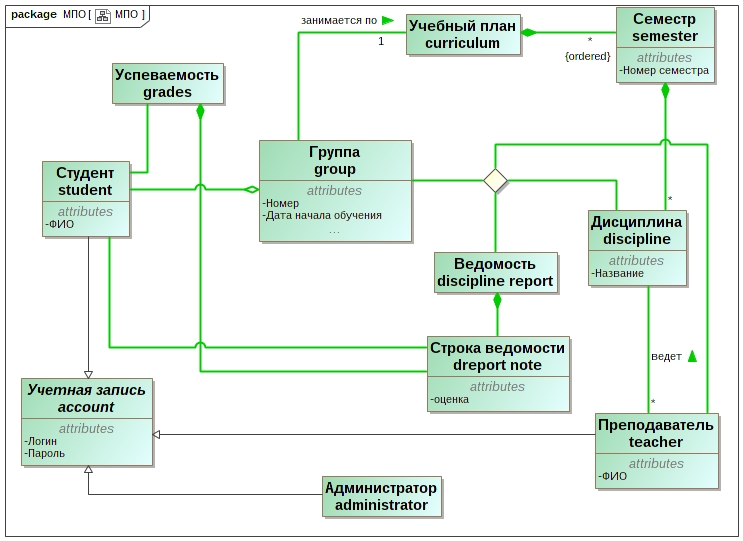
АСО ПМ, модуль «Dean»

# Описание подсистемы

Модуль «Dean» (далее Подсистема) позволяет:

* вносить через графический интерфейс данные, необходимые для процесса обучения (студенты, группы, преподаватели, дисциплины)
* создавать (изменять при надобности) процесс обучения
  + составлять учебный план на группу
  + назначать преподавателя на чтение лекций по определенной дисциплине (если несколько преподавателей могут вести эту дисциплину)
* просматривать через графический интерфейс данные, связанные с процессом обучения (успеваемость за предыдущие семестры, промежуточные данные за текущий семестр)
* автоматически / по требованию преподавателя выставлять оценки студентам по различным курсам на основании промежуточных данных.

# Описание предметной области

****

На данной диаграмме представлены сущности и отношения, принадлежащие подсистеме. В таблице ниже приведен глоссарий.

|  |  |  |
| --- | --- | --- |
| Термин | | Значение |
| Учетная запись | account | Абстрактный класс аккаунта с общими для всех данными (логин, пароль) |
| Администратор | administrator | Аккаунт администратора |
| Студент | student | Аккаунт студента |
| Успеваемость | grades | Массив строк ведомостей, связанных с конкретным студентом |
| Группа | group | Список студентов, учащихся в данной группе |
| Преподаватель | teacher | Аккаунт преподавателя |
| Учебный план | curriculum | Список упорядоченных семестров |
| Семестр | semester | Список дисциплин, проводимых в данном семестре |
| Дисциплина | discipline | Курс лекций по определенной тематике |
| Ведомость | discipline report | Массив, состоящий из строк ведомости |
| Строка ведомости | dreport note | Массив, состоящий из ФИО студента и итоговой оценки |

* Студент, преподаватель и администратор (сотрудник деканата) имеют свои аккаунты в АСО ПМ
* Студент является отделимой частью группы
* Группа имеет свой учебный план на весь срок обучения
* Семестр является неотделимой частью учебного плана
* Дисциплина является неотделимой частью семестра
* Преподаватель может вести одну или несколько дисциплин (возможны ситуации, когда одну дисциплину ведут несколько преподавателей)
* Группа посещает дисциплину, которую ведет преподаватель. Результат всех посещений — ведомость (итоговое оценивание каждого студента)
* Строка ведомости является неотделимой частью ведомости
* У студента есть своя успеваемость, состоящая из строк ведомостей, связанных с данным студентом

