

Министерство образования и молодежной политики Свердловской области  
Государственное автономное профессиональное образовательное учреждение  
Свердловской области  
«Камышловский педагогический колледж»

**Проектирование информационной системы  
«Расписание занятий «ГАПОУ СО «Камышловский  
педагогический колледж»**

специальность: 09.02.07 Информационные системы и программирование

**Исполнитель:**  
Човнюк М.А.,  
студент 3<sup>II</sup> группы  
**Руководитель:**  
Черноскутов А.С.,  
преподаватель

Камышлов, 2021

## ОГЛАВЛЕНИЕ

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА.....	3
Введение.....	3
Постановка задачи.....	6
ИССЛЕДОВАНИЕ ЗАДАЧИ.....	7
3.1. Описание предметной области.....	7
3.2. Создание функциональной модели ПО.....	9
3.3. Создание информационной модели предметной области.....	10
3.4. Моделирование потоков данных (выделение бизнес-процессов).....	11
3.5. Определение целей, задач и функций проектируемой ИС.....	15
3.6. Анализ существующих аналогов.....	17
РАЗРАБОТКА ЗАДАЧИ.....	21
4.1. Проектирование базы данных.....	21
4.1.1. Создание концептуальной модели.....	21
4.1.2. Создание логической модели.....	25
4.1.3. Составление глоссариев и спецификаций.....	31
4.1.4. Создание физической модели БД.....	35
4.2. Проектирование и реализация приложений конечных пользователей.....	41
4.2.1. Выбор среды разработки приложения для ИС.....	41
4.2.2. Разработка интерфейса прототипа приложения.....	42
4.2.2.1. Расписание.....	43
4.2.2.2. Преподаватели.....	51
4.2.2.3. Группы.....	59
4.2.2.4. Рабочие программы.....	67
4.2.2.5. Аудитории.....	79
4.2.2.6. Справка.....	85
ТЕСТИРОВАНИЕ ПРОГРАММНОГО ПРОДУКТА.....	87
5.1. Тестирование по классам эквивалентности входных данных.....	87
ПРАКТИЧЕСКАЯ ЧАСТЬ.....	96
ЗАКЛЮЧЕНИЕ.....	97
СПИСОК ИСТОЧНИКОВ И ЛИТЕРАТУРЫ.....	98

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

### Введение

В современном мире информация становится фактором, определяющим эффективность любой сферы деятельности. Возрастают информационные потоки, повышаются требования к скорости обработки данных. Большинство операций теперь уже не может быть выполнено вручную, они требуют применения передовых компьютерных технологий, в которых большую роль играют системы хранения и обработки информации.[1]

Современная образовательная система требует оптимизации процесса обучения в вузах. Инновации актуальны в связи с повсеместным расширением и популяризацией образования. Поэтому, организация оптимального учебного расписания является особо острой для эффективного планирования образовательного процесса. Создание общей базы данных решает множество проблем с аудиторным фондом, профессорским составом и планировании времени студентов. [3]

Из вышесказанного вытекает **проблема**, актуальная для современной системы образования: необходимость автоматизации процесса составления расписания и других операций над расписанием средствами автоматизированных информационных систем (АИС) или баз данных.

Поэтому актуальным является рассмотрение **темы** проектирования информационной системы «Расписание занятий «ГАПОУ СО «Камышловский педагогический колледж»».

Любое образовательное учреждение в ходе своей деятельности составляет расписание занятий. Студенты и преподаватели «ГАПОУ СО «Камышловского педагогического колледжа» (далее колледж) должны знать свое расписание занятий. Расписание составляется заведующим очным

отделением колледжа вручную, заполняя специальную таблицу в документе Word, Данный процесс составления расписаний является очень трудоемким. В нем необходимо учитывать огромное количество ограничений. Решить данную проблему позволит информационная система «Расписание занятий «ГАПОУ СО «Камышловский педагогический колледж»».

Следовательно, объектом моего курсового проекта является АИС, обеспечивающая автоматизацию процесса составления расписания и других операций над расписанием, для колледжа, предметом – модель АИС «Расписание занятий «ГАПОУ СО «Камышловский педагогический колледж»».

**Цель исследования:** проектирование информационной системы «Расписание занятий «ГАПОУ СО «Камышловский педагогический колледж»»

**Задачи исследования:**

- Исследование предметной области разрабатываемого проекта информационной системы «Расписание занятий «ГАПОУ СО «Камышловский педагогический колледж»»
- Проведение анализа понятия «информационная система» и технических и программных средств для ее создания.
- Разработка и описание основных элементов модели информационной системы «Расписание занятий «ГАПОУ СО «Камышловский педагогический колледж»»

Практическая значимость курсового проекта состоит в возможности проектирование информационной системы, которая позволит разгрузить заведующего очного отделения и использовать современные цифровые технологии для работы с расписанием. Исследование может представлять интерес для студентов, обучающихся по специальности 09.02.07

«информационные системы и программирование», при изучении профессиональных модулей «разработка, администрирование и защита баз данных» и «Разработка программных модулей программного обеспечения для компьютерных систем», а также преподавателей, работающих на данной специальности.

## Постановка задачи

### Составление расписания

Разработать АИС, которая должна содержать сведения...

- О группах: курс и литерал;
- О графике обучения группы;
- О преподавателях: фамилия, имя, отчество, опция плотного графика, логин и пароль;
- О графике работы преподавателя;
- О заявках преподавателя;
- О аудиториях: наименование аудитории;
- О рабочих программах: Полное и краткое наименование рабочей программы, общее количество требуемых к реализации академических и самостоятельных часов, группу к которой привязывается рабочая программы;
- О дисциплинах: даты начала и конца реализации дисциплины, количество требуемых к реализации академических и самостоятельных часов, рабочую программу к которой привязывается дисциплина, семестр и подгруппа;
- О занятиях: дата и время начала занятия, количество реализованных часов в занятии, дисциплина, самостоятельные или академические часы, аудитория, в которой проводится занятие.

АИС должна обеспечивать возможность автоматического формирования отчета о выполнении УП на основе данных из базы данных, а также обеспечивать возможность печати расписания на неделю.

# ИССЛЕДОВАНИЕ ЗАДАЧИ

## 3.1. Описание предметной области

Проектирование любой АИС начинается с изучения предметной области, вот что мне удалось найти на тему «составление расписания»:

В учебный отдел перед наступлением учебного периода поступают рабочие программы от заместителя директора по научно-методической и исследовательской работе. В программе указано:

- Непосредственно название рабочей программы (название предмета, для отображения в расписании);
- Для какой группы\подгруппы предназначена программа;
- Сколько аудиторных часов необходимо выдать группе;
- Сколько самостоятельных работ выдать группе;
- Какие преподаватели должны вести этот предмет и для какой группы\подгруппы;
- В какие сроки необходимо выдать все академические часы (за год, за семестр, за четверть и т.д.);
- Дополнительные требования для аудиторий.

На основе этих данных необходимо составлять расписания (на неделю, на день) так, чтобы:

- Все рабочие программы выдавались в срок;
- У преподавателей не было одновременных пар;
- В 1 аудитории не было нескольких групп, и аудитория должна быть подготовлена для таких занятий;
- Пары назначались преподавателю и группе в пределах их рабочего расписания;
- Также необходимо учитывать, что аудиторный час может быть выдан не группе, а подгруппе.

Длительность академического часа можно считать глобальным параметром, который будет редко изменяться.

Любой преподаватель в любой момент ввиду форс-мажора может попросить учебный отдел не назначать занятий в некоторый промежуток времени (либо изменить рабочее расписание).

Составленное по всем требованиям расписание распечатывается и вывешивается у вахты, чтобы все учащиеся\преподаватели могли его увидеть и ознакомиться с ним.

Ввиду особых обстоятельств занятие может не состояться (занятие сорвано, преподаватель заболел и т.д.).

По ходу учебного процесса директор учебного заведения может запросить отчёт, в котором будет отражаться степень выдачи рабочих программ.

Также в период обучения необходимо выдавать практику для студентов, заранее определить место для практики в расписании нельзя, заранее определяется только неделя, когда ее необходимо выдать. Практики бывают учебные и производственные. Часть практик может быть концентрированной (займет всю неделю\день), а другая часть рассредоточенной (смешана по дням\парам). Практики назначает руководитель практики, и он же передает эту информацию в учебный отдел.



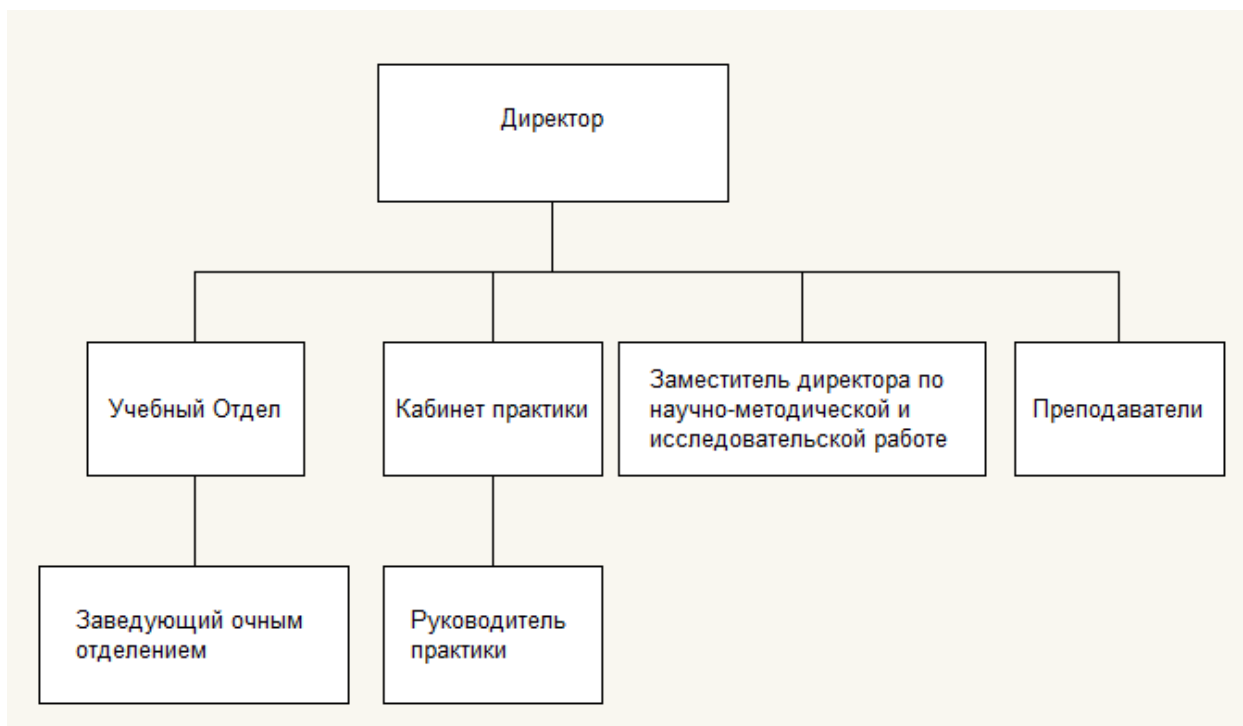


Рисунок 1. Организационная схема.

### 3.2. Создание функциональной модели ПО.

Преподаватели, сформировав рабочую программу, передают ее заместителю по УИРС для утверждения. Заместитель может вернуть преподавателю рабочую программу для доработки.

Расписание составляется на основе утверждённых рабочих программ (передает заместитель по УИРС) и назначенных практик (назначает руководитель практики), полученных перед началом семестра.

По запросу директора формируется отчет выполнения учебных программ, в котором отражается соотношение количества выданных академических часов к общему числу, требуемых к выдаче.

Преподаватели ввиду форс-мажоров могут написать заявку для корректировки своего расписания.

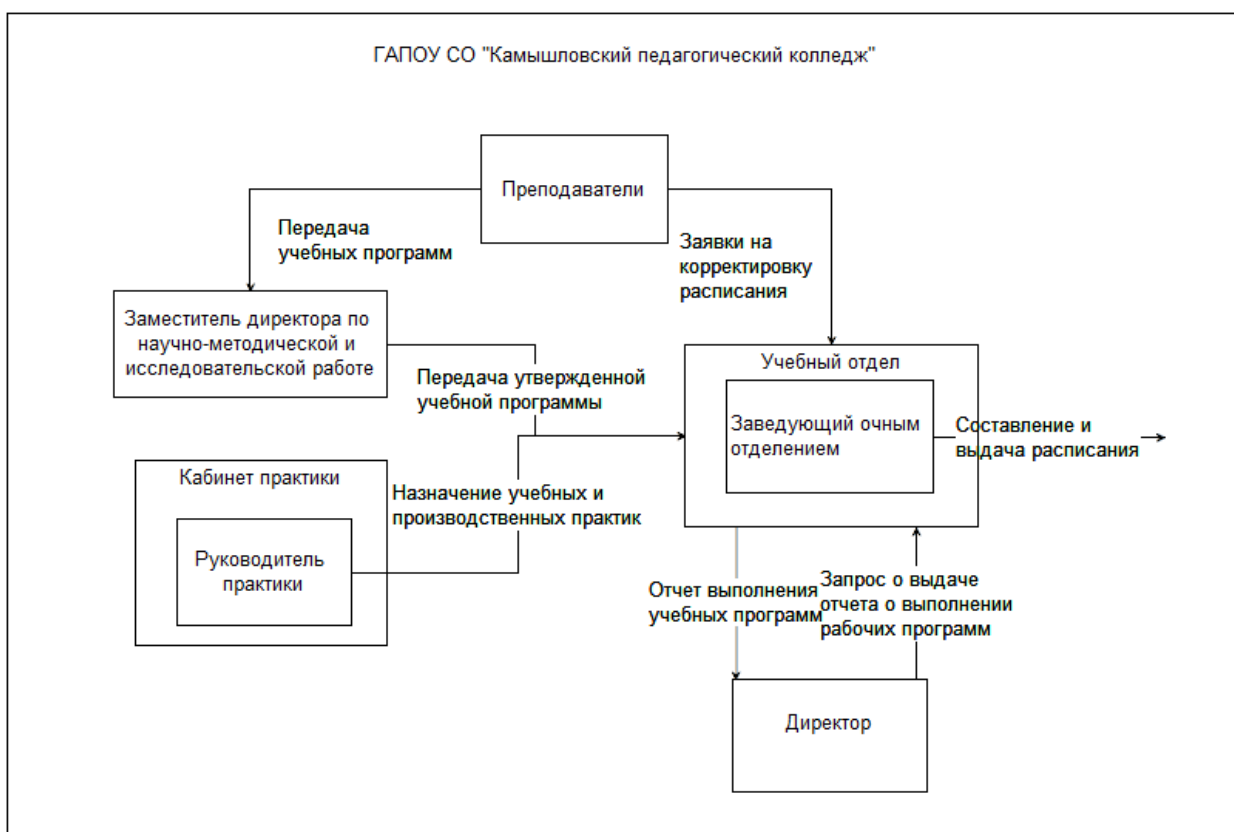


Рисунок 2. Функциональная схема.

### 3.3. Создание информационной модели предметной области.

Рабочие программы, необходимые для составления расписаний, формируют преподаватели, после рабочей программа передается заместителю по УИРС для утверждения. Заместитель может вернуть преподавателю рабочую программу для доработки.

Расписание составляется на основе утверждённых рабочих программ (передает заместитель по УИРС) и назначенных практик (назначает руководитель практики), полученных перед началом семестра.

По запросу директора формируется отчет выполнения учебных программ, в котором отражается соотношение количества выданных академических часов к общему числу, требуемых к выдаче.

Преподаватели ввиду форс-мажоров могут написать заявку в произвольной форме (телефонный звонок, по электронной почте и т.д.) для корректировки своего расписания.

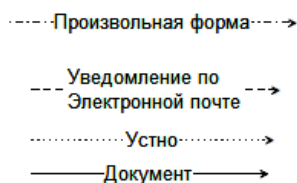
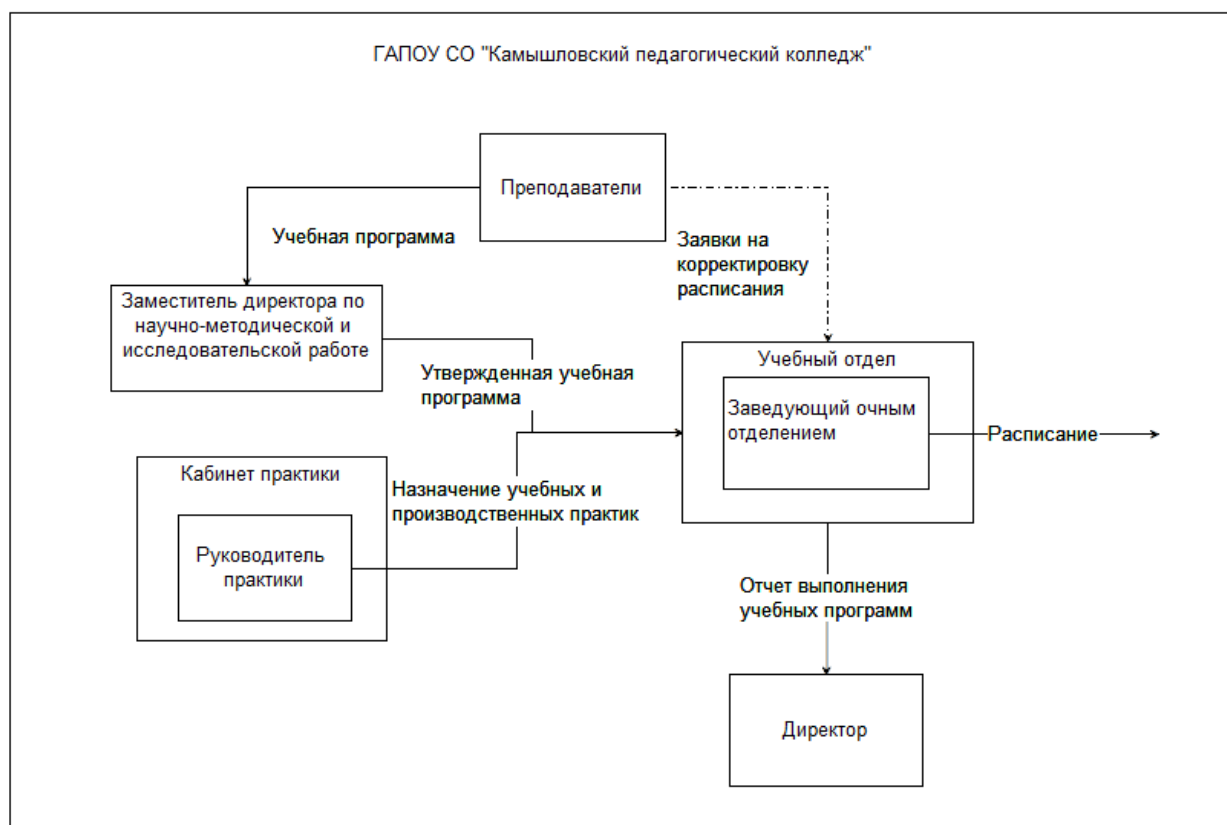


Рисунок 3. Информационная схема.

### 3.4. Моделирование потоков данных (выделение бизнес-процессов)

Передо мной стояла задача сформировать модели бизнес-процессов до реинжиниринга и после с помощью DiaPortable. В первой модели я описал процесс «составления расписания», происходящие в моей предметной области до введения будущей информационной системы (Рис 1). В этой модели также отражена ключевая роль в заведующим очным отделением, на которого возложена большая рабочая нагрузка ввиду составления расписания. Для составления расписания заведующему очным отделением требуются данные:

- О рабочих программах, которые передает заместитель директора по

научно-методической и исследовательской работе;

- О расписании практик. Их назначает руководитель практики, он же и передает данные заведующему очным отделением;
- Заявки о корректировке расписания. Заявки оформляются преподавателями, заведующий очным отделением учитывает их при составлении расписания.

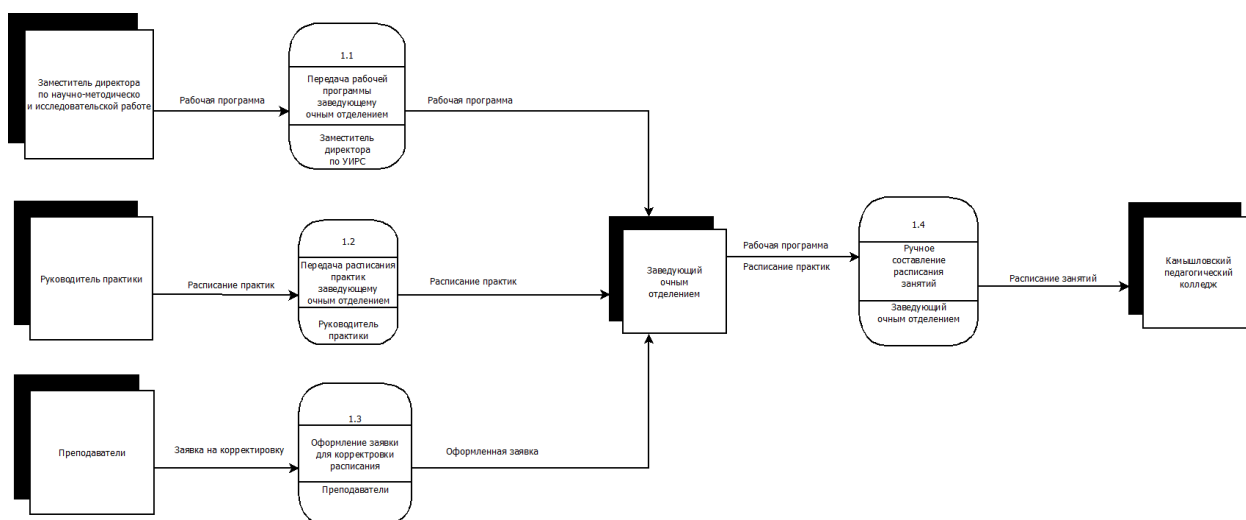


Рисунок 4. Модель бизнес-процессов №1 «Составление расписания» по методологии DFD до реинжиниринга.

Далее я смоделировал другой бизнес-процесс «Составление отчета» (Рис 2). Его суть состоит в следующем: по мере составления расписания выдаются академические часы, директор в любой момент может потребовать отчет о выдаче академических часов, чтобы убедиться в том, что выдача рабочих программ выполняется своевременно.

