Пример:

```
[
  {
    "id":1,
    "sender":"Иванов И.И.",
    "topic":"Консультация",
    "text":"Уважаемые студенты!\r\n
           Консультация состоится в указанном месте
           в назначенное время \r\n",
    "date":"25.02.2016",
    "isRead":false,
    "isFileAttached":false,
    "fileName":""
  },
  {
    "id":1,
    "sender":"Иванов И.И.",
    "topic":"вопросы к экзамену",
    "text":"Уважаемые студенты!\r\n
           Высылаю вопросы к экзамену \r\n",
    "date":"25.02.2016",
    "isRead":false,
    "isFileAttached":true,
    "fileName":"вопросы.txt"}
]
```

5 Привязка аккаунта внешней службы единого входа к профилю пользователя

Метод, позволяющий привязать аккаунт внешней службы единого входа к существующей учетной записи системы AcademiNT.

Предназначен для использования в виде параметра `redirect_url` при запросах к службам единого входа.

Формат запроса:

```
https://de.ifmo.ru/api/public/oauth/connect/<sso_name>?
state=<back_url_extension>
```

Параметры:

- `sso_name` – идентификатор службы единого входа, зарезервированный в AcademicNT.
- `state` – расширение адреса веб-приложения сервлета для возврата (например, `</servlet/distributedCDE?Rule=editPersonProfile>`)

Результатом выполнения является привязывание внешнего аккаунта к учетной записи и осуществление обратного перехода по указанному расширению url.