# Действующие лица

|  |  |
| --- | --- |
| Администратор (через подсистему GUI) | * Добавить данные в БД * Создание / изменение учебного плана * Назначение учебного плана на группу * Назначение преподавателя на чтение дисциплины |
| Преподаватель (через подсистему GUI) | * Просмотр промежуточных итогов по читаемой дисциплине в текущем семестре * Создание ведомости по читаемой дисциплине в текущем семестре |
| Студент (через подсистему GUI) | * Просмотр промежуточных итогов по дисциплине из учебного плана в текущем семестре * Просмотр своей успеваемости |

# Описание бизнес-процессов

|  |  |  |  |
| --- | --- | --- | --- |
| Сценарий | Приоритет(\*) | Пояснение | Тип интерфейса |
| Добавление данных (студенты, преподаватели, группы, дисциплины) | 1 | Администратор загружает в данные в БД. | GUI + API |
| Создание учебного плана | 1 | Администратор назначает количество семестров и распределяет определенные дисциплины по ним. Учебный план сохраняется в БД | GUI + API |
| Изменение учебного плана | 2 | Администратор вносит поправки в существующий учебный план. Результат сохраняется в БД | GUI + API |
| Назначение учебного плана на группу | 1 | Администратор выбирает группу и назначает на нее учебный план. Результат сохраняется в БД | GUI + API |
| Назначение преподавателя на чтение дисциплины | 1 | Администратор назначает преподавателя на чтение дисциплины из списка возможных. Результат сохраняется в БД | GUI + API |
| Просмотр промежуточных итогов | 2 | Преподаватель / студент запрашивает отчет в виде таблицы по определенной дисциплине в текущем семестре | GUI + API |
| Просмотр успеваемости | 2 | Студент запрашивает отчет в виде таблицы по всем прослушанным им дисциплинам | GUI + API |
| Создание ведомости по определенной дисциплине | 1 | Преподаватель создает ведомость посредством запроса либо назначая дату автоматического создания | GUI + API |

(\*) – приоритет убывает по возрастанию (1 – наивысший приоритет)

# Описание функций подсистемы, задействованных в сценариях

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Сценарий | П-т | Прототип | Входные данные | Выходные данные |
| Добавление данных (студенты, преподаватели, группы, дисциплины) | 1 | **add\_data(type, is\_file, data)** | * **type** – тип данных (студент, преподаватель, группа, дисциплина) * **data** – файл c данными или данные с GUI-формы * **is\_file** – флаг для обработки data | Код завершения работы функции |
| Создание учебного плана | 1 | **make\_curriculum(data)** | * **data** – данные с GUI-формы | Код завершения работы функции |
| Изменение учебного плана | 2 | **update\_curriculum(id, data)** | * **id** – id учебного плана из БД * **data** – данные с GUI-формы | Код завершения работы функции |
| Назначение учебного плана на группу | 1 | **link\_group\_curr(id\_group, id\_curr)** | * **id\_group** – id группы из БД * **id\_curr** – id учебного плана из БД | Код завершения работы функции |
| Назначение преподавателя на чтение дисциплины | 1 | **link\_disc\_teach(id\_group, id\_disc, id\_teach)** | * **id\_group** – id группы из БД * **id\_disc** – id дисциплины из БД * **id\_teach** – id учебного плана из БД | Код завершения работы функции |
| Просмотр промежуточных итогов | 2 | **get\_temp\_sheet(id\_group, id\_disc)** | * **id\_group** – id группы из БД * **id\_disc** – id дисциплины из БД | Код завершения работы функции |
| Просмотр успеваемости | 2 | **get\_marks(id\_stud)** | * **id\_ stud** – id студента из БД | Код завершения работы функции |
| Создание ведомости по определенной дисциплине | 1 | **make\_sheet(id\_group, id\_disc)** | * **id\_group** – id группы из БД * **id\_disp** – id дисциплины из БД | Код завершения работы функции |

# Описание программных интерфейсов (запросов)

1. **Добавление студента вручную** 
   1. Контракты
      1. PutStudentByHand\_In

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| string | name | ФИО студента | true |
| int | group\_id | id группы | true |