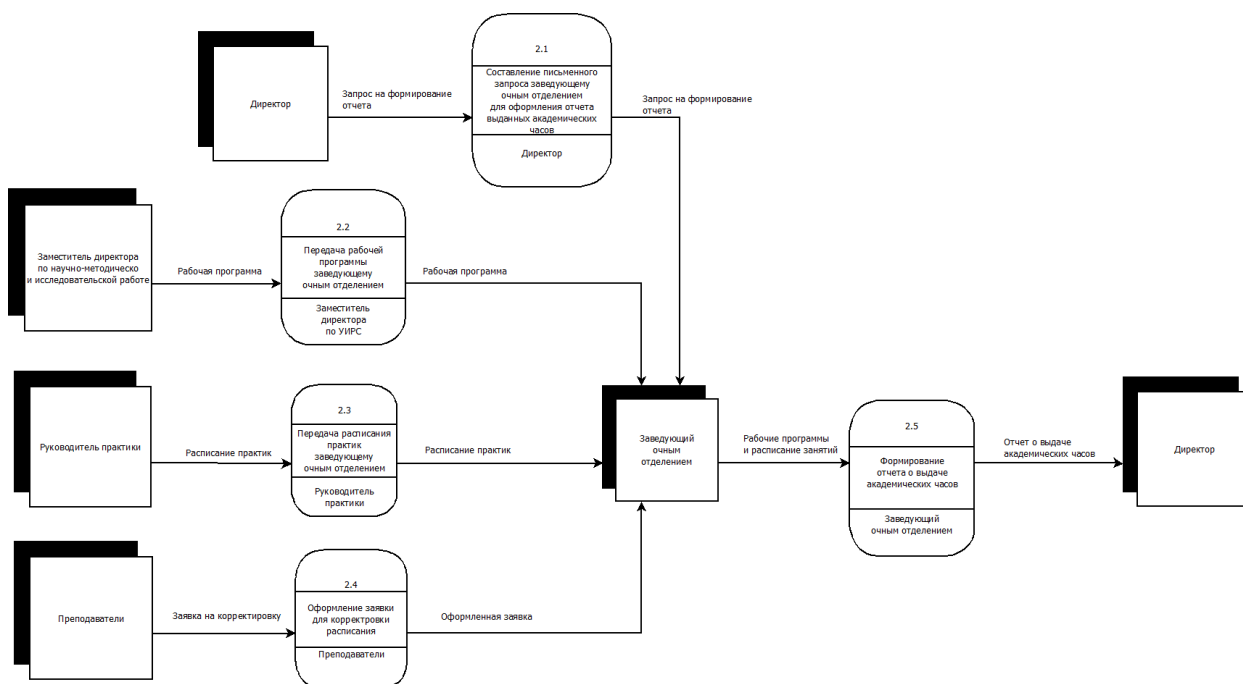


Рисунок 5. Модель бизнес-процессов №2 «Составление отчета» по методологии DFD до реинжиниринга.

Далее я сформировал модель бизнес-процесса «Составление расписания» после реинжиниринга с помощью программы DiaPortable (Рис 3). В ранее описанном процессе я включил работу с базой данных для уменьшения круговорота «бумажной» работы. Помимо этого, внедрение работы с базой данных поможет более эффективно хранить, искать и обрабатывать данные. Непосредственно сам процесс составления расписания будет уже не таким сложным, перегруженным.

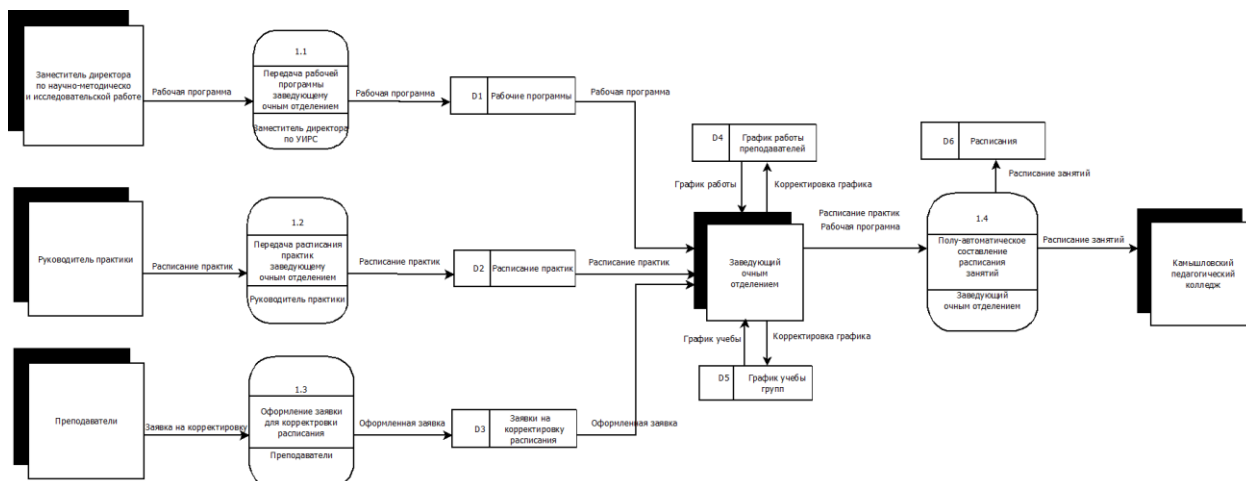


Рисунок 6. Модель бизнес-процесса №1 «Составление расписания» по методологии DFD после реинжиниринга.

После я сформировал модель бизнес-процесса «Формирование отчета» после реинжиниринга с помощью программы DiaPortable (Рис 4). В ранее описанном процессе я автоматизировал формирование отчета, т.е. теперь не потребуется участие заведующего очным отделением для формирования отчета, отчет будет формироваться автоматически, основываясь на данных в базе данных.

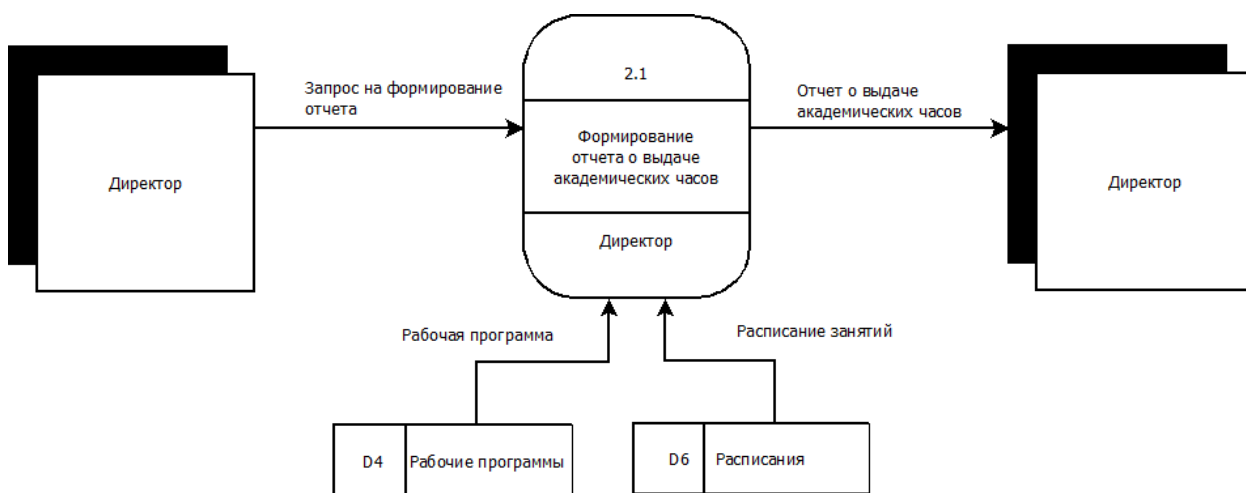


Рисунок 7. Модель бизнес-процесса №2 «Формирование отчета» по методологии DFD после реинжиниринга.

### **3.5. Определение целей, задач и функций проектируемой ИС**

Для определения целей, задач и функций проектируемой ИС использовалась методология функционального моделирования SADT.

Передо мною стояла задача в реализации SADT-модель с помощью программы DiaPortable. Сначала я смоделировал SADT-модель нулевого уровня (Рис. 8), в которой я отразил входящие данные, которые потребуются для работы моей будущей информационной системы. Также отразил управляющую информацию, от которой зависит то, как будет составляться расписание. Среди пользователей информационной системы я ожидаю, что системой будут пользоваться:

- Преподаватели - просматривают расписание и составляют заявки на корректировку расписания;
- Заведующий очного отделения – составляет расписание, корректирует график работы преподавателей, график учебы групп, а также управляет состоянием аудиторий, отраженных в базе данных;
- Студенты – просматривают расписание;
- Директор – формирует запрос на создание отчета о выдаче академических часов.



Рисунок 8. Функциональная модель нулевого уровня по методологии SADT

Далее я приступил к более детальному моделированию информационной системы (Рис 9). В ней будет существовать 4 подсистемы. Первая подсистема будет самой важной, отвечать за составление расписания, управлять этой подсистемой будет заведующий очным отделением. В качестве исходной информации будут служить рабочие программы, расписание практик, т.к. именно от них напрямую зависит процесс составления расписания. В качестве управляющей информации будут служить длительность академического часа, график работы учебного учреждения и заявки на корректировку расписания, т.к. они определяют ограничения, которым нужно следовать при составлении расписания. Помимо этой подсистемы существуют еще 3 подсистемы.

Подсистема формирования заявок на корректировку расписания служит для преобразования заявок преподавателей в необходимую форму, пригодную для хранения в базе данных и для автоматического создания ограничения влияющего на процесс составления расписания.

Подсистема просмотра расписания отвечает за просмотр составленного расписания в удобной форме для целевого пользователя (студенту или преподавателю).

Подсистема формирования отчета о выдаче академических часов служит для автоматической генерации отчета о выдаче академических часов



на основе данных о составленном расписании, которое храниться в базе данных. Ожидается что данной подсистемой будет пользоваться директор.

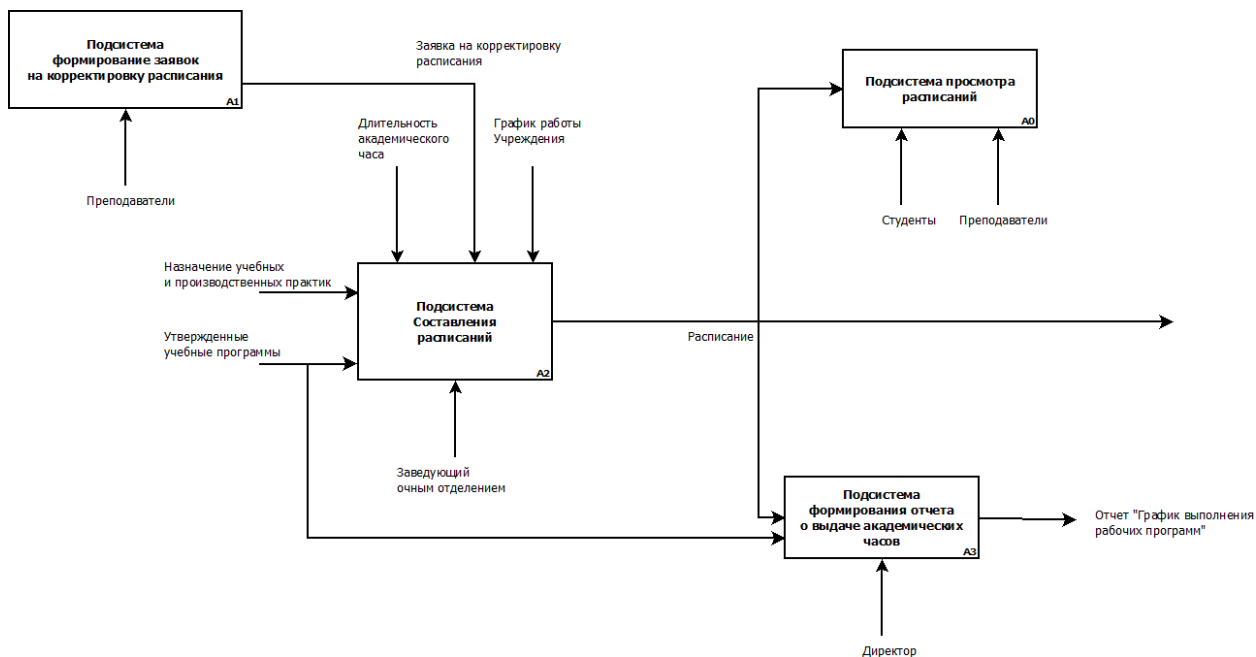


Рисунок 9. Функциональная модель первого уровня по методологии SADT.

### 3.6. Анализ существующих аналогов

Заведующий очным отделением не использует ИС в составлении расписания, он вручную заполняет заранее заготовленную таблицу.

Существует ряд программ для автоматизации составления расписания:

#### 1. FET – Free Timetabling Software

Данная программа не самая многофункциональная, но достаточно мощная. Она позволит создать грамотное и понятное школьное расписание. У программы достаточно узкая специфика. То есть, целесообразно использовать приложение учителям и сотрудникам школ и других учебных заведений.

Программа позволяет назначать дисциплины группам, а также преподавателей, которые закреплены за этими дисциплинами. После заполнения данных о группах, дисциплинах, преподавателях и настройки особенностей расписания (поток, начало рабочего дня, длительность уроков и т.д.) можно автоматически составить расписание.

С этой программой справится и обычный учитель. Нет никакой необходимости в системном администраторе.

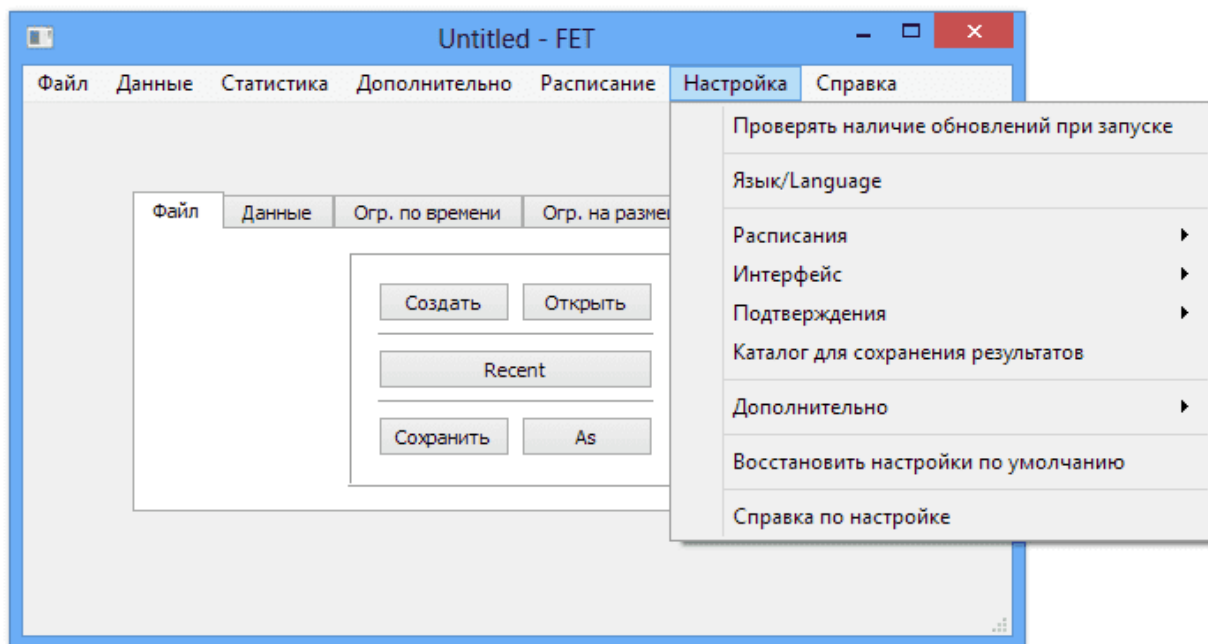


Рисунок 10. Программа FET – Free Timetabling Software.

Данная программа имеет один весомый минус – она является однопользовательской системой, т.е. первичные данные вводит тот же человек (учитель), что и генерирует расписание. Расписание передается все также учителю. После ему необходимо отдельно заняться распространением расписания и самостоятельно обрабатывать исключительные ситуации.

## 2. 1С: Автоматизированное составление расписания. Школа

Программа предназначена для составления расписания, индивидуальных траекторий и дополнительной занятости учащихся с учетом основного расписания и аудиторного фонда в школах. Гибкие настройки программы позволяют успешно использовать ее в учреждениях со сложной структурой: образовательные комплексы, объединяющие детские сады, школы; центры творчества и дополнительного образования детей; частные школы или центры развития с индивидуальным графиком для каждого ребенка.

Данная программа также имеет богатый функционал, но она также является однопользовательской.

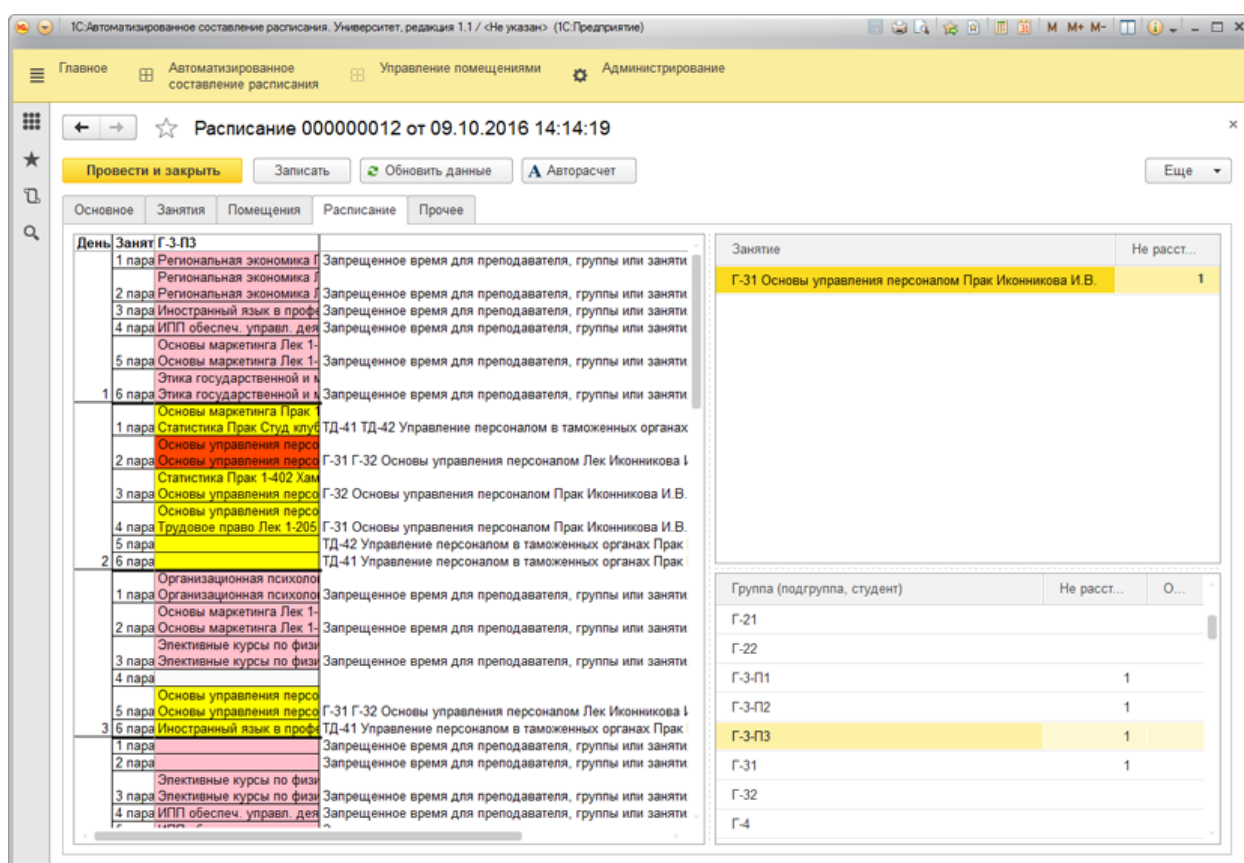


Рисунок 11. Скриншот программы 1С: Автоматизированное составление расписания. Школа

### 3. Экспресс-расписание Колледж Полная

В программе можно вести расписание как очного, так и заочного отделения, учет практик, консультаций, экзаменов, учет замещения преподавателей, передачу нагрузки другому преподавателю в течение семестра, назначение занятий без основного расписания (в текущем).

При автоматическом составлении основного расписания учитываются запреты — графики преподавателей совместителей, методические дни и т. п. Полученное расписание можно корректировать вручную. Также можно назначить занятия вручную, а затем досоставить расписание автоматически.

Данное приложение аналогично другим является однопользовательской.

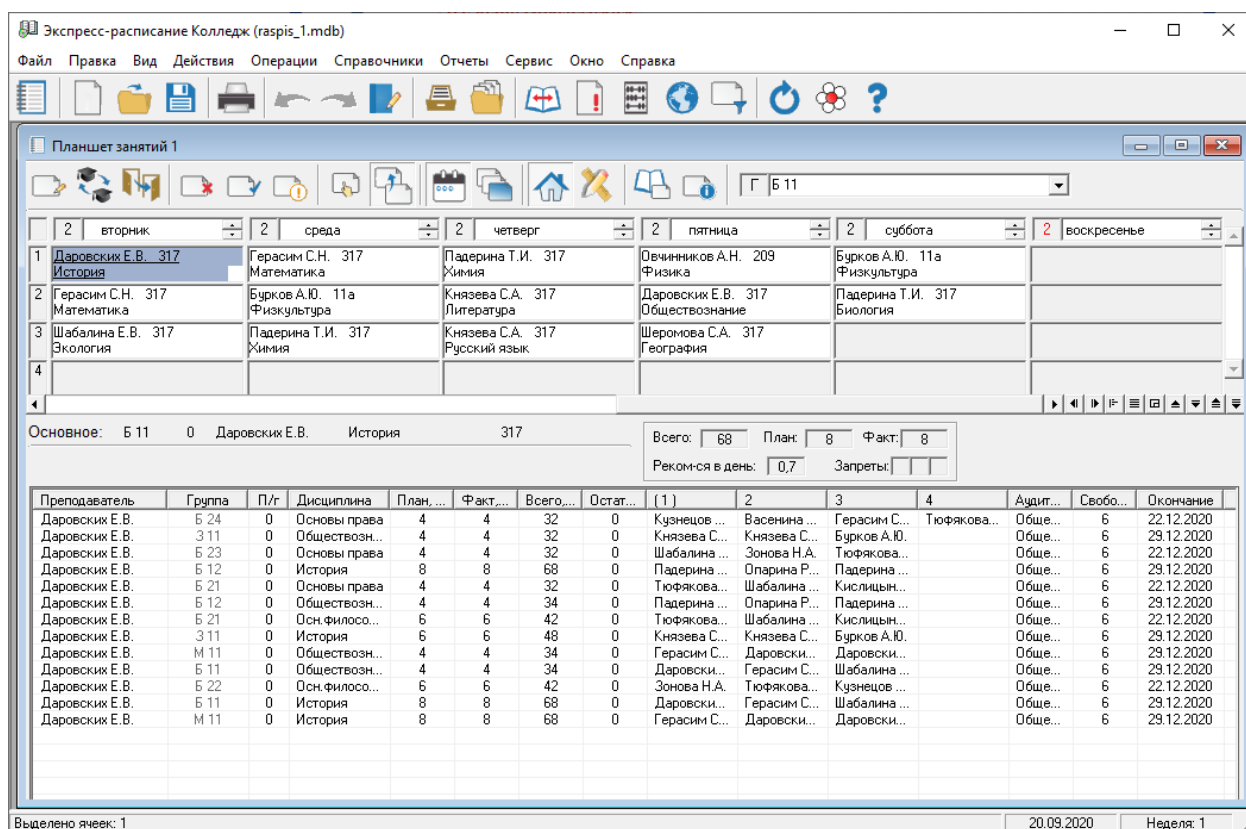


Рисунок 12. Скриншот программы Экспресс-расписание Колледж Полная.

## РАЗРАБОТКА ЗАДАЧИ

### 4.1. Проектирование базы данных

#### 4.1.1. Создание концептуальной модели

Мне предстояло создать концептуальную модель моей будущей базы данных. Для этого я сначала для моделирования выбрал нотацию Чена, после приступил к моделированию самой модели (Рис. 13).

Начну описывать модель с сущности «Группа». Данная сущность отражает группу учащихся в колледже, потому в базе данных необходимо хранить ее буквенный литерал, численный литерал (для автоматического перевода на следующий курс).

Сущность «Группа» имеет идентифицирующую связь с сущностью «График учебы группы». Данная сущность отражает промежутки времени в недели, в которых может обучаться группа. В совокупности эти промежутки формируют график учебы группы. Из вышесказанного, очевидно, что в базе данных необходимо хранить внешний ключ группы, собственный идентификатор, начало и конец временного промежутка, и день недели, на который распространяется промежуток времени.

Сущность «Группа» также имеет связь с сущностью «Рабочая программа». Данная сущность отражает рабочую программу для некоторой группы, которую передает заместитель директора по УИРС заведующему очным отделением. В рабочей программе указаны важные данные для составления расписания такие как: количество академических часов, количество самостоятельных часов, наименование рабочей программы и, собственно, для какой группы предназначена рабочая программа.

Сущность «Рабочая программа» имеет идентифицирующую связь с сущностью «Дисциплина», в ней указываются, какие преподаватели ведут занятия для конкретных подгрупп и в каком семестре. Помимо этого, в

данной сущности отражен временной диапазон, в пределах которого необходимо выдать некоторую часть аудиторных и самостоятельных часов дисциплины. Для каждого семестра и подгруппы\группы создается отдельная дисциплина, в которой описывается аудиторные и самостоятельные часы. Эти поля потребуются для составления расписания в рамках семестра. Отдельно необходимо проверять, чтобы сумма академических часов по семестрам совпадала с значением в рабочей программе.

Сущность «Преподаватель» отражает реального преподавателя в колледже, который способен вести некоторые дисциплины. Для моей информационной системы необходимо хранить в базе данных имя, фамилию и отчество преподавателя, а также логин и пароль для каждого преподавателя.

Сущность «преподаватель» имеет идентифицирующую связь с сущностью «График работы преподавателя». Данная сущность отражает график работы аналогично сущности «График учебы группы», потому имеет схожие атрибуты.

Сущность «Преподаватель» также имеет идентифицирующую связь с сущностью «Заявка преподавателя». Данная сущность отражает заявки преподавателей, в которых может быть указано, что преподаватель не сможет провести занятие в следующий понедельник. Заведующему очным отделением необходимо сначала рассмотреть заявку преподавателя, а после решить удовлетворить ее или отказать. Из вышесказанного, становится очевидно, что в базе данных необходимо хранить текстовое описание заявки, внешний ключ преподавателя, собственный идентификатор, метку об удовлетворенности\отказе и ряд полей, которые указывают на временной промежуток.

Сущность «Аудитория» отражает некоторый кабинет в колледже, в базе данных будет храниться только лишь его наименование.

При составлении расписания важно учитывать то, что для проведения занятий некоторой дисциплины потребуется проводить занятия только в определенном наборе аудиторий. Например, для проведения информатики требуется аудитории, в которой есть компьютеры для студентов, потому в модель я ввел сущность «Дисциплина аудитория», которая сама по себе является лишь таблицей соответствий, показывающая в каких аудиториях можно проводить занятия некоторой дисциплины.

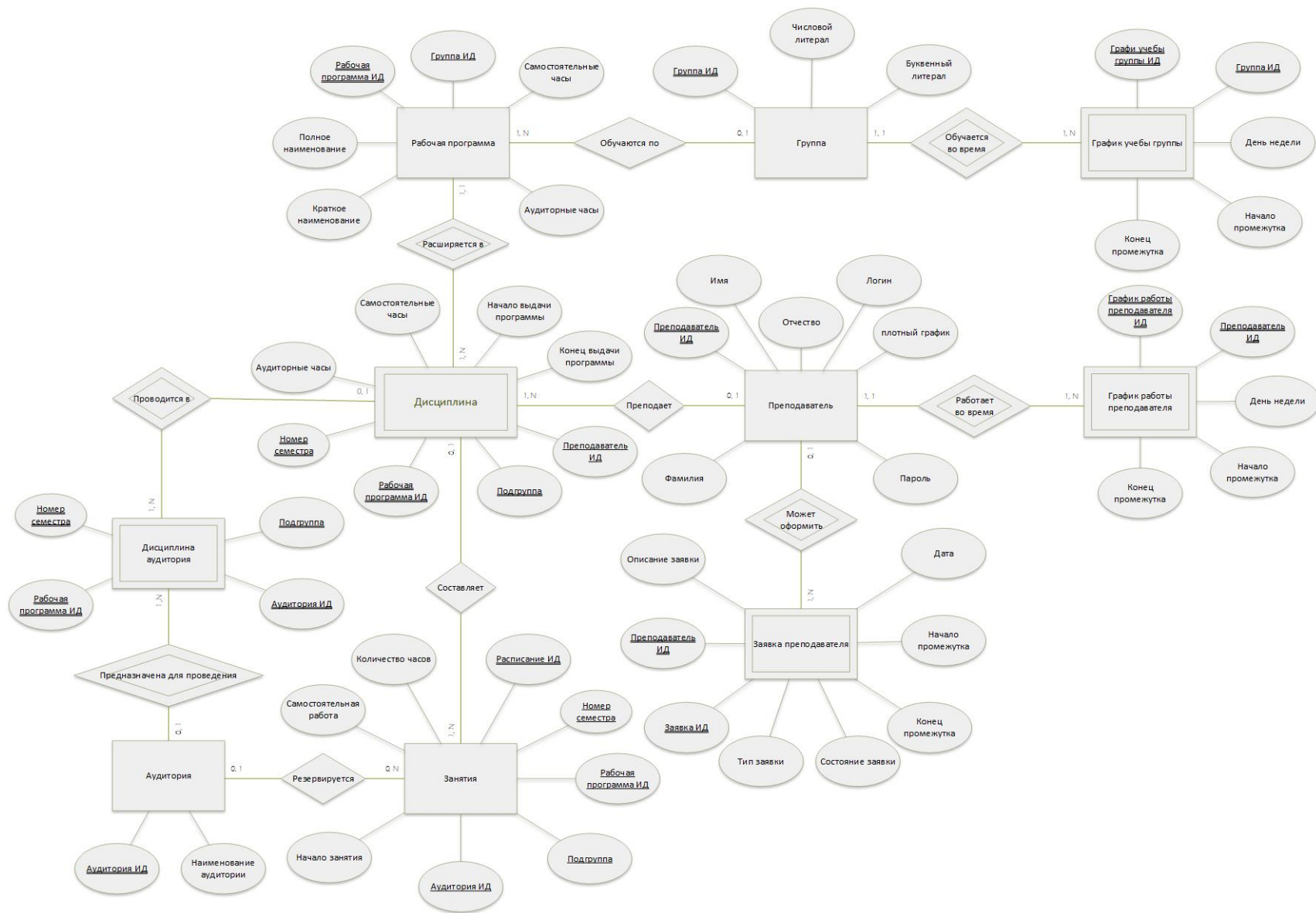


Рисунок 13. Концептуальная модель базы данных по нотации Чена.



#### **4.1.2. Создание логической модели**

На данном этапе мне предстояло создать 3 модели разных уровней: ER, KB и FA уровней по методологии IDEF1X.

Для этого я воспользовался программой Microsoft Visio, в ней выбрал тип документа «нотация IDEF1X для моделирования баз данных», в которой создал 3 страницы для разных уровней моделирования.

Не вижу необходимости повторно пояснять сущности и атрибуты модели, т.к. описание к ним можно получить в предыдущей главе.

##### **Уровень «сущность-связь» (ER).**

В данной модели представлено описание к сущностям и их взаимосвязях по методологии IDEF1X с помощью программы Microsoft Visio Professional. Сама модель логически повторяет концептуальную схему, потому не вижу смысла детально описывать данную модель.

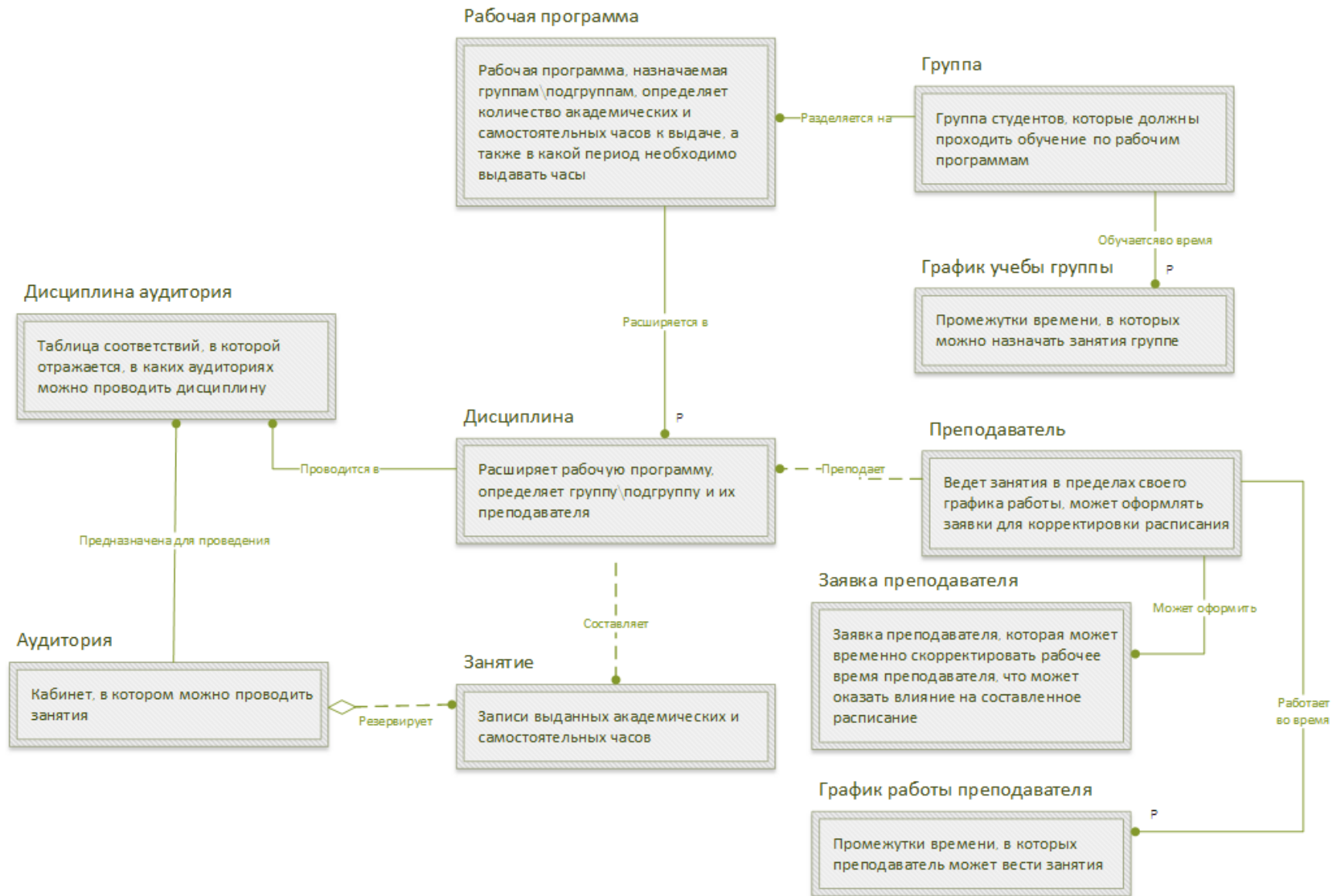


Рисунок 14. Модель базы данных ER уровня по методологии IDEF1X.

### **Уровень ключей (КВ).**

В данной модели представлено описание всех ключей сущностей по методологии IDEF1X с помощью программы Microsoft Visio Professional. Сама модель логически повторяет концептуальную схему, потому не вижу смысла детально описывать данную модель.

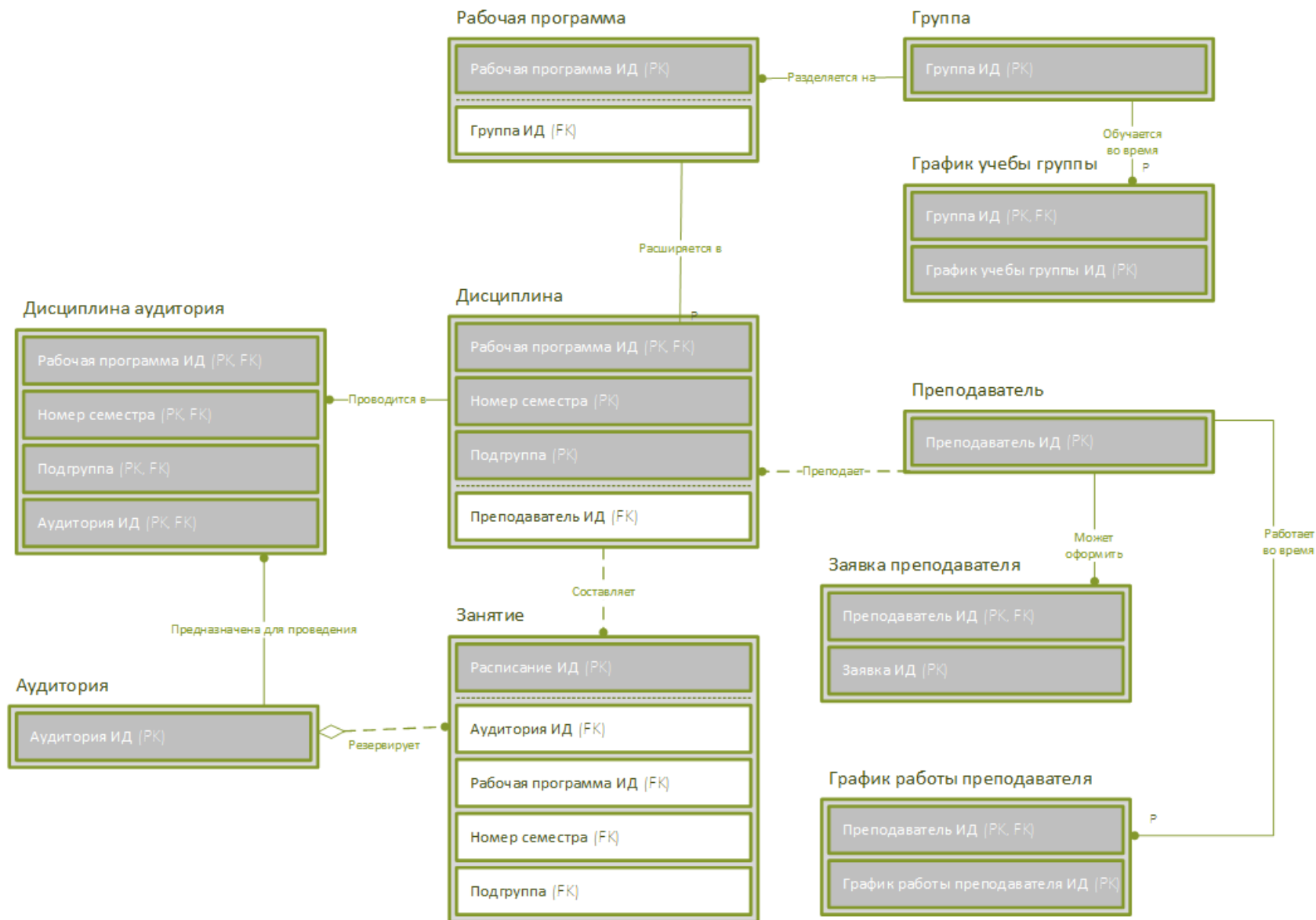


Рисунок 15. Модель базы данных КВ уровня по методологии IDEF1X.

## **Уровень атрибутов (FA).**

В данной модели представлено описание атрибутов сущностей по методологии IDEF1X с помощью программы Microsoft Visio Professional. Сама модель логически повторяет концептуальную схему, потому не вижу смысла детально описывать данную модель.

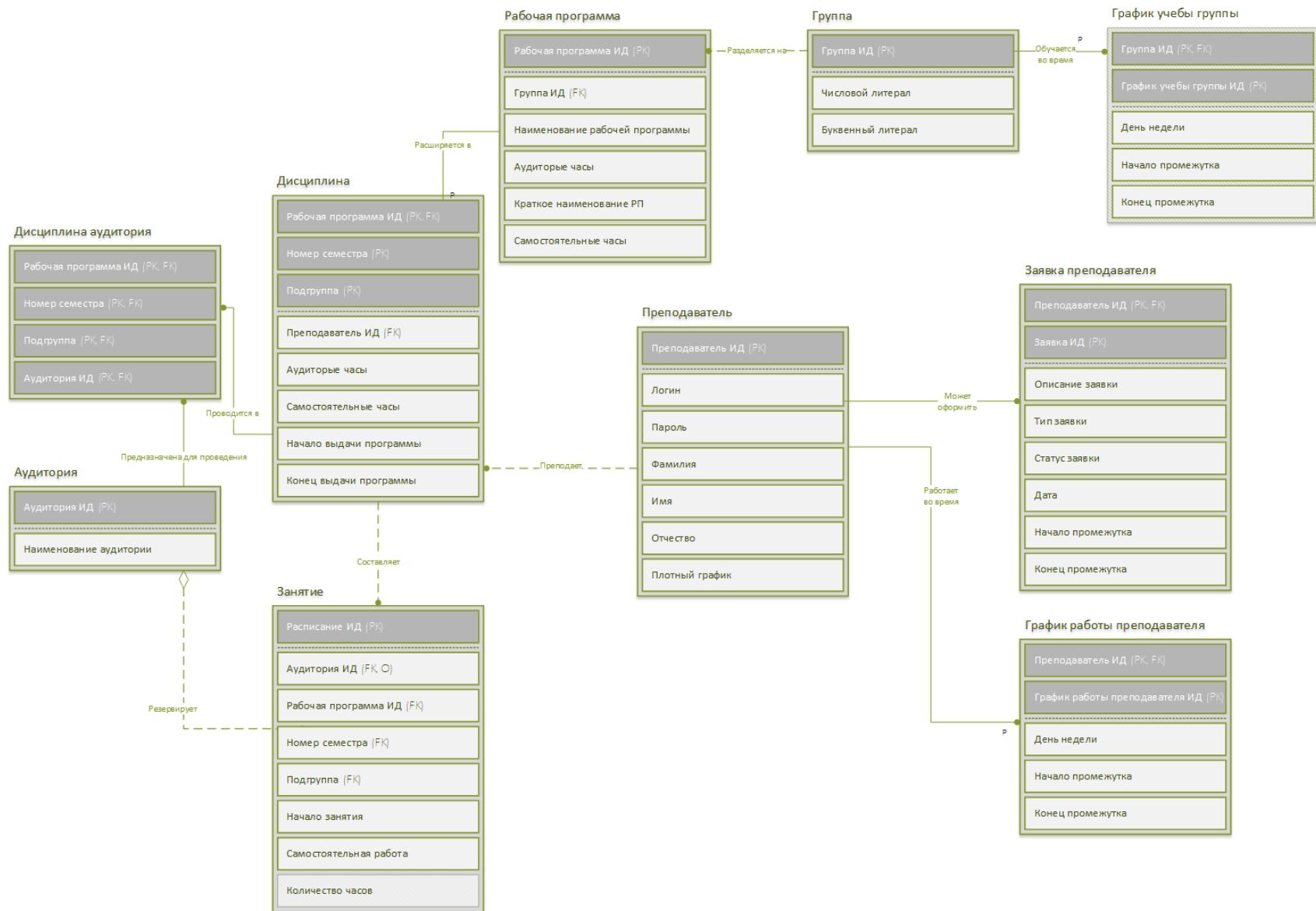


Рисунок 16. Модель базы данных FA уровня по методологии IDEF1X.

### 4.1.3. Составление глоссариев и спецификаций.

На основе изученной предметной области уже можно сформировать все сущности:

**ПРЕПОДАВАТЕЛЬ** – сущность характеризует реальных преподавателей, которые должны вести занятия.

Преподаватели могут иметь уникальный график работы в учебном учреждении, соответственно выявим сущность **РАБОЧЕЕ ВРЕМЯ ПРЕПОДАВАТЕЛЯ**. Преподаватели также имеют право, в случаях форс-мажора (например: вызвали в полицию), оформить заявку на корректировку своего графика работы, потому потребуется сущность **ЗАЯВКА ПРЕПОДАВАТЕЛЯ** для хранения таких заявок.

**ГРУППА** – эта сущность описывает учебные группы, группа ввиду особых обстоятельств может также иметь особый график обучения, такой график обучения будет отражаться в сущности **РАБОЧЕЕ ВРЕМЯ ГРУППЫ**.

Для составления расписания нужно знать, по каким дисциплинам необходимо выставлять занятия. Дисциплины и все виды практики отразим в сущности **РАБОЧАЯ ПРОГРАММА**.

**КЛАСС** – эта сущность описывает все кабинеты в учебном заведении.

Выдача академических часов по дисциплине может назначаться другой группе или другой подгруппе. Каждой группе и подгруппе необходимо закрепить одного преподавателя, отразим это в сущности **ДИСЦИПЛИНА**.

Для составления отчета «график выполнения учебных программ» необходимо учитывать каждую выдачу занятия некоторой дисциплины некоторой группе (подгруппе), отразим такие записи в сущности **ЗАНЯТИЕ**.

Согласно ранее описанной предметной области, я составил описание каждой сущности, а далее и каждого атрибута в программе Erwin data

modeler, позже используя функционал программы, я сгенерировал глоссарий сущностей и глоссарий атрибутов.

Таблица 4.1. Глоссарий сущностей.

<b>Имя сущности</b>	<b>Описание сущности</b>
ГРУППА	Группа студентов, которые должны проходить обучение по рабочим программам
ЗАЯВКА ПРЕПОДАВАТЕЛЯ	Заявка преподавателя, которая может временно скорректировать рабочее время преподавателя, что может оказать влияние на составленное расписание
АУДИТОРИЯ	Кабинет, в котором можно проводить занятия
ДИСЦИПЛИНА АУДИТОРИЯ	Таблица соответствий, в которой отражается в каких аудиториях можно проводить занятия некоторой дисциплины
ПРЕПОДАВАТЕЛЬ	Ведет занятия в пределах своего графика работы, может оформлять заявки для корректировки расписания
РАБОЧАЯ ПРОГРАММА	Рабочая программа, назначаемая группам\подгруппам, определяет количество академических и самостоятельных часов к выдаче, а также в какой период необходимо выдавать часы
ДИСЦИПЛИНА	Расширяет рабочую программу, определяет группу\подгруппу, их преподавателя на некоторый семестр и число академических и самостоятельных часов, которые необходимо выдать за семестр.
ГРАФИК РАБОТЫ ПРЕПОДАВАТЕЛЯ	Промежутки времени, в которых преподаватель может вести занятия
ГРАФИК УЧЁБЫ ГРУППЫ	Промежутки времени, в которых можно назначать занятия группе
ЗАНЯТИЕ	Записи выданных академических и самостоятельных часов

Таблица 4.2. Глоссарий атрибутов.