* + 1. PutStudentByHand\_Out

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| int | result\_code | 0 если операция прошла успешно,  иначе код ошибки | true |
| string | message | Текстовое описание результата | false |

* 1. Описание

|  |  |
| --- | --- |
| Метод | POST |
| URL | http://host\_name:port\_number/dean/admin/add\_student |
| Контракт запроса | PutStudentByHand\_In |
| Пример запроса | {  “name”: “Ковалев М.С.”,  “group\_id”: 145  } |
| Контракт ответа | PutStudentByHand\_Out |
| Пример ответа | {  “result\_code”: 0  }  или  }  “result\_code”: 1114,  “message”: “Таблица «%s» заполнена”  } |

1. **Добавление преподавателя вручную** 
   1. Контракты
      1. PutTeacherByHand\_In

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| string | name | ФИО преподавателя | true |
| int[] | discipline\_ids | id предметов, которые может вести преподаватель | true |

* + 1. PutTeacherByHand\_Out

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| int | result\_code | 0 если операция прошла успешно,  иначе код ошибки | true |
| string | message | Текстовое описание результата | false |

* 1. Описание

|  |  |
| --- | --- |
| Метод | POST |
| URL | http://host\_name:port\_number/dean/admin/add\_teacher |
| Контракт запроса | PutTeacherByHand\_In |
| Пример запроса | {  “name”: “Новиков Ф.А.”,  “discipline\_ids”: [14, 31]  } |
| Контракт ответа | PutTeacherByHand\_Out |
| Пример ответа | {  “result\_code”: 0  }  или  }  “result\_code”: 1114,  “message”: “Таблица «%s» заполнена”  } |

1. **Добавление дисциплины вручную** 
   1. Контракты
      1. PutDisciplineByHand\_In

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| string | name | Название дисциплины | true |

* + 1. PutDisciplineByHand\_Out

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| int | result\_code | 0 если операция прошла успешно,  иначе код ошибки | true |
| string | message | Текстовое описание результата | false |

* 1. Описание

|  |  |
| --- | --- |
| Метод | POST |
| URL | http://host\_name:port\_number/dean/admin/add\_discipline |
| Контракт запроса | PutDisciplineByHand\_In |
| Пример запроса | {  “name”: “Квантовая физика”  } |
| Контракт ответа | PutDisciplineByHand\_Out |
| Пример ответа | {  “result\_code”: 0  }  или  }  “result\_code”: 1114,  “message”: “Таблица «%s» заполнена”  } |

1. **Добавление группы вручную** 
   1. Контракты
      1. PutGroupByHand\_In

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| string | name | Название группы | true |
| string | start\_data | Дата начала обучения группы | true |

* + 1. PutGroupByHand\_Out

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| int | result\_code | 0 если операция прошла успешно,  иначе код ошибки | true |
| string | message | Текстовое описание результата | false |

* 1. Описание

|  |  |
| --- | --- |
| Метод | POST |
| URL | http://host\_name:port\_number/dean/admin/add\_group |
| Контракт запроса | PutGroupByHand\_In |
| Пример запроса | {  “name”: “13641/1”,  “start\_date”: “01.09.2017”  } |
| Контракт ответа | PutGroupByHand\_Out |
| Пример ответа | {  “result\_code”: 0  }  или  }  “result\_code”: 1114,  “message”: “Таблица «%s» заполнена”  } |

1. **Добавление данных из файла**
   1. Контракты
      1. PutDataFromFile\_In

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| string | type | Тип данных | true |
| string | file | Содержимое файла | true |

* + 1. PutDataFromFile\_Out

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| int | result\_code | 0 если операция прошла успешно,  иначе код ошибки | true |
| string | message | Текстовое описание результата | false |