<b>Имя сущности</b>	<b>Имя атрибута</b>	<b>Описание атрибута</b>
ГРУППА	Группа ИД	Идентификатор группы
	Числовой литерал	Численное значение группы, которое в будущем можно увеличивать (переход на следующий курс)
	Буквенный литерал	Неизменный буквенный литерал группы
РАБОЧАЯ ПРОГРАММА	Рабочая программа ИД	Идентификатор рабочей программы
	Группа ИД	Идентификатор группы
	Наименование рабочей программы	Наименование учебной дисциплины
	Аудиторные часы	Количество академических часов, которые необходимо выдать некоторой группе с участием преподавателя



	Краткое наименование РП	Условное обозначение учебной дисциплины, которое будет отображаться в расписании
	Самостоятельные часы	Количество самостоятельных часов, которые необходимо выдать некоторой группе без участия преподавателя
АУДИТОРИЯ	Аудитория ИД	Идентификатор аудитории
	Наименование аудитории	Наименование аудитории
ДИСЦИПЛИНА АУДИТОРИЯ	Рабочая программа ИД	Идентификатор рабочей программы
	Номер семестра	Характеризует порядковый номер семестра, в пределах которого реализуется данная подпрограмма
	Подгруппа	Идентификатор подгруппы, где 0 – это вся группа, а N – это N-ая подгруппа
	Аудитория ИД	Идентификатор аудитории
ПРЕПОДАВАТЕЛЬ	Преподаватель ИД	Идентификатор преподавателя
	Логин	Кодовая строка для идентификации преподавателя
	Пароль	Парольная фраза преподавателя для подтверждения входа в систему
	Фамилия	Фамилия преподавателя
	Имя	Имя преподавателя
	Отчество	Отчество преподавателя
	Плотный график	Свойство, определяющее характер составления расписания для преподавателя. 1 – Назначать плотное расписание без «окон», 0 – допускаются «окна» в расписании
ЗАЯВКА ПРЕПОДАВАТЕЛЯ	Преподаватель ИД	Идентификатор преподавателя
	Заявка ИД	Идентификатор заявки преподавателя о корректировке рабочего расписания
	Описание заявки	Содержит объяснение и обстоятельства для удовлетворения запроса
	Тип заявки	Отражает характер запроса, где 0 – это запрос на отсутствие, 1 – присутствие преподавателя в некоторый промежуток времени
	Статус заявки	Отражает состояние запроса, где 0 – отклонено, 1 – принято, Null – решение еще не принято
	Дата	Отражает, на какой конкретно день должна влиять эта заявка

	Начало промежутка	Начало промежутка времени, на который ориентирована данная заявка
	Конец промежутка	Конец промежутка времени, на который ориентирована данная заявка
ДИСЦИПЛИНА	Рабочая программа ИД	Идентификатор рабочей программы
	Номер семестра	Характеризует порядковый номер семестра, в пределах которого реализуется данная дисциплина
	Подгруппа	Идентификатор подгруппы, где 0 – это вся группа, а N – это N-ая подгруппа
	Преподаватель ИД	Идентификатор преподавателя
	Аудиторные часы	Количество академических часов, которые необходимо выдать некоторой группе с участием преподавателя
	Самостоятельные часы	Количество самостоятельных часов, которые необходимо выдать некоторой группе без участия преподавателя
	Начало выдачи программы	Отражает дату начала выдачи данной подпрограммы, начиная с этой даты необходимо выдать все часы данной группе\подгруппы
	Конец выдачи программы	Отражает дату конца выдачи данной подпрограммы, до этой даты необходимо выдать все часы данной группе\подгруппы
ГРАФИК РАБОТЫ ПРЕПОДАВАТЕЛЯ	Преподаватель ИД	Идентификатор преподавателя
	График работы преподавателя ИД	Идентификатор отрезка времени, когда преподаватель может вести занятия
	День недели	Обозначает день недели, для которого назначается временной промежуток
	Начало промежутка	Начало временного промежутка, когда преподаватель может вести занятия
	Конец промежутка	Конец временного промежутка, когда преподаватель может вести занятия
ГРАФИК УЧЁБЫ ГРУППЫ	Группа ИД	Идентификатор группы
	График учебы группы ИД	Идентификатор отрезка времени, когда группе можно назначать занятия
	День недели	Обозначает день недели, для которого назначается временной промежуток
	Начало промежутка	Начало временного промежутка, когда группе можно назначать занятия
	Конец промежутка	Конец временного промежутка, когда группе можно назначать занятия

ЗАНЯТИЕ	Расписание ИД	Идентификатор занятия
	Аудитория ИД	Идентификатор аудитории
	Рабочая программа ИД	Идентификатор рабочей программы
	Номер семестра	Характеризует порядковый номер семестра, в пределах которого реализуется данная подпрограмма
	Подгруппа	Идентификатор подгруппы, где 0 – это вся группа, а N – это N подгруппа
	Начало занятия	Дата и время начала проведения данного занятия
	Самостоятельная работа	Свойство, характеризующая тип занятия, где 0 – это аудиторное занятие с участием преподавателя, а 1 – самостоятельная работа без участия преподавателя
	Количество часов	Обозначает количество реализуемых часов в данном занятии

#### 4.1.4. Создание физической модели БД

Базы данных – это хранилища, в которых содержатся большие массивы информации. Отправлять запросы, получать информацию из них можно при помощи СУБД – систем управления. Популярная СУБД, разработанная компанией Microsoft, – SQL Server. Эта программа доступна в нескольких редакциях, каждая из которых имеет свою специфику. Альтернатива этой коммерческой СУБД – система MySQL с открытым программным кодом.

Для своего проекта я выбрал СУБД от компании Microsoft, т.к. она имеет множество плюсов в сравнении с другими СУБД:

- СУБД масштабируется, поэтому работать с ней можно на портативных ПК или мощной мультипроцессорной технике. Процессор может одновременно обрабатывать большой объем запросов;
- Размер страниц – до 8 кб, поэтому данные извлекаются быстро, подробную и сложную информацию хранить удобнее. Система позволяет обрабатывать транзакции в интерактивном режиме, есть динамическая блокировка;

- Рутинные административные задачи автоматизированы: это управление блокировками, памятью, редактора размеров файлов. У системы продуманы настройки, можно создать профили пользователей;
- Реализован поиск по фразам, тексту, словам, можно создавать ключевые индексы;
- В SQL Server есть репликации через интернет, предусмотрена синхронизация. Есть полноценный веб-ассистент для форматирования страниц;
- В систему интегрирован сервер интерактивного анализа для принятия решений, создания корпоративных отчетов. Есть службы преобразования информации;
- Запросы можно формулировать на английском языке, без программирования;
- СУБД поддерживает работу с другими продуктами Microsoft: Access, MS Excel.

Не смотря на все плюсы, данная СУБД имеет и недостатки:

- Зависимость от операционной среды: СУБД работает только с системой Windows;
- Высокая цена программы.

Первый недостаток для меня совершенно не страшен, т.к. сервер базы данных будет функционировать на сервере на базе ОС Windows. Вторым недостатком в виде цены я решу использованием бесплатной версии СУБД Express для малых баз данных.

Сформировал ранее модель FA-уровня по методологии IDEF1X в программе Microsoft Visio, мне нужно было сформировать из этой модели физическую базу данных в целевой БД. Microsoft Visio не имеет функцию прямого инженеринга (Forward Engineer), потому я открыл Microsoft SQL Server Management Studio, создал новую диаграмму для моделирования и

начал описывать свою базу данных на основе ранее смоделированной модели.

Перейду к более детальному описанию модели. Описание сущностей и атрибутов уже были представлены ранее, потому расскажу только о выбранных типах данных для атрибутов и их особенностях.

В сущности ПРЕПОДАВАТЕЛЬ атрибуты фамилия, имя и отчество являются строками, в которых будут храниться соответствующие данные, потому типом данных для них будут `nvarchar` длиной в 50 символов. Атрибут логин должен быть уникальным среди других логинов, потому к данному столбцу дополнительно укажу ограничение уникальности. Логин и пароль также являются строками, потому тип данных для них будет `nvarchar`. Строка логина будет длиной в 64 символа, а пароля - в 128 символов, для большей безопасности. Атрибут плотного графика является лишь переключателем, потому его тип данных это 1 бит. Атрибут «ПРЕПОДАВАТЕЛЬ ИД» является первичным ключом для сущности ПРЕПОДАВАТЕЛЬ, я указал его как целое число `int` со свойством счетчика.

В сущности ЗАЯВКА ПРЕПОДАВАТЕЛЯ атрибуты ПРЕПОДАВАТЕЛЬ ИД и ЗАЯВКА ПРЕПОДАВАТЕЛЯ ИД образуют составной первичный ключ, потому второй атрибут имеет целочисленный тип данных `int` со свойством счетчика. Атрибут описание заявки представляет собой развернутое описание заявки, в которой указываются обстоятельства и причины заявки на естественном языке. Следовательно, возможно потребуется хранить достаточно большой текст, потому укажу тип данных `nvarchar` с запасом длины текста в 512 символов. Атрибут даты служит для хранения даты некоторого дня, на который и составлена заявка, укажу тип данных `date`. Атрибуты начала и конца промежутка представляют собой промежуток времени в указанной ранее дате, потому их тип данных будет `time`. Атрибут тип заявки указывает лишь на отсутствие или

присутствие преподавателя в ранее указанный промежуток времени в колледже, потому укажу тип данных как 1 бит, где 0 – это будет отсутствием преподавателя, 1 – присутствием. Атрибут статус заявки должен отражать состояние заявки. Для заявки определено 3 состояния: рассматривается, принята и отклонена, потому укажу тип данных как однобитовое поле с возможным хранением null значения, отсюда следует что null будет означать о состоянии рассмотрения, 1 – принятии, 0 – отклонении.

В сущности РАБОЧЕЕ ВРЕМЯ ПРЕПОДАВАТЕЛЯ атрибуты ПРЕПОДАВАТЕЛЬ ИД и РАБОЧЕЕ ВРЕМЯ ПРЕПОДАВАТЕЛЯ ИД образуют собой составной ключ, потому для второго атрибута укажу целочисленный тип данных int со свойством счетчика. Атрибуты день недели, начало и конец промежутка аналогичны одноименным атрибутам из сущности ЗАЯВКА ПРЕПОДАВАТЕЛЯ, потому типы данных совпадают.

В сущности ГРУППА атрибут ГРУППА ИД определяет первичный ключ, потому укажу тип данных как целочисленное число int со свойством счетчика. Атрибуты численный литерал и буквенный литерал определяют наименование группы, причем численный литерал — это число, потому тип данных – int, буквенный литерал – строка nvarchar с избыточной длиной в 16 символов.

Типы данных атрибутов в сущности ГРАФИК УЧЕБЫ ГРУППЫ аналогичны атрибутам из сущности РАБОЧЕЕ ВРЕМЯ ПРЕПОДАВАТЕЛЯ

В сущности РАБОЧАЯ ПРОГРАММА атрибут РАБОЧАЯ ПРОГРАММА ИД составляет первичный ключ, потому его тип данных – int со свойством счётчика. Атрибуты аудиторные и самостоятельные часы отражают количество часов к выдаче, т.е. отражают целое число, следовательно тип данных int. Атрибут наименование программы это строка, потому тип данных nvarchar с избыточной длиной в 128 символов.

В сущности ДИСЦИПЛИНА атрибуты РАБОЧАЯ ПРОГРАММА ИД, Подгруппа и Номер семестра являются первичным ключем, потому их типы данных int, счётчиков нет. Поля Аудиторные часы и Самостоятельные часы отражают количество академических часов, потому тип данных int. Атрибуты начала и конца выдачи программы отражают даты, они образуют промежуток времени, когда необходимо выдать дисциплину, потому тип данных date. Поле Преподаватель ИД является внешним ключем сущности ПРЕПОДАВАТЕЛЬ.

В сущности АУДИТОРИЯ атрибут АУДИТОРИЯ ИД определяет первичный ключ, потому тип данных для него int со счётчиком. Атрибут наименование аудитории – это строка, следовательно тип данных nvarchar с длиной в 64 символа.

Сущность ДИСЦИПЛИНА АУДИТОРИЯ является ассоциативной, т.е. все атрибуты является внешними ключами, они образуют составной первичный ключ.

В сущности РАСПИСАНИЕ атрибут РАСПИСАНИЕ ИД определяет первичный ключ, потому его тип данных – int со счётчиком. Атрибуты РАБОЧАЯ ПРОГРАММА, ГРУППА ИД, ПОДГРУППА и АУДИТОРИЯ ИД являются внешними ключами, причем АУДИТОРИЯ ИД может принимать значение null. Атрибут начало занятия отражает момент времени, когда начнется занятие, т.е. необходимо знать когда и во сколько начнется занятие, потому тип данных для этого атрибута – datetime. Атрибут самостоятельная работа отражает аудиторное или самостоятельное занятие, потому достаточно 1 бита. Поле Количество часов отражает количество реализуемых часов в данном занятии, потому тип данных int.

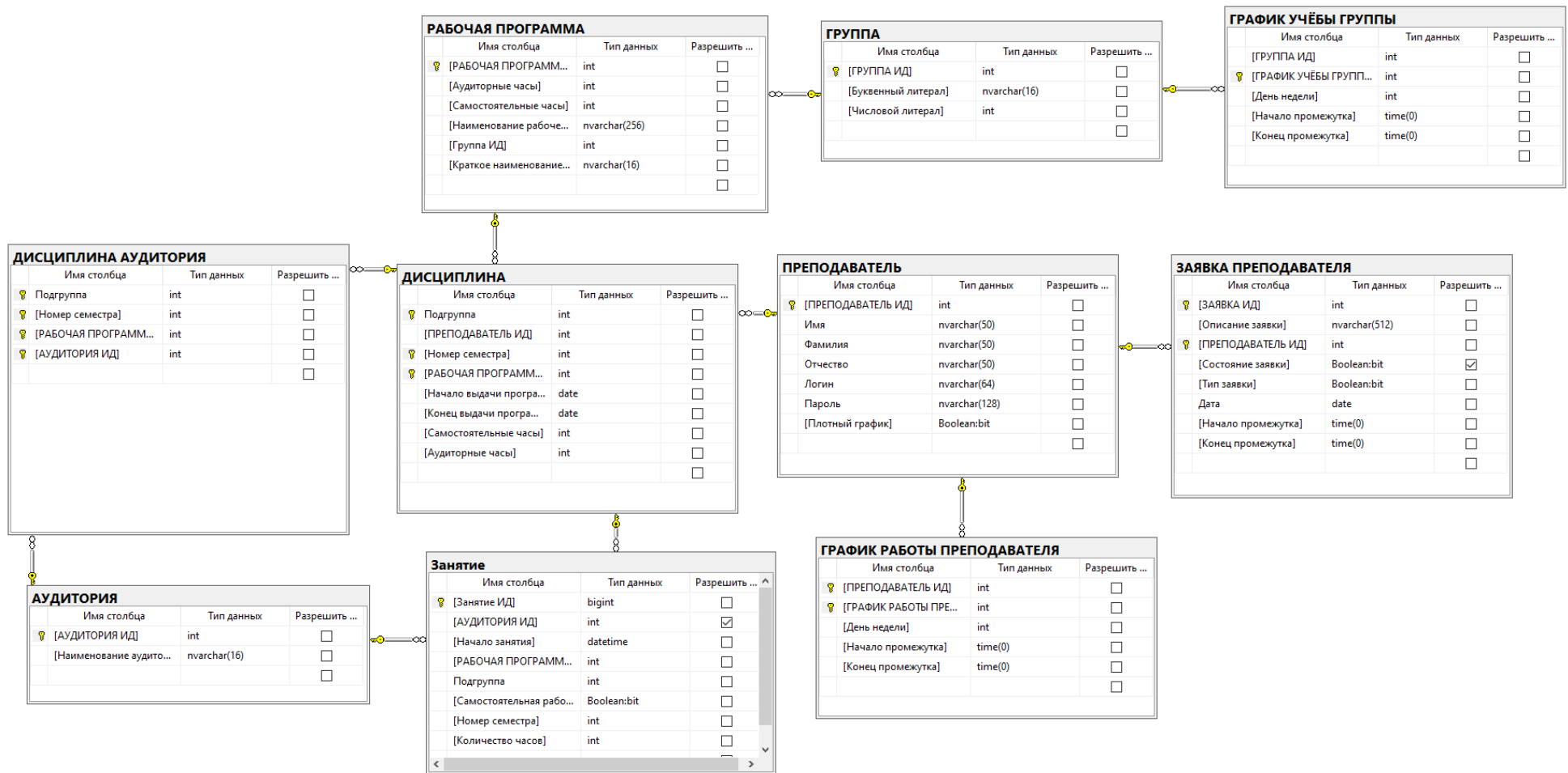


Рисунок 17. Физическая модель базы данных в СУБД MySQL.



## **4.2. Проектирование и реализация приложений конечных пользователей.**

### **4.2.1. Выбор среды разработки приложения для ИС**

Для выбора среды разработки приложения для информационной системы, потребуется провести анализ существующих решений.

#### **1. Visual Studio**

Описание: один из старейших программных продуктов для создания как консольных приложений, так и обладающие графическим интерфейсом. Добавление сторонних плагинов позволяет серьёзно расширить функциональность среды, в том числе до кроссплатформенного состояния.

Недостатки: новичку будет просто невозможно самостоятельно разобраться с Visual Studio без прохождения специальных курсов и чтения литературы. Это продукт скорее для опытных разработчиков, обращающих внимание на качество редактора и функции тестирования.

#### **2. Netbeans**

Описание: мощная IDE для разработки приложений на Java, JavaScript, Python, PHP, C, C++ и даже Ада.

Недостатки: невысокое быстродействие из-за концепции «всё в одном». Некоторые плагины (в том числе для разработки приложений для Android) имеют существенные ограничения функциональности.

#### **3. IntelliJ IDEA**

Описание: IDE, разработанная компанией JetBrains, позволяющая создавать программы на множестве популярных языков, среди которых Java, JavaScript, Python, Ruby, Groovy, Scala, PHP, C, C++.

Недостатки: производительность. Томительное ожидание выполнения компиляции, перекомпиляции, тестирования порой действительно раздражает.

В качестве среды разработки приложения я выберу Visual Studio, ввиду поддержки среды .NET (а в ней технологию ADO.NET) и языка программирования C#. Именно эти технологии я собираюсь использовать при разработке приложения. Помимо этого, Visual Studio обеспечивает необходимый функционал для отладки и тестирования приложения.

#### **4.2.2. Разработка интерфейса прототипа приложения**

Весь интерфейс приложения разделен на 8 разделов (Рис. 18). Разделы «Преподаватель» и «Группа» максимально автономны, они не зависят от других разделах, в них можно управлять одноименными информационными объектами. Раздел «Рабочие программы» сильно зависит от существующих групп, т.к. любая рабочая программа прикрепляется к какой-либо группе. В этом разделе также можно управлять дисциплинами. А так как дисциплину обязательно ведет какой-либо преподаватель, то возникает зависимость от преподавателей. Тут же можно выбрать в каких аудиториях допускается проведение занятий некоторой дисциплины, это порождает зависимость от аудиторий, но в разделе «Аудитория» также можно выбрать какие дисциплины допускаются проводить в некоторой аудитории. Раздел «Расписание» является основным в приложении. В нем можно управлять занятиями, т.е. добавлять занятия, изменять и удалять. Для редактирования расписания нужно соблюдать все ограничения, которые описаны в других разделах приложения. Раздел «Справка» является просто информационным справочником. Раздел с заявками преподавателей будет реализован в дипломной работе.



Рисунок 18. Главное меню приложения.

#### 4.2.2.1. Расписание

Данный раздел является основным в приложении, он предназначен для управления такими информационными объектами как занятия. В верхней части окна можно выбрать некоторую дату (Рис. 19), по умолчанию указывается текущая дата. При выборе даты приложение считывает из базы данных все занятия, которые находятся в пределах недели, выбранной пользователем, и выстраивает занятия в табличном виде, группируя их по дням и группам (Рис. 20).

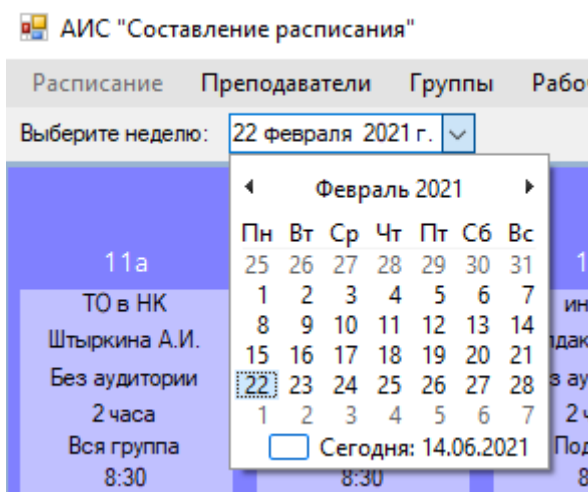


Рисунок 19. Демонстрация выбора недели, через выбор некоторой даты.

АИС "Составление расписания"

Расписание Преподаватели Группы Рабочие программы Аудитории Справка Новых заявок нет

Выберите неделю: 23 февраля 2021 г. Экспорт

**Понедельник (22.02.2021)**

11а	1а	1до	1кп	1п	21а	2а	2до	2и	2кп	3а	3до	3кп	3п	4а	4до		
ТО в НК Штыркина А.И. Без аудитории 2 часа Вся группа 8:30	Литература Красноперова Л.М. Без аудитории 2 часа Вся группа 8:30	ин. яз. Булдакова Л.А. Без аудитории 2 часа 1 Подгруппа 8:30	история Михайлова А.А. Без аудитории 2 часа Вся группа 8:30	естеств. (а) Павлова О.Л. Без аудитории 2 часа Вся группа 8:30	РЯ с МП Касьянова А.В. Без аудитории 2 часа Вся группа 8:30	педагогика Абрамова Е.А. Без аудитории 2 часа Вся группа 8:30	ТиМОПВД Болотова Е.Ю. Без аудитории 2 часа Вся группа 8:30	психология Квашина А.С. Без аудитории 2 часа Вся группа 8:30	психология Якимова А.В. Без аудитории 2 часа Вся группа 8:30	ТОМ Петрова А.А. Без аудитории 2 часа Вся группа 8:30	ППОО Якимова А.В. Без аудитории 2 часа Вся группа 10:10	ТиМО окр. м. Усольцева А.Д. Без аудитории 2 часа Вся группа 8:30	МП Мясников С.В. Без аудитории 2 часа Вся группа 8:30	ПП 05.01 Шаркова Е.В. Без аудитории 2 часа Вся группа 8:30	ПКИЯ Пробст С.В. Без аудитории 2 часа Вся группа 8:30		
Основ. философ. Михайлова А.А. Без аудитории 2 часа Вся группа 10:10	ОБЖ Цыганкова С.В. Без аудитории 2 часа Вся группа 10:10	ин. яз. Терещенко Л.Л. Без аудитории 2 часа 2 Подгруппа 8:30	рус. яз. Касьянова А.В. Без аудитории 2 часа Вся группа 10:10	инф. Черносутов А.С. Без аудитории 2 часа 1 Подгруппа 10:10	Естеств. с МП Усольцева А.Д. Без аудитории 2 часа Вся группа 10:10	ТО в НК Штыркина А.И. Без аудитории 2 часа Вся группа 10:10	психология Квашина А.С. Без аудитории 2 часа Вся группа 10:10	ин. яз. Булдакова Л.А. Без аудитории 2 часа 1 Подгруппа 10:10	педагогика Абрамова Е.А. Без аудитории 2 часа Вся группа 10:10	ТОМ Петрова А.А. Без аудитории 2 часа Вся группа 10:10	ППОО Якимова А.В. Без аудитории 2 часа Вся группа 12:40	РЯ с МП Попова У.А. Без аудитории 2 часа Вся группа 10:10	МП Мясников С.В. Без аудитории 2 часа Вся группа 10:10	ПП 05.01 Шаркова Е.В. Без аудитории 2 часа Вся группа 10:10	Тех. тв. Порсина А.В. Без аудитории 2 часа Вся группа 8:30		
ПКИЯ Булдакова Л.А. Без аудитории 2 часа Вся группа 12:40	Русский язык Шипицина Г.А. Без аудитории 2 часа Вся группа 12:40	русс. яз. Шипицина Г.А. Без аудитории 2 часа Вся группа 10:10	литер. Красноперова Л.М. Без аудитории 2 часа Вся группа 12:40	компл. гр. Павлова О.Л. Без аудитории 2 часа 1 Подгруппа 10:10	ТОМ Петрова А.А. Без аудитории 2 часа Вся группа 12:40	БЖ Цыганкова С.В. Без аудитории 2 часа Вся группа 12:40	ТОО в рег Штыркина А.И. Без аудитории 2 часа Вся группа 12:40	ин. яз. Терещенко Л.Л. Без аудитории 2 часа 2 Подгруппа 10:10	ТО КРО Смирнова К.И. Без аудитории 2 часа Вся группа 12:40	Ест. с МП Усольцева А.Д. Без аудитории 2 часа Вся группа 12:40	ТиММР Шаркова Е.В. Без аудитории 2 часа Вся группа 14:20	РЯ с МП Попова У.А. Без аудитории 2 часа Вся группа 12:40	МП Мясников С.В. Без аудитории 2 часа Вся группа 12:40	ПП 05.01 Шаркова Е.В. Без аудитории 2 часа Вся группа 12:40	ПКИЯ Пробст С.В. Без аудитории 2 часа Вся группа 10:10		
спорт-озд. Касьянова А.В. Без аудитории 2 часа Вся группа 12:40		Введ. в спец. Квашина А.С. Без аудитории 2 часа Вся группа 12:40		инф. Черносутов А.С. Без аудитории 2 часа 2 Подгруппа 12:40				педагогика Абрамова Е.А. Без аудитории 2 часа Вся группа 12:40							Тех. тв. Порсина А.В. Без аудитории 2 часа Вся группа 10:10	ПКИЯ Пробст С.В. Без аудитории 2 часа Вся группа 12:40	Тех. тв. Порсина А.В. Без аудитории 2 часа Вся группа 12:40

**Вторник (23.02.2021)**

11а	1а	1до	1кп	1п	21а	2а	2до	2и	2кп	3а	3до	3кп	3п	4а	4до

**Среда (24.02.2021)**

11а	1а	1до	1кп	1п	21а	2а	2до	2и	2кп	3а	3до	3кп	3п	4а	4до
ТОМ	История	литер.	матем.	ин. яз.	ТиМ муз.	СВС	физ-ра	педагогика	педагогика	Рук. ПИД	ПП 02.01	ТиМО окр. м.	МП	ПП 05.01	ПКИЯ

Рисунок 20. Отображение расписания в приложении.

Выбрав некоторую неделю, ее можно экспортировать в файл Excel, PDF или распечатать. Для этого необходимо нажать на кнопку «Экспорт» (Рис. 21), которая находится над таблицей расписания и прижата к правому краю окна. По нажатию этой кнопки откроется новое окно, в котором можно просмотреть расписание в более привычной форме (Рис. 22) и сохранить его в файле требуемого формата или распечатать (Рис. 23).

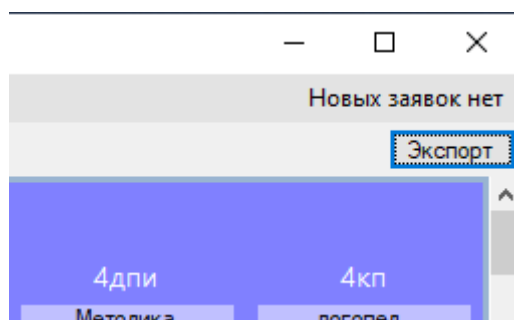


Рисунок 21. Кнопка экспорта расписания.

Экспорт расписания																			
1 из 2 ?																			
Найти   Следующий																			
Дата	Время	11а	1а	1до	1кп	1п	21а	2а	2до	2п	2кп	3а	3до	3кп	3п	4а	4до	4дп	4кп
25.02.2021	08:30:00	УП 03.02 Афоняева Е.С. 2 часа 41	Математика Прожерина Н.И. 2 часа 14	естеств. (6) Усольцева А.Д. 2 часа 37	литер. Красноперова Л.М. 2 часа 55	история Михайлова А.А. 2 часа 60	ПКИЯ Булдакова Л.А. 2 часа 39 Тех. тв. Порфина А.В. 2 часа 23	ТОМ Петрова А.А. 2 часа 26	философ. Ленская Е.В. 2 часа 22	ОВОП Мохирев К.В. 1 подгруппа 2 часа 53	ОАП Квашина А.С. 2 часа 11	ОПИД Афоняева Е.С. 2 часа 7	ПП 02.01 Штыркина А.И. 2 часа Без аудиторий	Тим муз. Насонова И.А. 2 часа 56	ПитПМ Черноскотов А.С. 2 часа 43	ПП 05.01 Шаркова Е.В. 2 часа 66	Т и ПаМР Устьянцева И.Ю. 2 часа 9	ПП.03.01 Мутли А.Г. 2 часа 13	ин. яз. Терещенко Л.Л. 1 подгруппа 2 часа 27 ин. яз. Пробст С.В. 2 подгруппа 2 часа 26
	10:10:00	УП 03.02 Афоняева Е.С. 2 часа 41	Математика Прожерина Н.И. 2 часа 14	естеств. (6) Усольцева А.Д. 2 часа 37	литер. Красноперова Л.М. 2 часа 55	история Михайлова А.А. 2 часа 60	ПКИЯ Булдакова Л.А. 2 часа 39 Естеств. с МП Усольцева А.Д. 2 часа 23	ТОМ Петрова А.А. 2 часа 26	философ. Ленская Е.В. 2 часа 22	психология Квашина А.С. 2 часа 44	физ-ра Петкин И.Н. 2 часа Без аудиторий	ОПИД Афоняева Е.С. 2 часа 7	ПП 02.01 Штыркина А.И. 2 часа Без аудиторий	Ин. яз. Пробст С.В. 1 подгруппа 2 часа 23 Ин. яз. Терещенко Л.Л. 2 подгруппа 2 часа 56	ПитПМ Черноскотов А.С. 2 часа 43	ПП 05.01 Шаркова Е.В. 2 часа 66	Т и ПаМР Устьянцева И.Ю. 2 часа 9	ПП.03.01 Мутли А.Г. 2 часа 13	ИОС Лотова Г.П. 1 подгруппа 2 часа 25 Т и М Муз. Насонова И.А. 2 подгруппа 2 часа 27
	12:40:00	УП 03.02 Афоняева Е.С. 2 часа 41	Обществознание Ленская Е.В. 2 часа 14	история Михайлова А.А. 2 часа 37	естеств. Усольцева А.Д. 2 часа 55	литер. Красноперова Л.М. 2 часа 60	Ин. яз. Пробст С.В. 1 подгруппа 2 часа 23 Ин. яз. Булдакова Л.А. 2 подгруппа 2 часа 39	МРР Вилом М.Н. 2 часа 26	БЖ Цыганкова С.В. 2 часа 22	психология Квашина А.С. 2 часа 44	РЯ с МП Попова У.А. 2 часа 11	ТОМ Петрова А.А. 2 часа 7	ПП 02.01 Штыркина А.И. 2 часа Без аудиторий	Физ-ра Прожерина Н.И. 2 часа Без аудиторий	Ин. яз. Терещенко Л.Л. 2 часа 43	ПП 05.01 Шаркова Е.В. 2 часа 66	инф. Лотова Г.П. 1 подгруппа 2 часа 28 инф. Порфина А.В. 2 подгруппа 2 часа 56	ПП.03.01 Мутли А.Г. 2 часа 13	Т и М Муз. Насонова И.А. 2 подгруппа 2 часа 27 ИОС Лотова Г.П. 1 подгруппа 2 часа 25
	14:20:00		Обществознание Ленская Е.В. 2 часа 14	история Михайлова А.А. 2 часа 37	Физика Петрова А.А. 2 часа 55	литер. Красноперова Л.М. 2 часа 60	СОТ Шаркова Е.В. 2 часа 23	БЖ Цыганкова С.В. 2 часа 26	композиция Мутли А.Г. 1 подгруппа 2 часа 52	РЯ с МП Попова У.А. 2 часа 11				Физ-ра Петкин И.Н. 2 часа Без аудиторий		инф. Порфина А.В. 2 подгруппа 2 часа 56 инф. Лотова Г.П. 1 подгруппа 2 часа 28		ИОС Лотова Г.П. 1 подгруппа 2 часа 25 Т и М Муз. Насонова И.А. 2 подгруппа 2 часа 27	
26.02.2021	08:30:00	ин. яз. Булдакова Л.А. 1 подгруппа 2 часа 39 ин. яз. Терещенко Л.Л. 2 подгруппа 2 часа 41		обществ. Ленская Е.В. 2 часа 37	инф. Павлова О.Л. 1 подгруппа 2 часа 28 инф. Лотова Г.П. 2 подгруппа 2 часа 13	инф. Черноскотов А.С. 1 подгруппа 2 часа 59	ОПИД Афоняева Е.С. 2 часа 23	детск. литер. Шипицина Г.А. 2 часа 26	психология Квашина А.С. 2 часа 22	ОВОП Мохирев К.В. 2 подгруппа 2 часа 53	ТимОДКР Штыркина А.И. 2 часа 11	физ-ра Прожерина Н.И. 2 часа Без аудиторий	Тим взм. с род. Вилом М.Н. 2 часа Без аудиторий	РЯ с МП Попова У.А. 2 часа 56	ПМ Мясников С.В. 2 часа 43	ПП 05.01 Шаркова Е.В. 2 часа 66	МРЯ Пробст С.В. 2 часа 9 Тех. тв. Порфина А.В. 2 часа 25	ПП.03.01 Мутли А.Г. 2 часа 13	Т и М Муз. Насонова И.А. 1 подгруппа 2 часа 27 ИОС Шаркова Е.В. 2 подгруппа 2 часа 28
	10:10:00	УП 03.02 Афоняева Е.С. 2 часа 41	История Михайлова А.А. 2 часа 14	обществ. Ленская Е.В. 2 часа 37	матем. Прожерина Н.И. 2 часа 55	матем. Павлова О.Л. 2 часа 60	ОПИД Афоняева Е.С. 2 часа 23	физ-ра Петкин И.Н. 2 часа Без аудиторий	психология Квашина А.С. 2 часа 22	Анатом. Усольцева А.Д. 2 часа 44	ТимОДКР Штыркина А.И. 2 часа 11	ин. яз. Булдакова Л.А. 1 подгруппа 2 часа 39 ин. яз. Терещенко Л.Л. 2 подгруппа 2 часа 7	Тим ЭО Черноскотов А.С. 2 часа Без аудиторий	РЯ с МП Попова У.А. 2 часа 56	ПМ Мясников С.В. 2 часа 43	ПП 05.01 Шаркова Е.В. 2 часа 66	Тех. тв. Порфина А.В. 2 часа 25 МРЯ Пробст С.В. 2 часа 9	ПП.03.01 Мутли А.Г. 2 часа 13	ИОС Шаркова Е.В. 2 подгруппа 2 часа 28 Т и М Муз. Насонова И.А. 1 подгруппа 2 часа 27
	12:40:00	УП 03.02	История	матем.	обществ.	матем.	Физ-ра	БЖ	Тим игрд.	Анатом.	спец. п.	Т и М ФВ	ТимРР	ООВР	ПМ	ПП 05.01	ин. яз.	ПП.03.01	Т и М Муз.

Рисунок 22. Экспорт расписания.

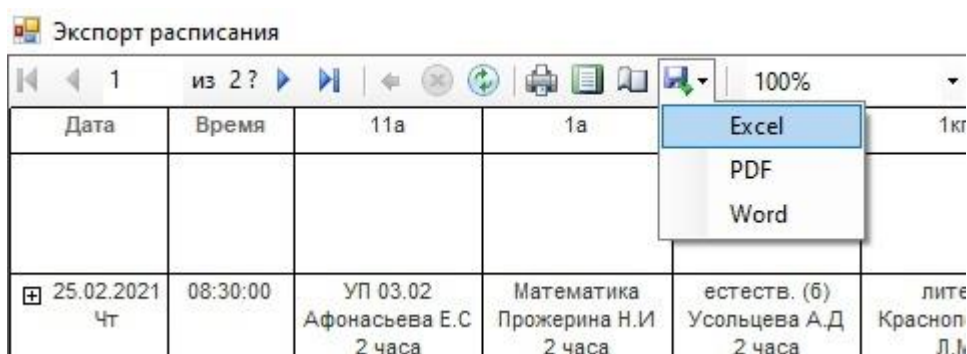


Рисунок 23. Сохранение расписания в файле требуемого формата.

Для добавления занятия необходимо сначала выбрать необходимую неделю. После того, как приложение отрисует расписание на выбранную неделю, необходимо найти требуемый день (отображается 7 дней выбранной недели, сверху всегда понедельник, следовательно снизу воскресенье). Найдя требуемый день, необходимо найти столбец нужной группы. После нужно вызвать контекстное меню у наименования группы и в контекстном меню нажать на пункт «Добавить занятие» (Рис. 24).

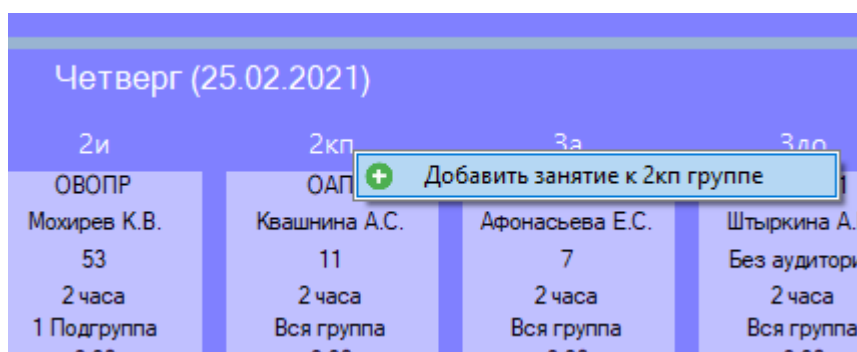


Рисунок 24. Функция добавления занятия.

Вызвав данную функцию, откроется диалоговое окно с редактором занятия (Рис. 25). В данном редакторе первым делом необходимо выбрать дату и время занятия, по умолчанию дата выбирается на основе дня, у которого вызвали контекстное меню, а время устанавливается на основе следующей логики:

1. Извлекаются занятия на выбранную дату для группы, у которой ранее вызывали контекстное меню;

2. После вычисляется самое позднее занятие: к началу занятия прибавляется количество часов, умноженное на длину академического часа;
3. К полученному времени добавляется длительность перемены и возвращается данный результат;
4. В случае отсутствия занятий, возвращается константа 8:30, что соответствует первой паре в колледже.

Установив дату и время занятия, нужно определить какое будет занятие: самостоятельная работа или с участием преподавателя? Это определяется элементом CheckBox с текстом «Самостоятельная работа», если данный элемент активен, значит занятие будет проводиться без участия преподавателя. В случае если данный элемент неактивен - то с участием преподавателя.

Далее необходимо определить количество часов, реализуемых в данном занятии. По умолчанию будет введено 2 часа, что соответствует 2 академическим часам в 1 паре.

После необходимо выбрать группу, для которой выставляется занятие, по умолчанию выбирается та группа, у которой ранее вызывали контекстное меню. Список групп опционально фильтруется по графику обучения, т.е. для каждой группы высчитывается возможность обучения в соответствии с графиком обучения. По умолчанию данная опция включена и управляется элементом CheckBox над списком групп.

Выбрав группу, в соседней списке отобразятся ее рабочие программы. Необходимо выбрать рабочую программу, которую мы хотим реализовать в занятии. Выбрав рабочую программу в последнем списке отобразится дисциплины выбранной рабочей программы. Список дисциплин также опционально фильтруется 3 функциями и 1 функцией обязательно. Единственная обязательная функция фильтрует дисциплины по вхождению



даты занятия в период дат реализации дисциплины. Опциональные функции фильтрации по умолчанию включены и управляются элементами CheckBox, расположенными над списком дисциплин. Опциональные функции фильтрации отражают:

1. Скрытие уже выданных дисциплин, т.е. если в дисциплине запланировано выдать N часов, и, сумма часов занятий данной дисциплины равно или больше числа N, то данную дисциплину следует исключить из списка;
2. Скрытие дисциплин, в которых преподаватель уже занят другим занятием.
3. Скрытие дисциплин, в которых преподаватель не может вести занятия ввиду несоответствия времени и длительности занятия его графику работы.

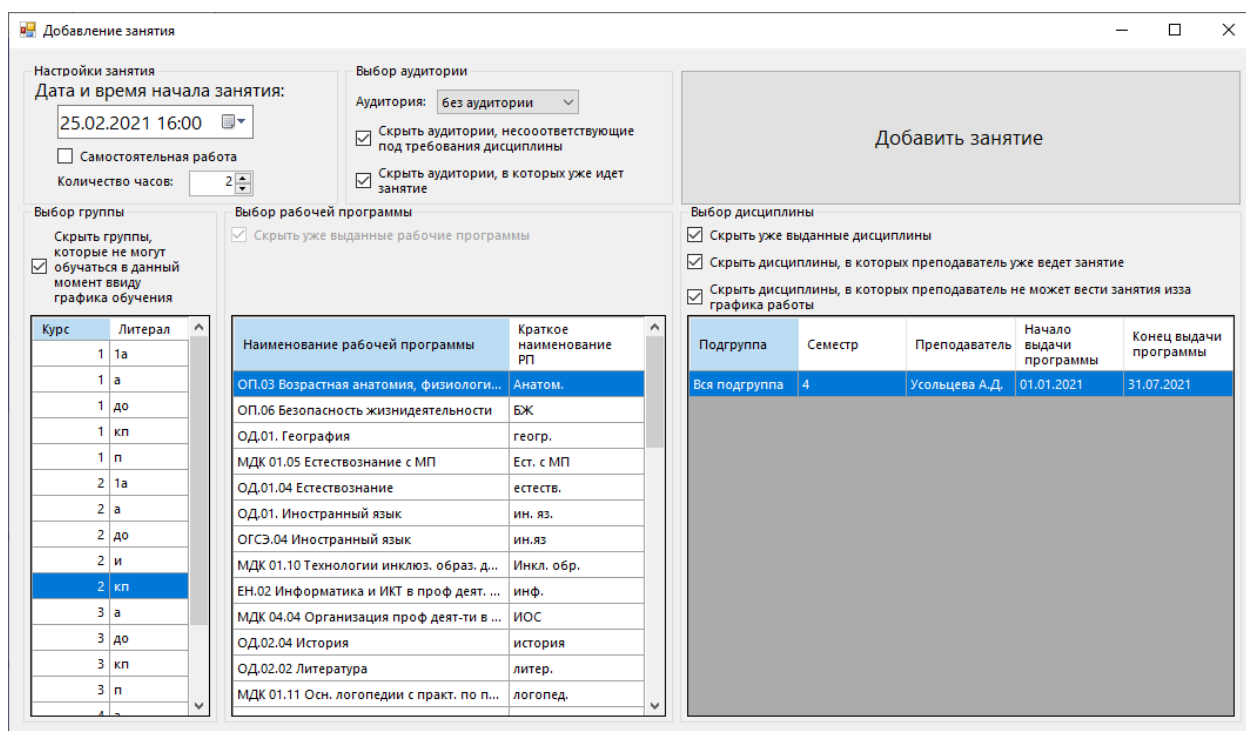


Рисунок 25. Редактор занятия

Определив дисциплину, приложение загрузит список аудиторий в элемент ComboBox соответствующих всем включенным опциям фильтрации для данной дисциплины. Необходимо определить аудиторию, в которой

будет проходить занятие или выбрать пункт «Без аудитории». Опциональные функции фильтрации также по умолчанию включены и управляются соответствующими элементами CheckBox.

Определив все данные для занятия, можно нажать на кнопку «Добавить занятие». При нажатии по этой кнопке сформируется занятие в базе данных, закроется окно редактора занятия и обновиться таблица расписания в основном окне приложения, где отобразиться добавленное занятие.

Для удаления некоторого занятия необходимо найти его в расписании, после нужно вызвать контекстное меню у требуемого занятия и кликнуть по пункту «Удалить занятие» (Рис. 26).

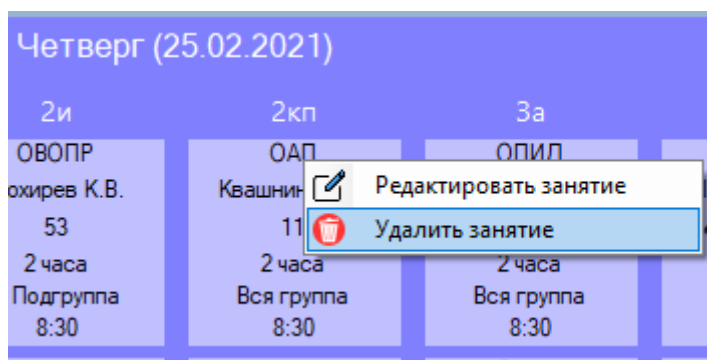


Рисунок 26. Функция удаления занятия.

Для редактирования занятия аналогично требуется вызвать контекстное меню у требуемого занятия и выбрать пункт «Редактировать занятие» (Рис. 27).

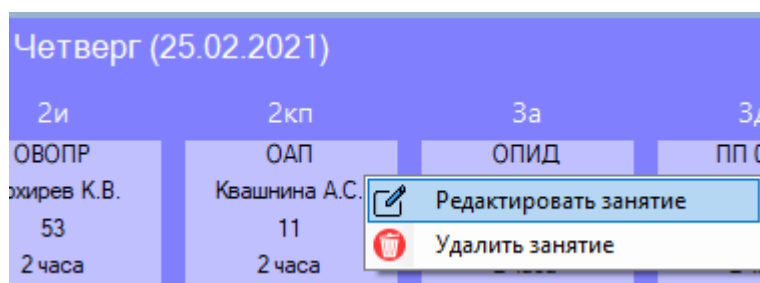


Рисунок 27. Функция редактирования занятия.

При вызове функции редактирования занятия откроется новое диалоговое окно редактора занятий, в котором все данные автоматически заполняются на основе данных из занятия, выбранного для редактирования (Рис. 28). Процесс редактирования должен происходить по аналогии с добавлением занятия, от даты и времени до аудитории.

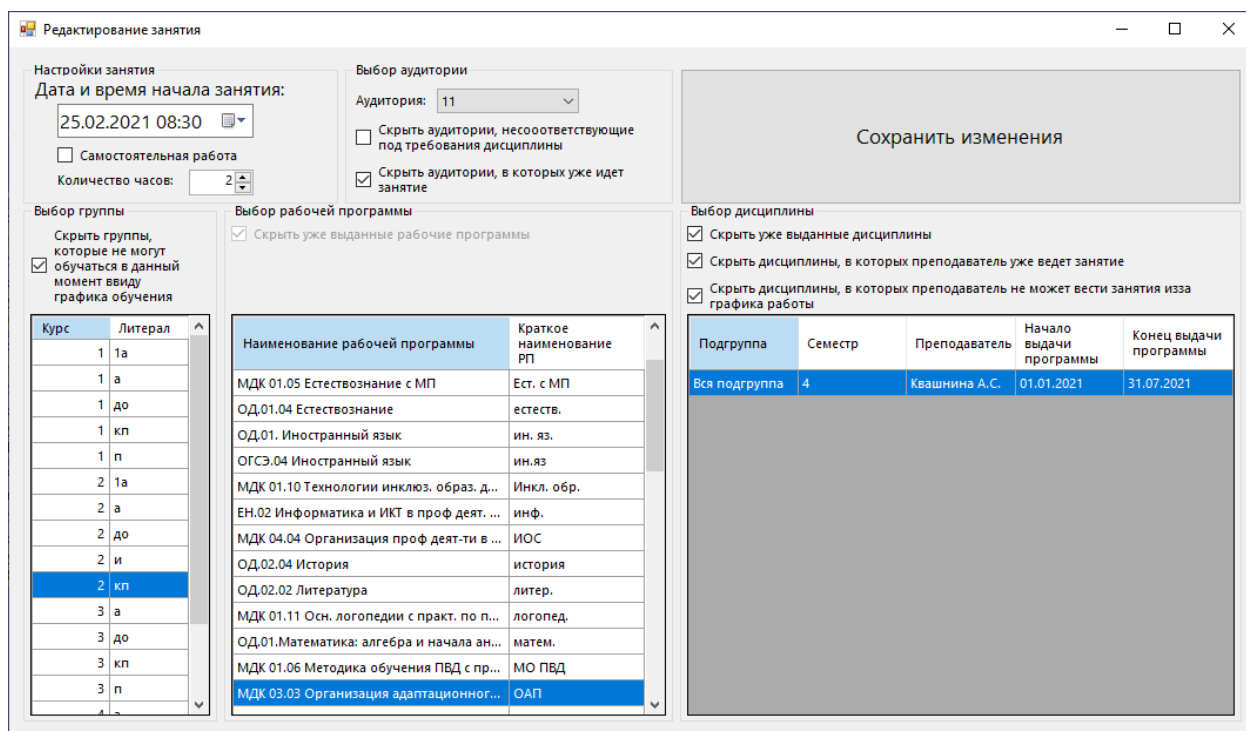


Рисунок 28. Редактирование существующего занятия.