* 1. Описание

|  |  |
| --- | --- |
| Метод | POST |
| URL | http://host\_name:port\_number/dean/admin/add\_data |
| Контракт запроса | PutDataFromFile\_In |
| Пример запроса | {  “type”: “disciplines”,  “file”: “Квантовая физика;Функциональный анализ;Теория вероятности”  } |
| Контракт ответа | PutDataFromFile\_Out |
| Пример ответа | {  “result\_code”: 0  }  или  }  “result\_code”: 1114,  “message”: “Таблица «%s» заполнена”  } |

1. **Создание учебного плана**
   1. Контракты
      1. MakeCurriculum\_In

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| string | name | Название учебного плана | true |
| int | num\_of\_sems | Количество семестров | true |
| int[] | semesters | Список повторяющихся семестров | true |
| long[] | discipline\_ids | ids дисциплин | true |

* + 1. MakeCurriculum\_Out

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| int | result\_code | 0 если операция прошла успешно,  иначе код ошибки | true |
| string | message | Текстовое описание результата | false |

* 1. Описание

|  |  |
| --- | --- |
| Метод | POST |
| URL | http://host\_name:port\_number/dean/admin/make\_curriculum |
| Контракт запроса | MakeCurriculum\_In |
| Пример запроса | {  “name”: “ИПММ\_ПМ\_1\_Магистры\_2017”,  “num\_of\_sems”: 4,  “semesters”: [1, 1, 2, 2, 3, 3, 4, 4],  “discipline\_ids”: [14, 12, 24, 21, 32, 23, 34, 94]  } |
| Контракт ответа | MakeCurriculum\_Out |
| Пример ответа | {  “result\_code”: 0  }  или  }  “result\_code”: 1114,  “message”: “Таблица «%s» заполнена”  } |

1. **Изменение учебного плана**
   1. Контракты
      1. UpdateCurriculum\_In

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| int | curriculum\_id | id учебного плана | true |
| string | name | Название учебного плана | true |
| int[] | semesters | Список повторяющихся семестров | true |
| long[] | discipline\_ids | ids дисциплин | true |

* + 1. UpdateCurriculum\_Out

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| int | result\_code | 0 если операция прошла успешно,  иначе код ошибки | true |
| string | message | Текстовое описание результата | false |

* 1. Описание

|  |  |
| --- | --- |
| Метод | PUT |
| URL | http://host\_name:port\_number/dean/admin/modify\_curriculum |
| Контракт запроса | MakeCurriculum\_In |
| Пример запроса | {  “curriculum\_id”: 35  “name”: “ИПММ\_ПМ\_1\_Магистры\_2017”,  “semesters”: [1, 1, 2, 2, 3, 3, 4, 4],  “discipline\_ids”: [14, 12, 24, 55, 32, 23, 34, 94]  } |
| Контракт ответа | MakeCurriculum\_Out |
| Пример ответа | {  “result\_code”: 0  }  или  }  “result\_code”: 1114,  “message”: “Таблица «%s» заполнена”  } |

1. **Назначение учебного плана на группу**
   1. Контракты
      1. LinkGroupAndCurriculum\_In

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| long | group\_id | id группы | true |
| long | curriculum\_id | id учебного плана | true |

* + 1. LinkGroupAndCurriculum\_Out

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| int | result\_code | 0 если операция прошла успешно,  иначе код ошибки | true |
| string | message | Текстовое описание результата | false |

* 1. Описание

|  |  |
| --- | --- |
| Метод | POST |
| URL | http://host\_name:port\_number/dean/admin/link\_group\_curriculum |
| Контракт запроса | LinkGroupAndCurriculum\_In |
| Пример запроса | {  “group\_id”: 145785,  “curriculum\_id”: 125748  } |
| Контракт ответа | LinkGroupAndCurriculum\_Out |
| Пример ответа | {  “result\_code”: 0  }  или  }  “result\_code”: 1114,  “message”: “Таблица «%s» заполнена”  } |

1. **Назначение преподавателя на чтение дисциплины**
   1. Контракты
      1. LinkGroupDiscpAndTeacher\_In

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| long | group\_id | id группы | true |
| long | discipline\_id | id дисциплины | true |
| long | teacher\_id | id преподавателя |  |