#### 4.2.2.2. Преподаватели

Данный раздел является тематическим, т.е. предназначен для управления такими информационными объектами как преподаватель. Для этого в левой части окна представлен список преподавателей, отсортированных по фамилии (Рис. 29).

Кликнув по имени преподавателя, можно более подробно рассмотреть информацию о нем (Рис. 30).

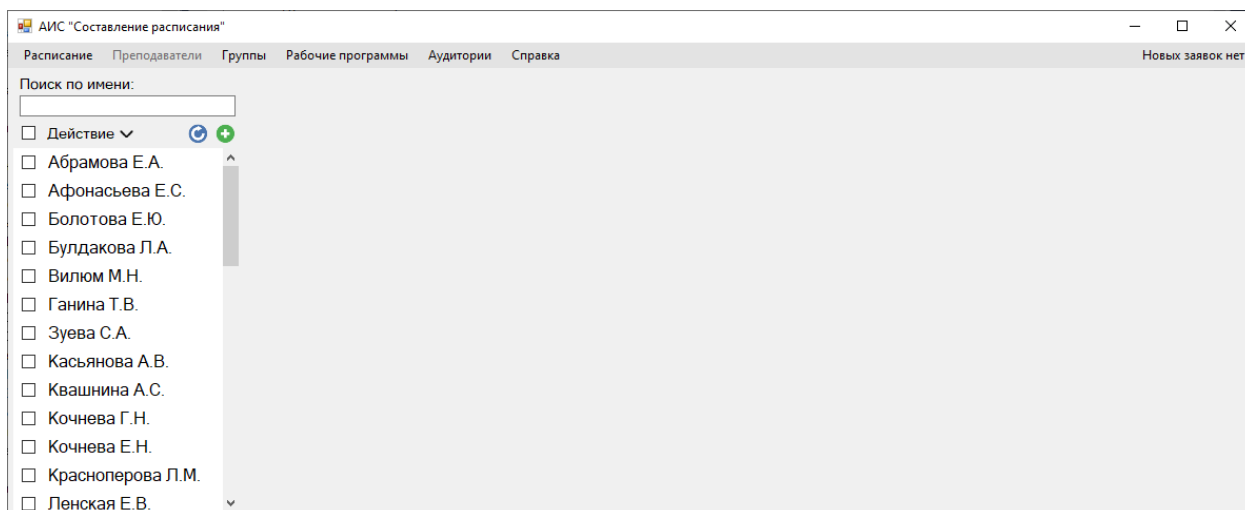


Рисунок 29. Интерфейс раздела преподавателей.

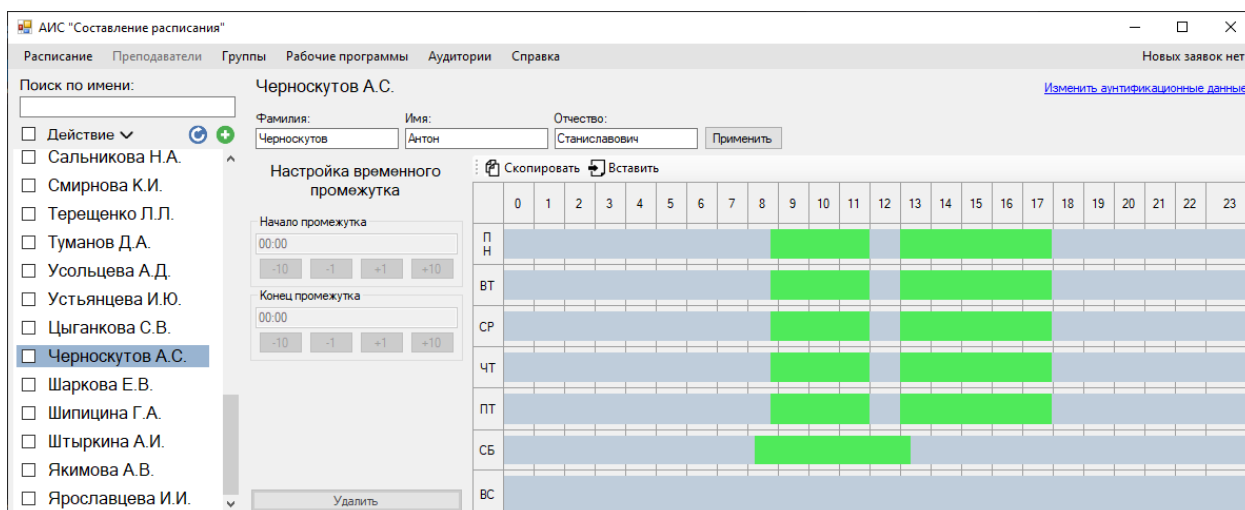


Рисунок 30. Интерфейс раздела преподавателей (с выбором преподавателя).

Для быстрого поиска преподавателя существует функция поиска. Для этого надо ввести в текстовое поле над списком хотя бы некоторую часть имени. Программа будет автоматически фильтровать список по мере ввода текста, совершая поиск подстроки в имени преподавателя (Рис. 31, регистр не учитывается, но требуется вводить подстроку без ошибок).

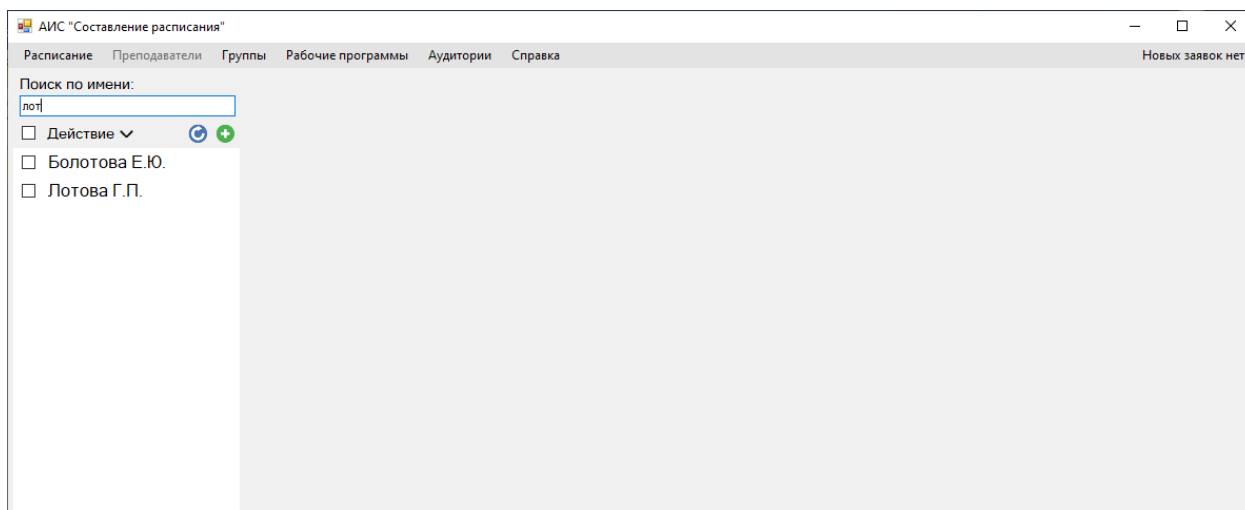


Рисунок 31. Демонстрация функции поиска преподавателя в списке.

Для создания нового преподавателя в базе данных существует кнопка-иконка в виде плюса в зеленом круге, она предназначена для создания нового преподавателя. При наведении курсора на кнопку пользователю будет представлена текстовая подсказка, описывающая, за что она отвечает (Рис. 32).

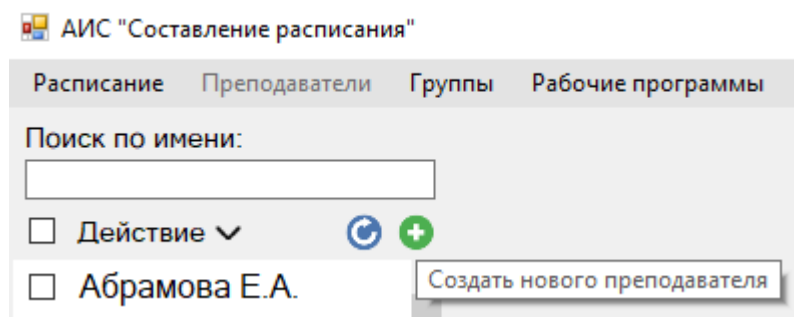


Рисунок 32. Текстовая подсказка кнопки создания преподавателя.

При нажатии по данной кнопке откроется новое диалоговое окно, в котором необходимо заполнить ряд обязательных полей (Рис. 33). После, на основе введенных данных, можно сформировать новую запись преподавателя в базе данных по нажатию по кнопке «Создать преподавателя». Также существует возможность отменить операцию создания преподавателя, для этого нужно закрыть диалоговое окно, щелкнув по крестику окна.

Рисунок 33. Форма создания преподавателей.

Для удаления преподавателя существует 2 способа:

- Удаление конкретного преподавателя – Для удаления потребуется навести курсор на имя преподавателя, которого необходимо удалить. Далее необходимо вызвать контекстное меню, щелкнув правой кнопкой мыши, находясь курсором на именем преподавателя. После остается лишь щелкнуть по пункту контекстного меню «Удалить» (Рис. 34). При вызове функции удаления программа запросит подтверждения действия (Рис. 35).

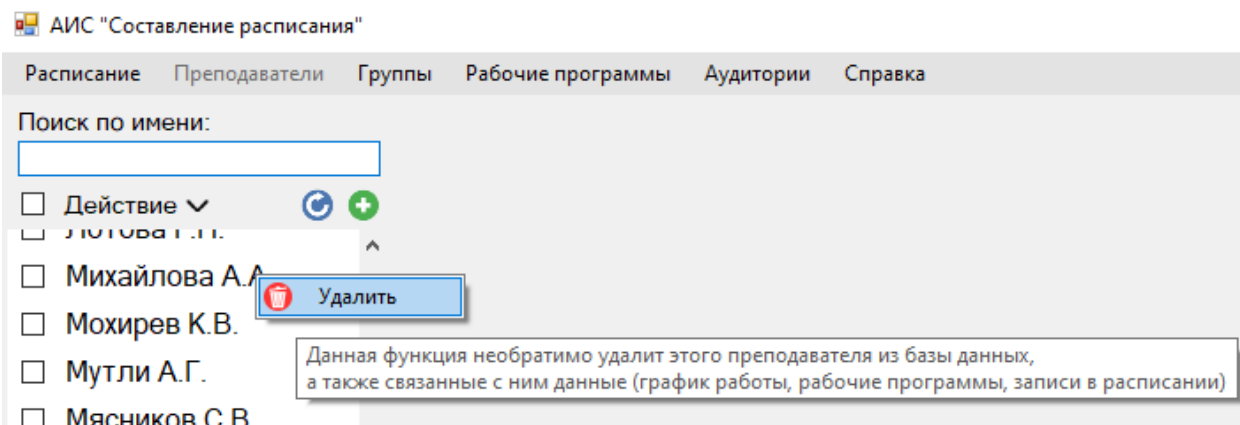


Рисунок 34. Демонстрация функции удаления конкретного преподавателя.

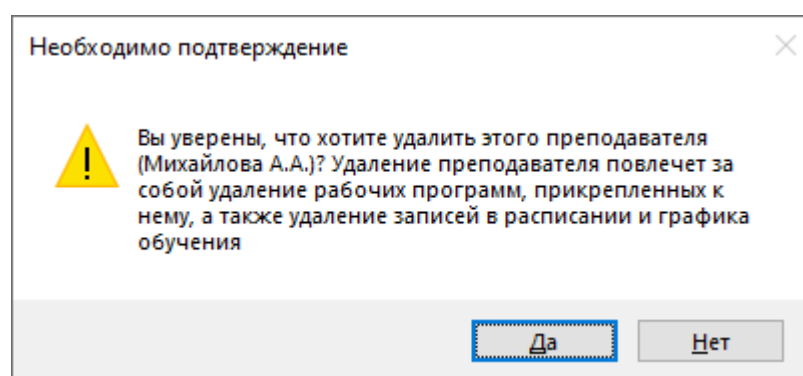


Рисунок 35. Подтверждение удаления конкретного преподавателя.

- Удаление набора преподавателей – Для этой функции сначала необходимо пометить тех преподавателей, которых необходимо удалить, этого можно достигнуть благодаря элементам CheckBox, стоящих перед именем преподавателя. Также существует возможность выбора всех преподавателей с помощью элемента ComboBox расположенного над списком преподавателей. Выбрав необходимых преподавателей, можно произвести над ними некоторые действия, например - удалить (Рис. 36). Для совершения этого действия также необходимо подтверждение (Рис. 37).

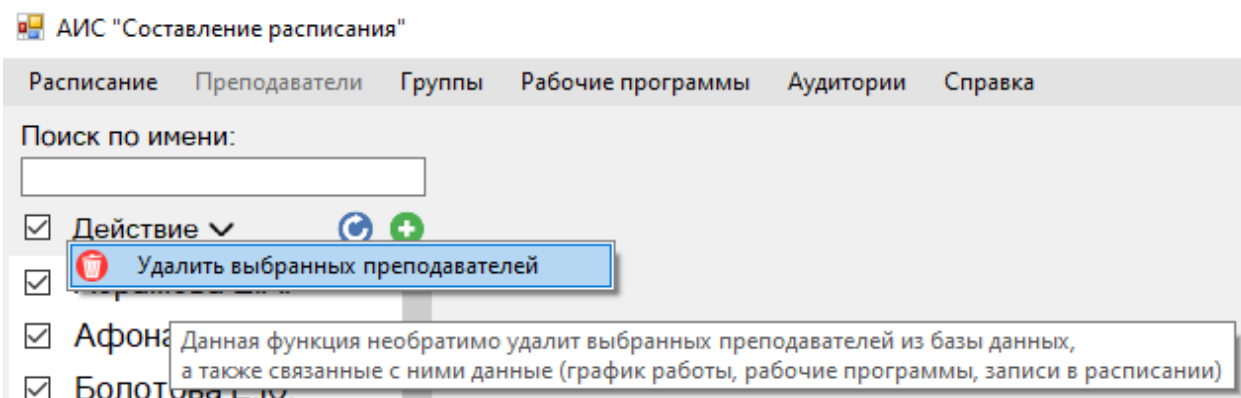


Рисунок 36. Демонстрация функции удаления набора преподавателей.

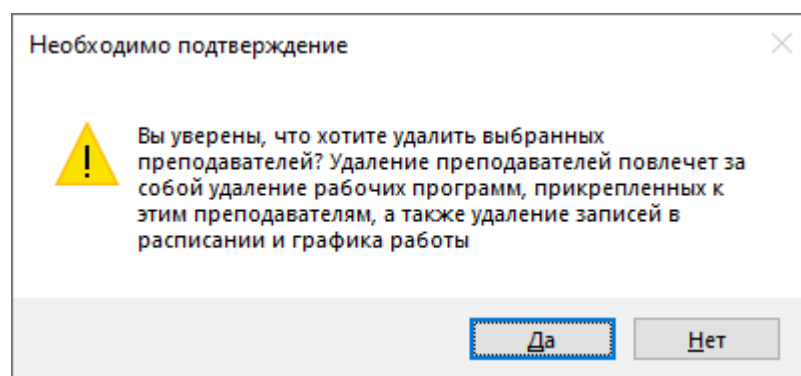


Рисунок 37. Подтверждение удаления набора преподавателей.

В интерфейсе списка преподавателей также существует кнопка обновления в виде иконки со стрелкой,двигающейся по окружности в синем круге. При наведении курсора на кнопку можно прочесть что делает данная кнопка (Рис. 38). Щелкнув по данной кнопке, произойдет обновление списка преподавателей, т.е. программа заново загрузит данные о преподавателях и перестроит список на основе новых данных.

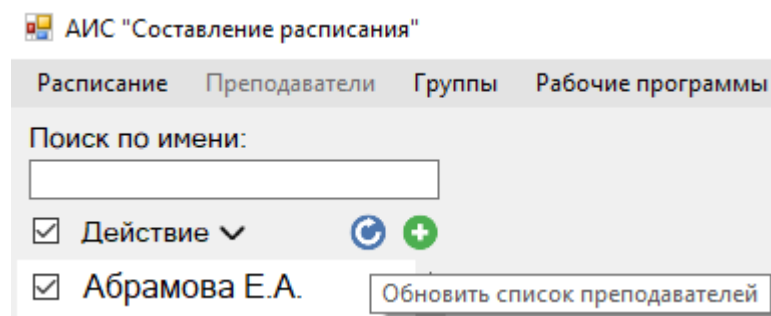


Рисунок 38. Обновление списка преподавателей.



Щелкнув по имени преподавателя в списке, можно более детально рассмотреть данную сущность (Рис. 30). В верхней части представлена основная информация о преподавателе, его фамилия, имя и отчество, которые можно свободно редактировать (Рис. 39), после редактирования необходимо нажать на кнопку «применить», для сохранения новых данных. Сущность преподавателей также имеет логин и пароль, для безопасности эти данные нельзя просматривать свободно, но можно открыть форму для изменения аунтификационных данных (Рис. 40). Эта форма предназначена для изменения логина и пароля преподавателя. Поле логина автоматически заполняется существующим логином преподавателя.

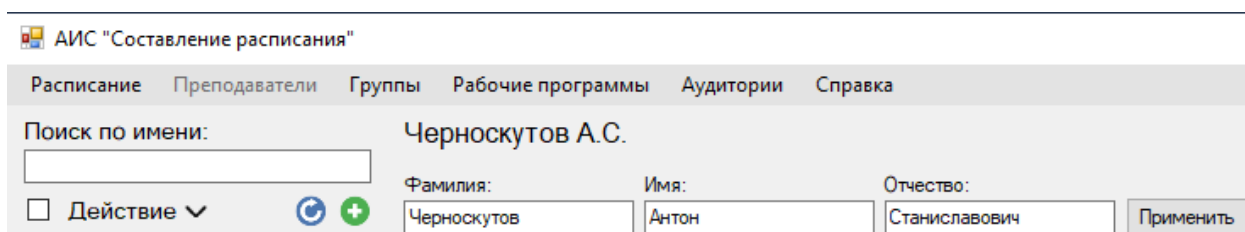


Рисунок 39. Поля для свободного редактирования данных о преподавателе.

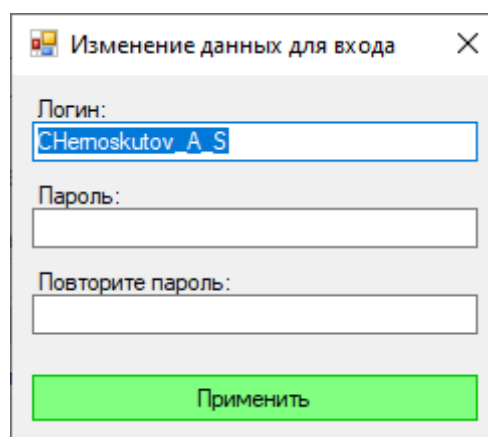


Рисунок 40. Форма для изменения аунтификационных данных.

Оставшуюся часть окна занимает редактор графика работы преподавателя (Рис 41), аналогичный редактор встретится еще в разделе «Группы».

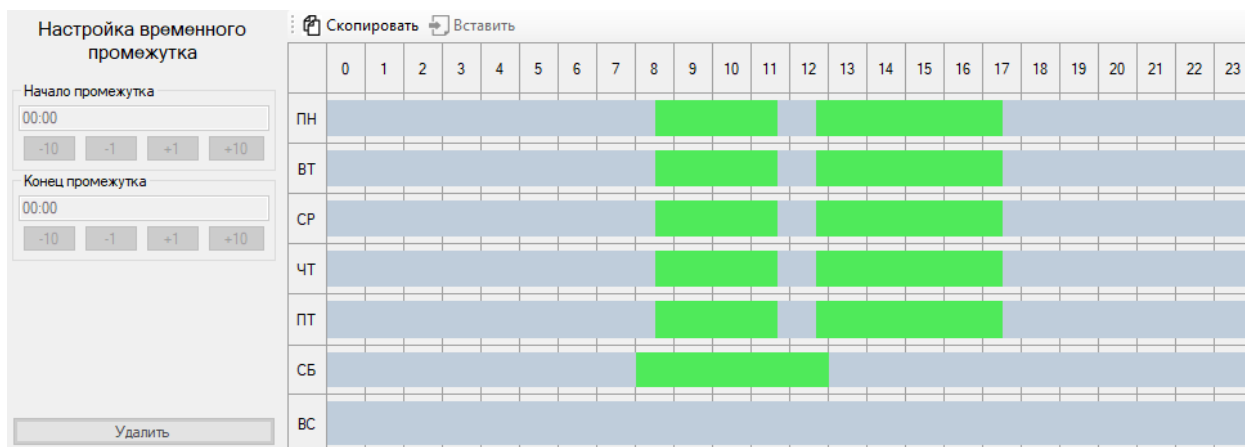


Рисунок 41. Редактор графика работы преподавателя.

В правой части редактора представлено 7 временных осей, каждая ось отражает график работы преподавателя за конкретный день, в совокупности 7 временных осей формируют график работы преподавателя на неделю. Далее недельный график работы преподавателя распространяется на любую реальную неделю в календаре. На временной оси располагаются зеленые временные отрезки, которые обозначают рабочее время преподавателя. Над временными осями равномерно расположены 24 столбца, обозначающие 24 часа. Данные столбцы визуальнo делят временные оси на часы, чтобы легко ориентироваться во временных отрезках.

Для настройки существующего временного отрезка необходимо сначала кликнуть по нему, чтобы выбрать (Рис. 42).

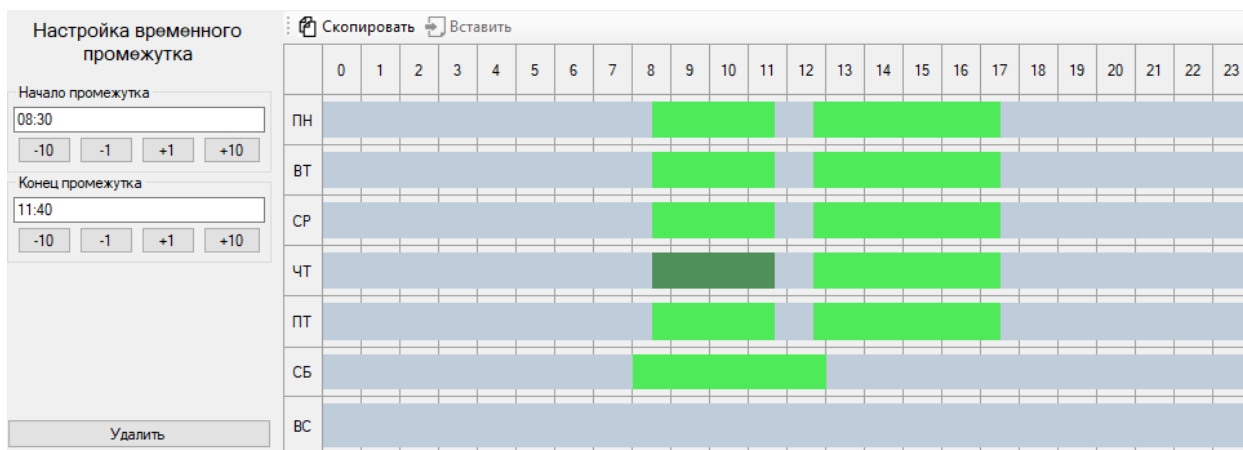


Рисунок 42. Редактор графика работы с выбранным временным отрезком.