* + 1. LinkGroupDiscpAndTeacher\_Out

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| int | result\_code | 0 если операция прошла успешно,  иначе код ошибки | true |
| string | message | Текстовое описание результата | false |

* 1. Описание

|  |  |
| --- | --- |
| Метод | POST |
| URL | http://host\_name:port\_number/dean/admin/link\_group\_disc\_teach |
| Контракт запроса | LinkGroupDiscpAndTeacher\_In |
| Пример запроса | {  “group\_id”: 145785,  “discipline\_id”: 125748,  “teacher\_id”: 746  } |
| Контракт ответа | LinkGroupDiscpAndTeacher\_Out |
| Пример ответа | {  “result\_code”: 0  }  или  }  “result\_code”: 1114,  “message”: “Таблица «%s» заполнена”  } |

1. **Просмотр промежуточных итогов** 
   1. Контракты
      1. GetTempMarks\_Out

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| int[][] | temp\_marks | Матрица с оценками | true |
| string[] | rownames | Список студентов | true |
| string[] | colnames | Список дат занятий | true |

* 1. Описание

|  |  |
| --- | --- |
| Метод | GET |
| URL | http://host\_name:port\_number/dean/common/current\_grades/group/id14/discipline/id1 |
| Контракт запроса |  |
| Пример запроса |  |
| Контракт ответа | GetTempMarks\_Out |
| Пример ответа | {  “temp\_marks”: [  [5, 3, 2],  [5, 2, 3],  [4, 4, 4]  ],  “rownames”: [  “Коновалов К.”,  “Гарбузов Ф.”,  “Ковалев М.”  ],  “colnames”: [  “08.02.2018”,  “15.02.2018”,  “22.02.2018”  ]  } |

1. **Просмотр успеваемости**
   1. Контракты
      1. GetAllStudMarks\_Out

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| int[] | marks | Массив оценок | true |
| string[] | disciplines | Список предметов | true |
| int[] | semesters | Массив из повторяющихся семестров | true |

* 1. Описание

|  |  |
| --- | --- |
| Метод | GET |
| URL | http://host\_name:port\_number/dean/student/all\_grades/id145785 |
| Контракт запроса |  |
| Пример запроса |  |
| Контракт ответа | GetAllStudMarks\_Out |
| Пример ответа | {  “temp\_marks”: [5, 5, 3, 4, 5, 4, 3],  “disciplines”: [  “Математический анализ”,  “Аналитическая геометрия”,  “Механика”,  “Молекулярная физика”,  “Атомная физика”,  “Квантовая физика”,  “Функциональный анализ”  ],  “semesters”: [1, 1, 2, 2, 3, 3, 3]  } |

1. **Создание ведомости по определенной дисциплине**
   1. Контракты
      1. DisciplineReport\_In

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| long | group\_id | id группы | true |
| long | discipline\_id | id дисциплины | true |

* + 1. DisciplineReport\_Out

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Field | Description | Req |
| int[] | marks | Массив оценок | true |
| string[] | students | Список студентов | true |
| string | discipline | Название дисциплины | true |
| string | teacher | ФИО преподавателя | true |
| string | group | Номер группы | true |
| int | semester | Семестр обучения | true |
| string | start\_date | Дата начала обучения группы | true |

* 1. Описание

|  |  |
| --- | --- |
| Метод | POST |
| URL | http://host\_name:port\_number/dean/teacher/dreport |
| Контракт запроса | DisciplineReport\_In |
| Пример запроса | {  “group\_id”: 145785,  “discipline\_id”: 125748  } |
| Контракт ответа | DisciplineReport\_Out |
| Пример ответа | {  “marks”: [5, 5, 3],  “students”: [  “Ковалев М.”,  “Гарбузов Ф.”,  “Коновалов К.”  ],  “discipline”: “UML”  “teacher”: “Новиков Ф.А.”  “group”: “13641/1”  “semester”: “1”  “start\_date”: “01.09.2017”  } |