Выбрав временной отрезок, он окраситься в более темный цвет, чтобы отображаться как выделенный объект. Также при выделении временного отрезка активируется редактор временного отрезка в левой части редактора графика работы преподавателя. В редакторе временных отрезков можно вручную вписать границы временного отрезка или с помощью кнопок. В данном редакторе также в нижней части присутствует кнопка для удаления отрезка.

Для создания нового временного отрезка необходимо кликнуть в незанятой серой области, на месте клика появиться новый временной промежуток с минимальной длиной.

В интерфейсе также предусмотрено копирование и вставка графика работы преподавателя, т. о. можно настроить график работы одному преподавателю, скопировать его и выставить такой же график работы остальным преподавателям.

### 4.2.2.3. Группы

Данный раздел аналогично является тематическим, т.е. предназначен для управления информационными объектами, в данном случае — это группы. Для этого в левой части окна представлен список групп, отсортированных по наименованию группы (Рис. 43).

Кликнув по наименованию группы, можно более подробно рассмотреть информацию о нем (Рис. 44).

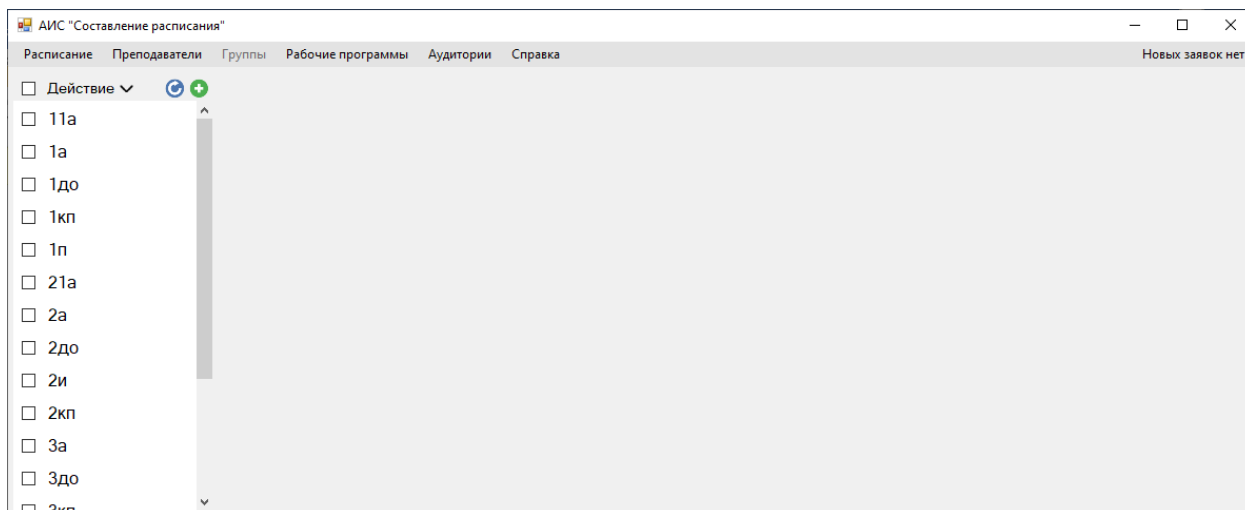


Рисунок 43. Интерфейс раздела группы.

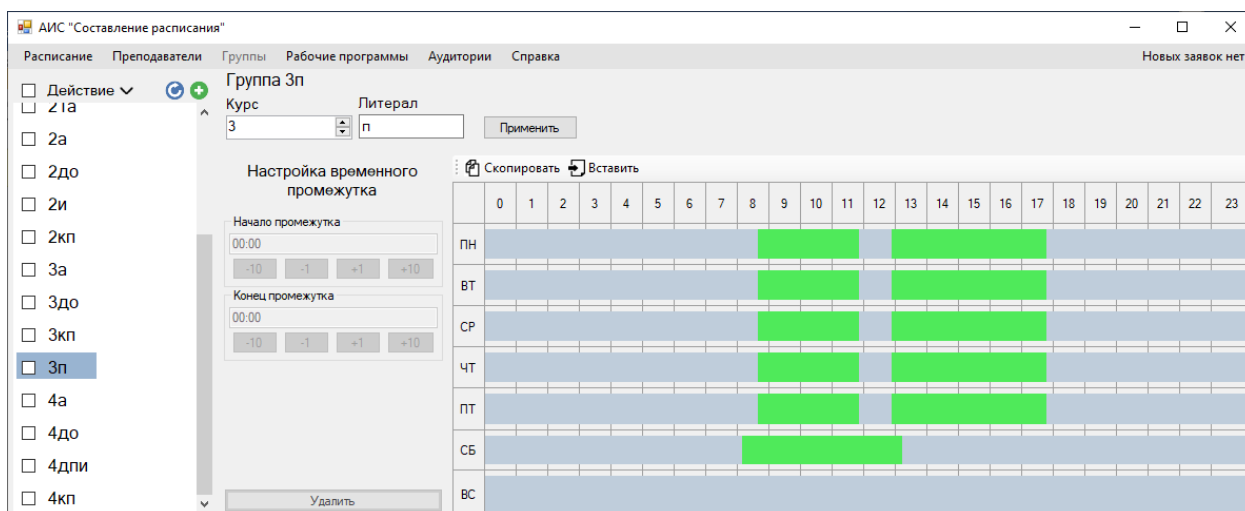


Рисунок 44. интерфейс раздела группы (с выбором группы).

Для создания новой группы существует специальная кнопка-иконка в виде плюса в зеленом круге. При наведении курсора на кнопку пользователю будет представлена текстовая подсказка, описывающая за что она отвечает (Рис. 45).

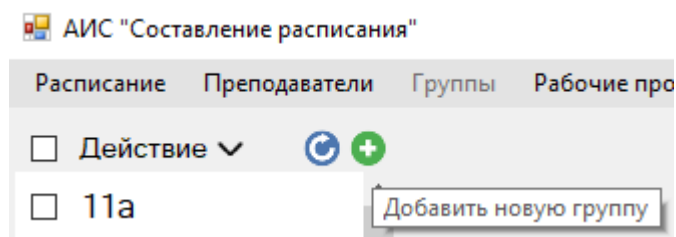


Рисунок 45. Текстовая подсказка кнопки создания группы.

При нажатии по данной кнопке откроется новое диалоговое окно, в котором необходимо ввести литеру группы (Рис. 46). После, на основе введенных данных, можно сформировать новую запись группы в базе данных по нажатию по кнопке «Продолжить». Также существует возможность отменить операцию создания группы, для этого нужно закрыть диалоговое окно, щелкнув по крестику окна, или нажав на кнопку отмены.

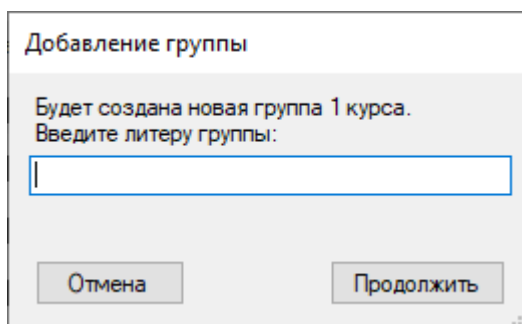


Рисунок 46. Форма создания группы.

Для удаления групп существует 2 способа:

- Удаление конкретной группы – Для удаления потребуется привести курсор на наименование группы, которую необходимо удалить. Далее необходимо вызвать контекстное меню, щелкнув правой кнопкой мыши, находясь курсором над наименованием группы. После остается лишь щелкнуть по пункту контекстного меню «Удалить» (Рис. 47). В процессе удаления программа запросит подтверждения действия (Рис. 48).

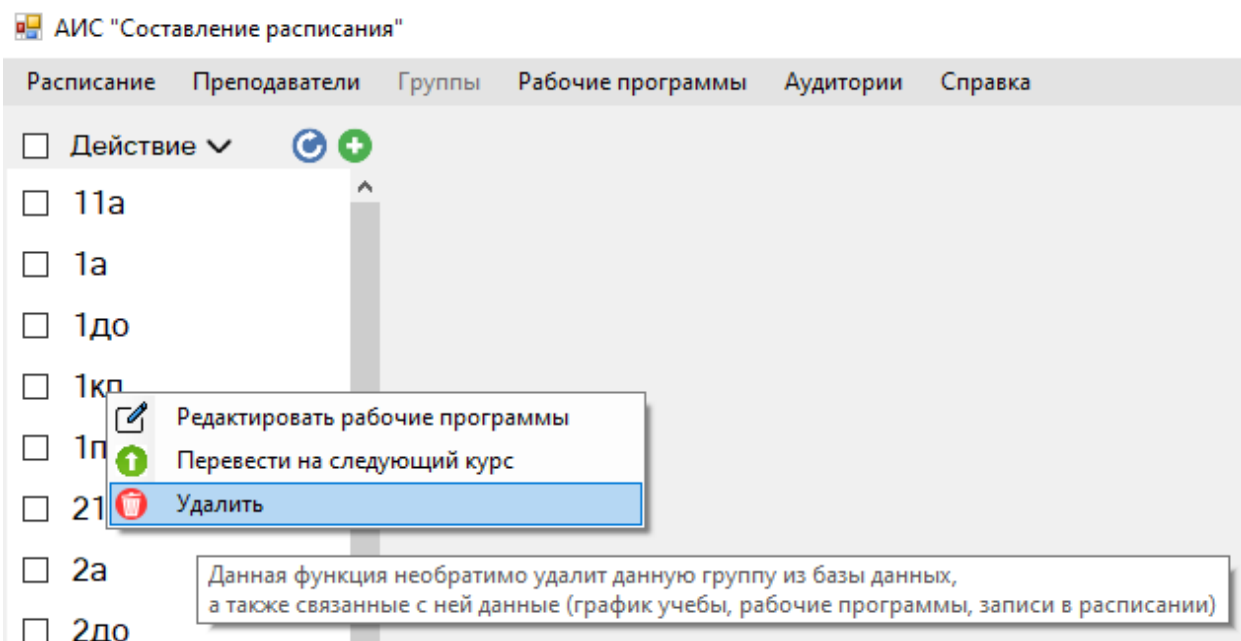


Рисунок 47. Демонстрация функции удаления конкретной группы.

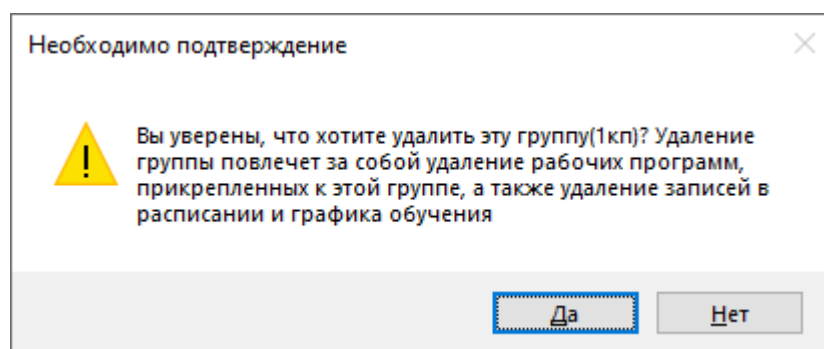


Рисунок 48. Подтверждение удаления конкретного преподавателя.

- Удаление набора групп – Для этой функции сначала необходимо пометить те группы, которые необходимо удалить, этого можно достигнуть благодаря элементам CheckBox, стоящих перед наименованиями групп. Также существует возможность выбора всех групп с помощью элемента ComboBox расположенного над списком групп. Выбрав необходимые группы, можно произвести над ними некоторые действия, например - удалить (Рис. 49). Для совершения этого действия также необходимо подтверждение (Рис. 50).

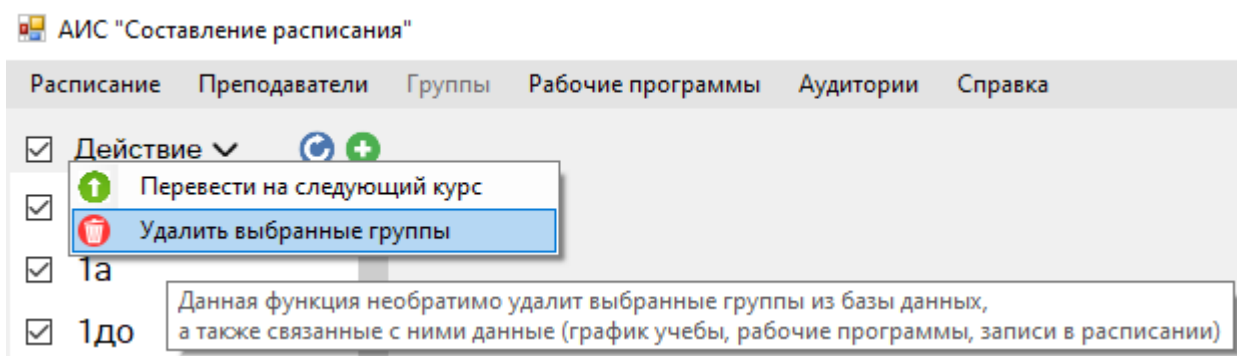


Рисунок 49. Демонстрация функции удаления набора групп.

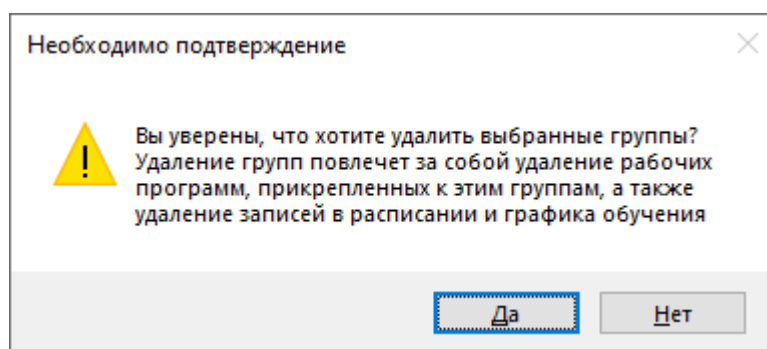


Рисунок 50. Подтверждение удаления набора групп.

В интерфейсе списка групп также существует кнопка обновления в виде иконки со стрелкой,двигающейся по окружности в синем круге. При наведении курсора на кнопку можно прочитать что делает данная кнопка (Рис. 51). Щелкнув по данной кнопке, произойдет обновление списка групп, т.е. программа заново загрузит данные о группах и перестроит список на основе новых данных.

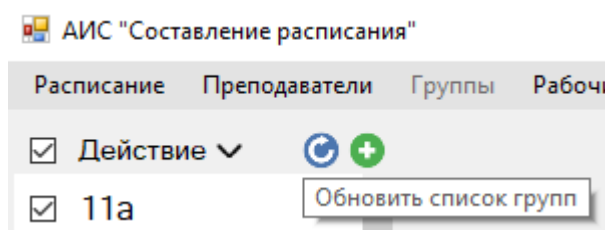


Рисунок 51. Обновление списка групп.

Так же в интерфейсе существует возможность перевода группы на следующий курс. Данная возможность представлена в 2 вариациях: перевод

конкретной группы на следующий курс (Рис. 52) и перевод набора групп на следующий курс (Рис. 53).

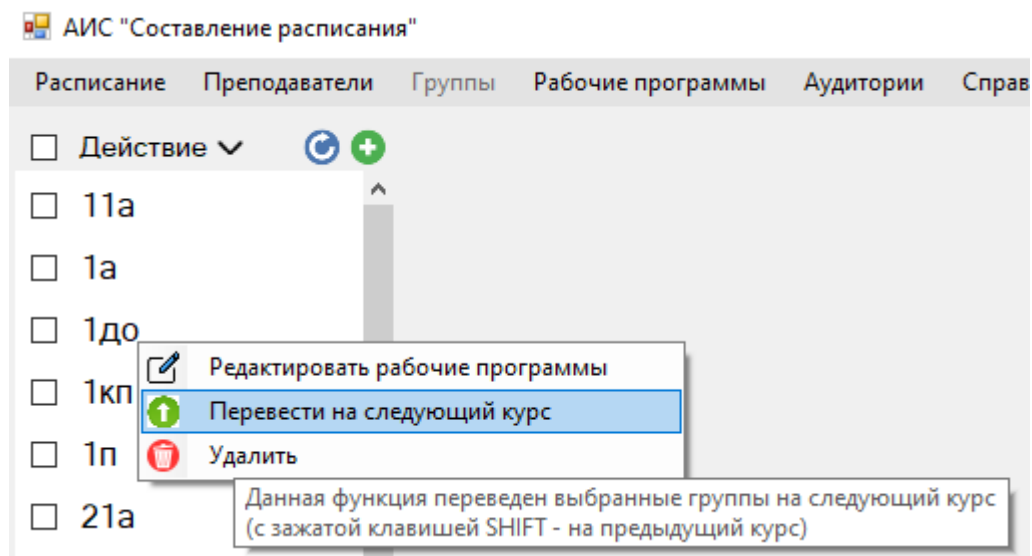


Рисунок 52. Функции перевода конкретной группы на следующий курс.

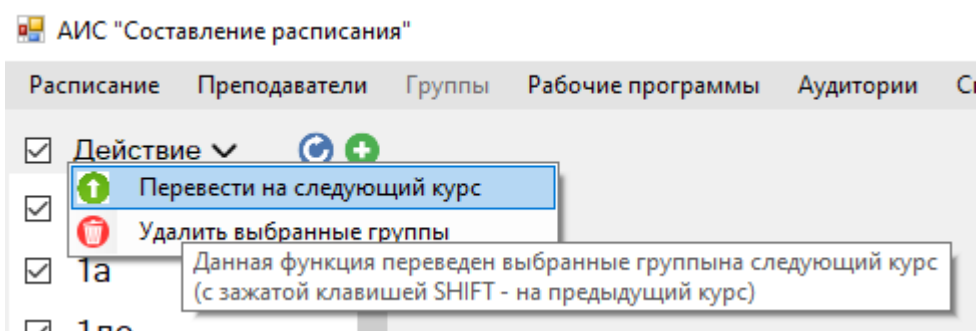


Рисунок 53. Функции перевода набора групп на следующий курс

Дополнительно в интерфейсе существует возможность быстрого перехода к редактированию рабочих программ для некоторой группы (Рис. 54).



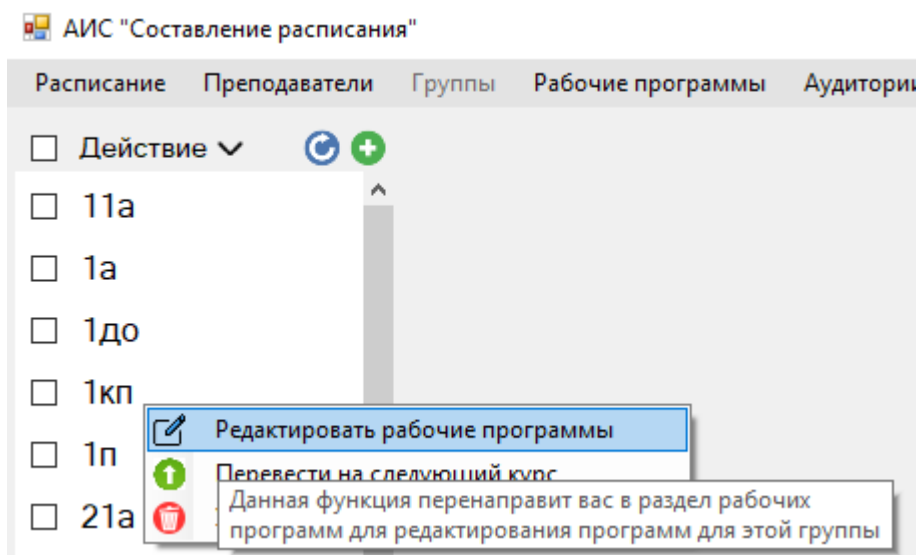


Рисунок 54. Функции перехода к редактированию рабочих программ.

Щелкнув по наименованию группы в списке, можно более детально рассмотреть данную сущность (Рис. 44). В верхней части представлена основная информация о группе, его курс и литера, которые можно свободно редактировать (Рис. 55), после редактирования необходимо нажать на кнопку «применить», для сохранения новых данных.

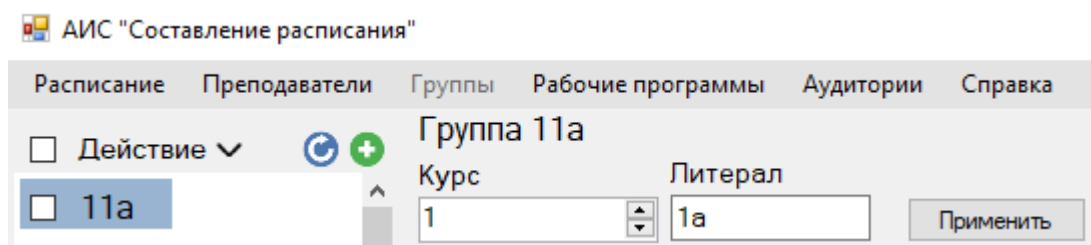


Рисунок 55. Основные поля для свободного редактирования данных о группе.

Оставшуюся часть окна занимает редактор графика учёбы группы (Рис 56), аналогичный редактор уже встречался в разделе «Преподавателя».

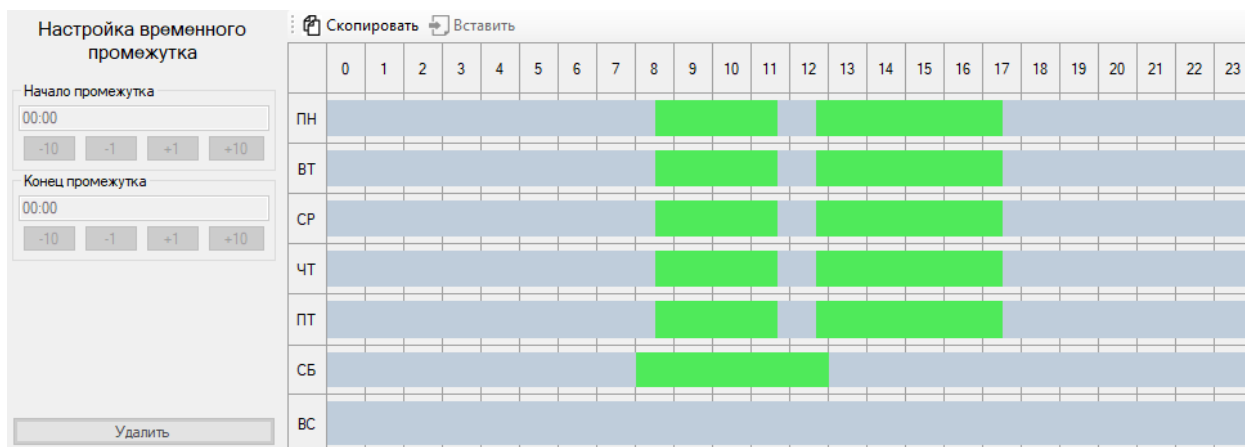


Рисунок 56. Редактор графика учёбы группы.

В правой части редактора представлено 7 временных осей, каждая ось отражает график учёбы группы за конкретный день, в совокупности 7 временных осей формируют график учёбы группы на неделю. Далее недельный график учёбы группы распространяется на любую реальную неделю в календаре. На временной оси располагаются зеленые временные отрезки, которые обозначают учебное время группы. Над временными осями равномерно расположены 24 столбца, обозначающие 24 часа. Данные столбцы визуально делят временные оси на часы, чтобы легко ориентироваться во временных отрезках.

Для настройки существующего временного отрезка необходимо сначала кликнуть по нему, чтобы выбрать (Рис. 57).

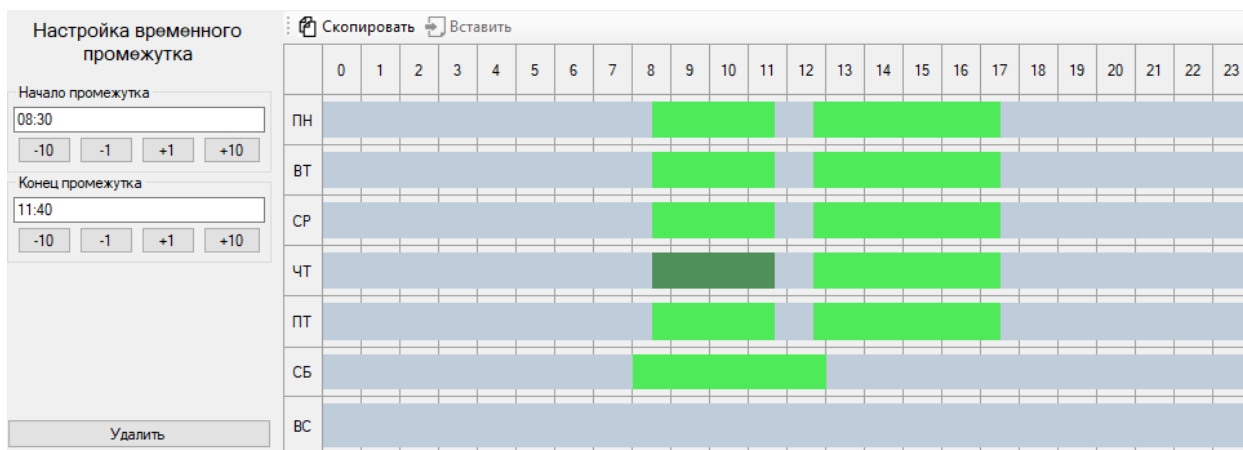


Рисунок 57. Редактор графика учёбы с выбранным временным отрезком.

Выбрав временной отрезок, он окраситься в более темный цвет, чтобы отображаться как выделенный объект. Также при выделении временного отрезка активируется редактор временного отрезка в левой части редактора графика учёбы группы. В редакторе временных отрезков можно вручную вписать границы временного отрезка или с помощью кнопок. В данном редакторе также в нижней части присутствует кнопка для удаления отрезка.

Для создания нового временного отрезка необходимо кликнуть в незанятой серой области, на месте клика появиться новый временной промежуток с минимальной длиной.

В интерфейсе также предусмотрено копирование и вставка графика учёбы группы, т. о. можно настроить график работы одной группы, скопировать его и выставить такой же график учёбы остальным группам.

#### 4.2.2.4. Рабочие программы

Данный раздел аналогично является тематическим, т.е. предназначен для управления информационными объектами, в данном случае — это рабочие программы. Для этого в левой части окна представлен список рабочих программ, отсортированных по полному наименованию рабочей программы и отфильтрованных по группе (Рис. 58).

Кликнув по наименованию рабочей программе, можно более подробно рассмотреть информацию о ней (Рис. 59).

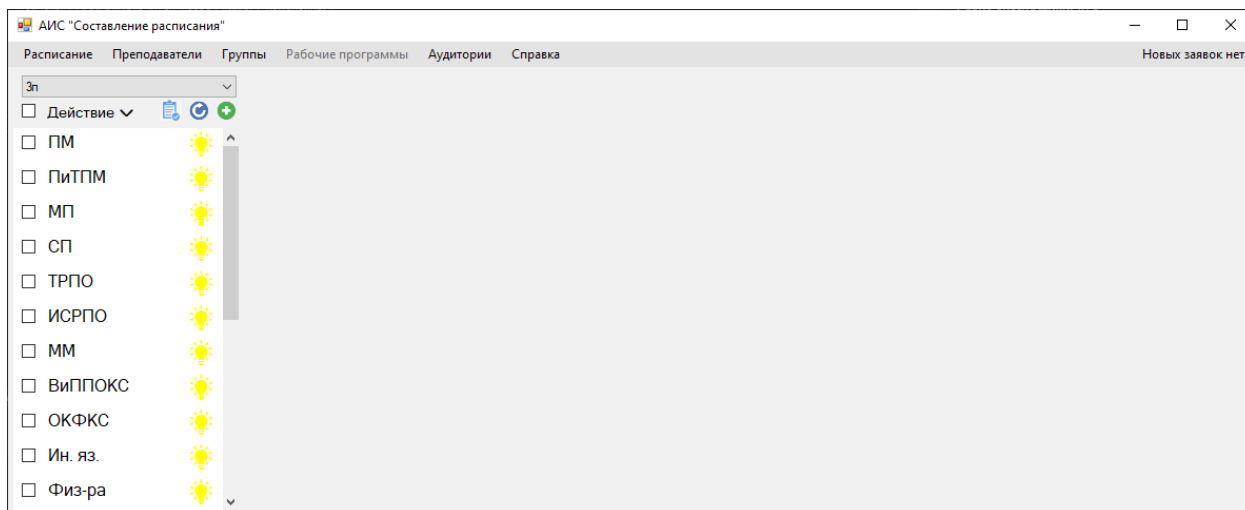


Рисунок 58. Интерфейс раздела рабочих программ.

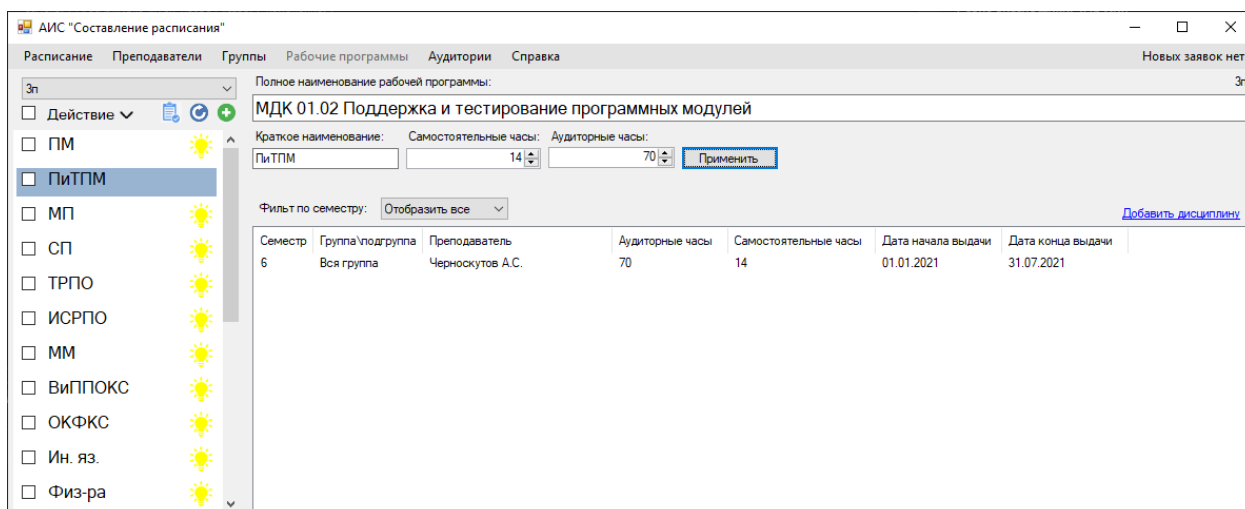


Рисунок 59. интерфейс раздела рабочие программы (с рабочей программы).

Для создания новой рабочей программы существует специальная кнопка-иконка в виде плюса в зеленом круге. При наведении курсора на кнопку пользователю будет представлена текстовая подсказка, описывающая за что она отвечает (Рис. 60).

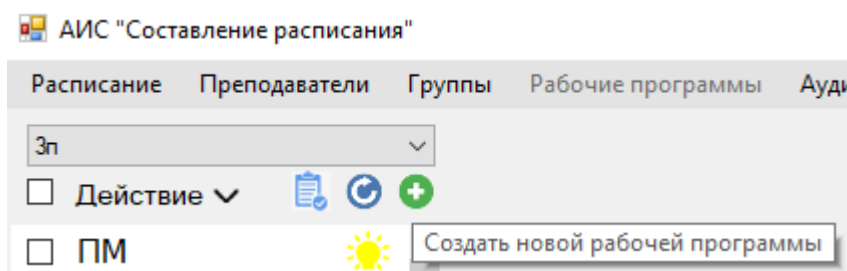


Рисунок 60. Текстовая подсказка кнопки создания рабочей программы.

При нажатии по данной кнопке откроется новое диалоговое окно, в котором необходимо ввести ряд обязательных данных (Рис. 61). После, на основе введенных данных, можно сформировать новую рабочую программу в базе данных по нажатию по кнопке «Создать рабочую программу». Также существует возможность отменить операцию, для этого нужно закрыть диалоговое окно, щелкнув по крестику окна.

Рисунок 61. Форма создания рабочей программы.

Для удаления рабочей программы существует 2 способа:

- Удаление конкретной рабочей программы – Для удаления потребуется навести курсор на краткое наименование рабочей программы в списке, которую необходимо удалить. Далее

необходимо вызвать контекстное меню, щелкнув правой кнопкой мыши, находясь курсором над наименованием рабочей программы. После остается лишь щелкнуть по пункту контекстного меню «Удалить» (Рис. 62). В процессе удаления программа запросит подтверждения действия (Рис. 63).

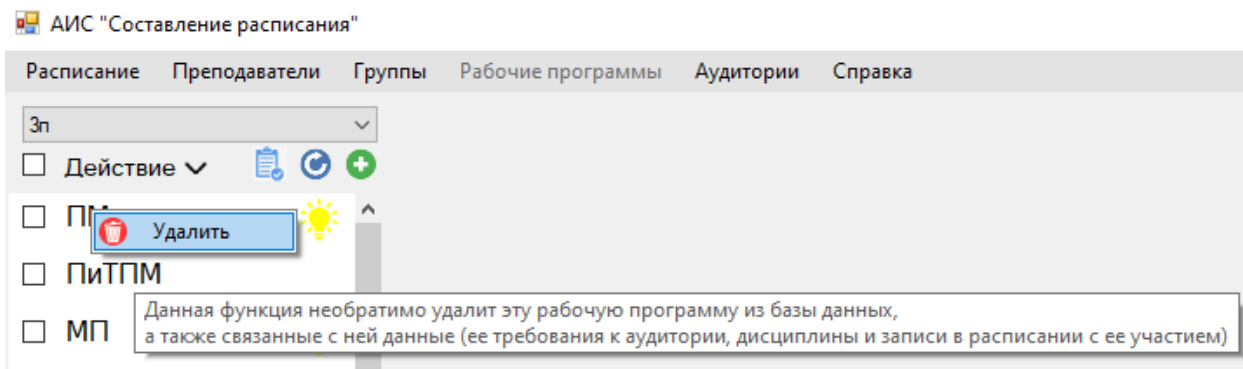


Рисунок 62. Функция удаления конкретной рабочей программы.

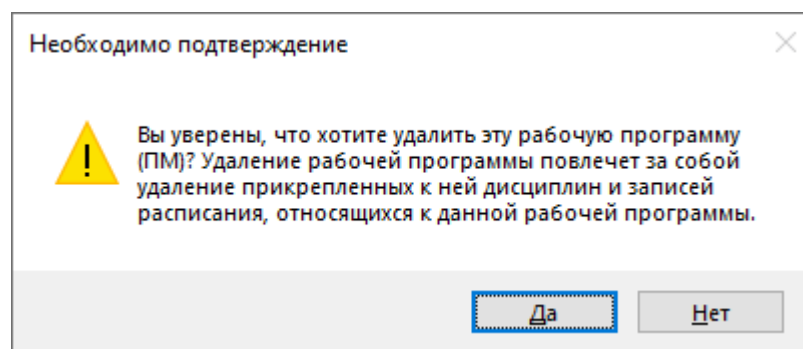


Рисунок 63. Подтверждение удаления конкретной рабочей программы.

- Удаление набора рабочих программ – Для этой функции сначала необходимо пометить те рабочие программы, которые необходимо удалить, этого можно достигнуть благодаря элементам CheckBox, стоящих перед краткими наименованиями рабочих программ. Также существует возможность выбора всех рабочих программ с помощью элемента ComboBox расположенного над списком групп. Выбрав необходимые рабочие программы, можно произвести над ними некоторые

действия, например - удалить (Рис. 64). Для совершения этого действия также необходимо подтверждение (Рис. 65).

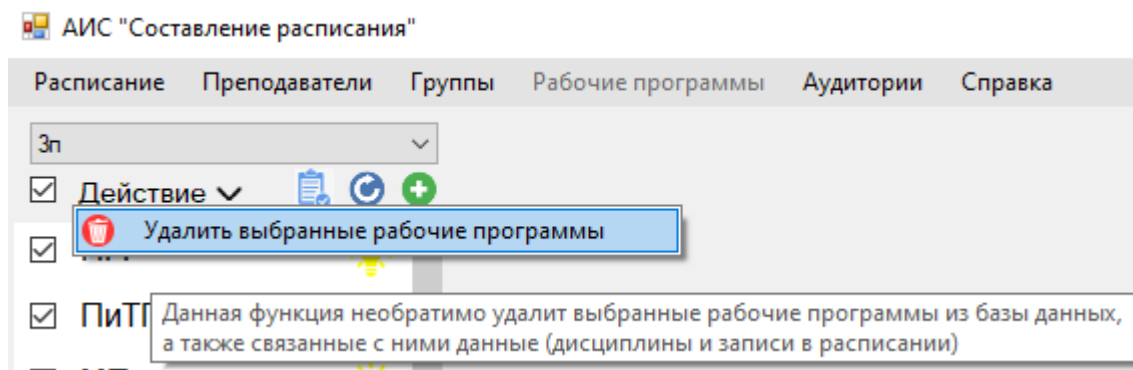


Рисунок 64. Функция удаления набора рабочих программ.

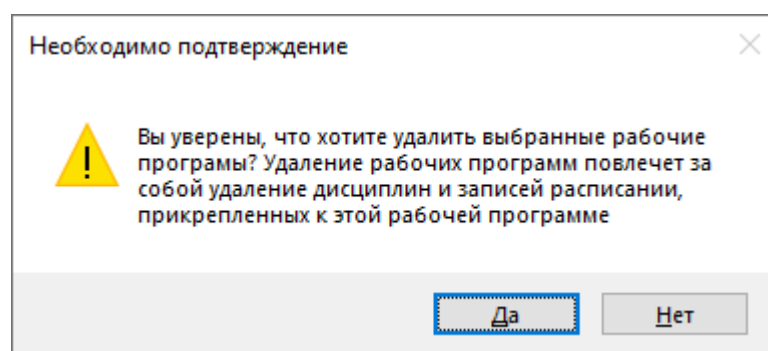


Рисунок 65. Подтверждение удаления набора рабочих программ.

В интерфейсе списка рабочих программ также существует кнопка обновления в виде иконки со стрелкой,двигающейся по окружности в синем круге. При наведении курсора на кнопку можно прочитать что делает данная кнопка (Рис. 66). Щелкнув по данной кнопке, произойдет обновление списка рабочих программ, т.е. программа заново загрузит данные о рабочих программах и перестроит список на основе новых данных.

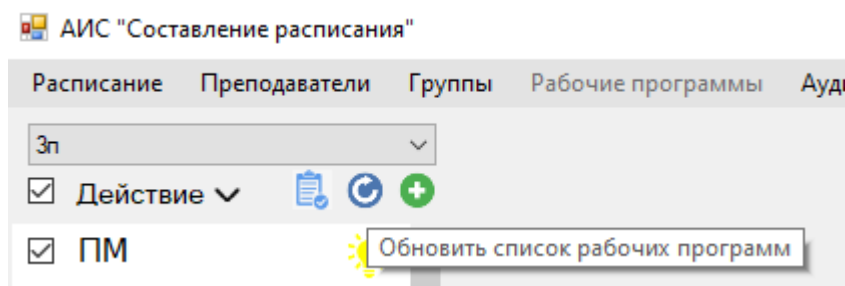


Рисунок 66. Обновление списка рабочих программ.

Для создания новой рабочей программы существует специальная кнопка-иконка. При наведении курсора на кнопку пользователю будет представлена текстовая подсказка, описывающая за что она отвечает (Рис. 67).

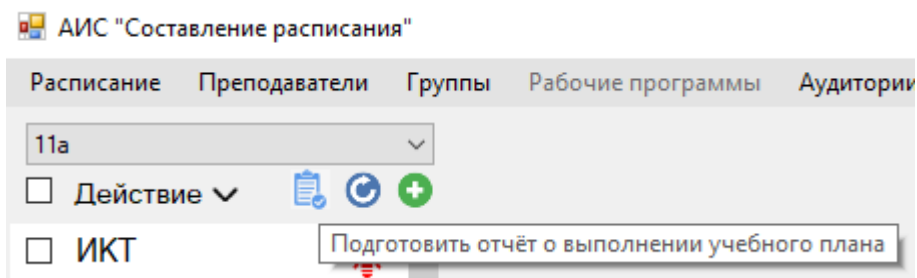


Рисунок 67. Формирование отчета о выполнении учебного плана.

По нажатию данной кнопки откроется новое окно, в котором в верхней части необходимо выбрать семестр, для которого необходимо сформировать отчет. Выбрав семестр, отчет автоматически пересформируется для выбранного семестра (Рис. 68).





Щелкнув по краткому наименованию рабочей программы в списке, можно более детально рассмотреть данную сущность (Рис. 59). В верхней части представлена основная информация о рабочей программе, её полное и краткое наименование, самостоятельные и академические часы, которые можно свободно редактировать (Рис. 69), после редактирования необходимо нажать на кнопку «применить», для сохранения новых данных.

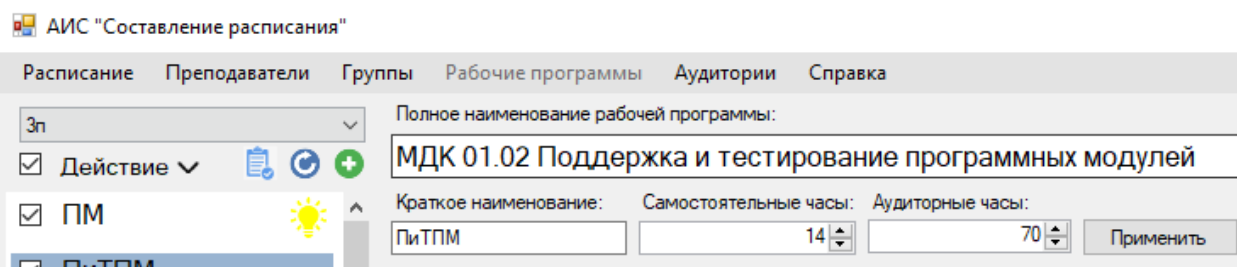


Рисунок 69. Поля для редактирования данных о рабочей программе.

Оставшуюся часть интерфейса отвечает за редактирование дисциплин (Рис. 70). Большую часть пространства занимает элемент ListView, в котором отображаются все существующие дисциплины для выбранной рабочей программы. В интерфейсе предусмотрена фильтрация дисциплин по семестру с помощью элемента ComboBox, расположенного над элементом ListView.

Семестр	Группа\подгруппа	Преподаватель	Аудиторные часы	Самостоятельные часы	Дата начала выдачи	Дата конца выдачи
6	Вся группа	Чернокутов А.С.	70	14	01.01.2021	31.07.2021

[Добавить дисциплину](#)

Рисунок 70. Интерфейс для просмотра дисциплин.

Для добавления новой дисциплины в базу данных существует ссылка «Добавить дисциплину», которая также расположена над элементом ListView, но прижата к правому краю окна. Щелкнув по данной ссылке, откроется новое диалоговое окно для добавления новой дисциплины (Рис. 71).

**МДК 01.02 Поддержка и тестирование программных модулей (3п)**

Начиная с:       Заканчивая:

Выберите порядковый номер семестра:

Дисциплина предназначена для:

Всей группы     Для подгруппы:

Данную дисциплину будет вести:

В течении данной дисциплины необходимо выдать...

Аудиторных часов:

Самостоятельных часов:

Добавить дисциплину

Рисунок 71. Диалоговое окно добавления новой дисциплины.

В открывшемся диалоговом окне пользователю сначала необходимо выбрать с какой даты и до какой даты будет реализовываться данная дисциплина, после нужно ввести порядковый номер семестра. После нужно выбрать для какой части группы будет предназначена данная дисциплина: для всей группы или для некоторой подгруппы? Далее необходимо выбрать преподавателя, который будет вести данную дисциплину, и ввести количество аудиторных и самостоятельных часов. Когда все необходимые данные будут введены нужно будет нажать на кнопку «Добавить дисциплину». После нажатия по этой кнопке в базе данных появится новая запись о дисциплине, а также закроется диалоговое окно и в основном окне приложения обновиться список дисциплин.

Для изменения существующей дисциплины необходимо вызвать контекстное меню у требуемой записи в элементе ListView. В открывшемся контекстном меню нажать на «Редактировать» (Рис. 72).

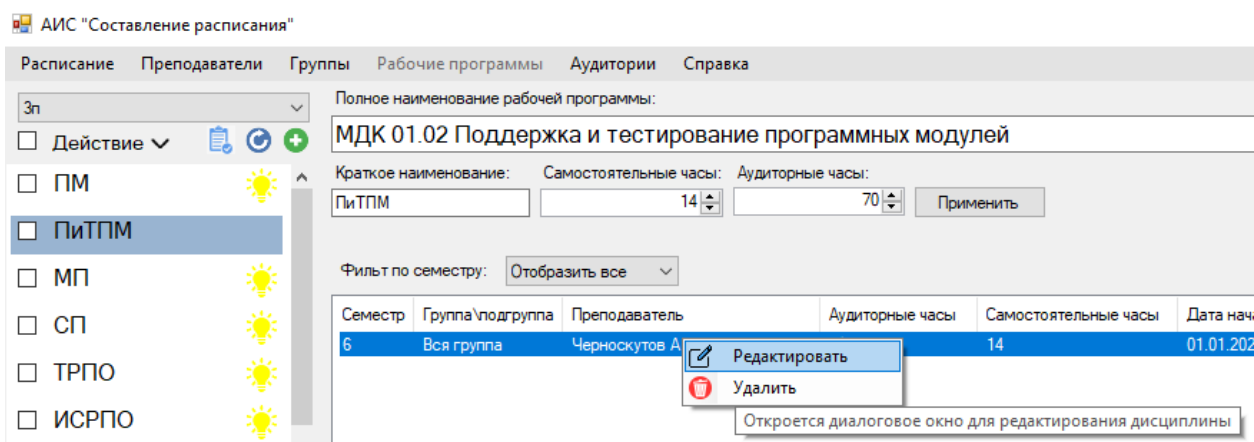


Рисунок 72. Демонстрация функции для редактирования рабочих программ.

После нажатия по пункту контекстного меню «редактировать» откроется диалоговое окно для редактирования дисциплины. В этом окне можно изменить преподавателя, аудиторные и академические часы, даты начала и конца выдачи дисциплины, семестр и подгруппу\группу (Рис. 73). После изменения данных необходимо нажать на кнопку «Применить» для сохранения данных.

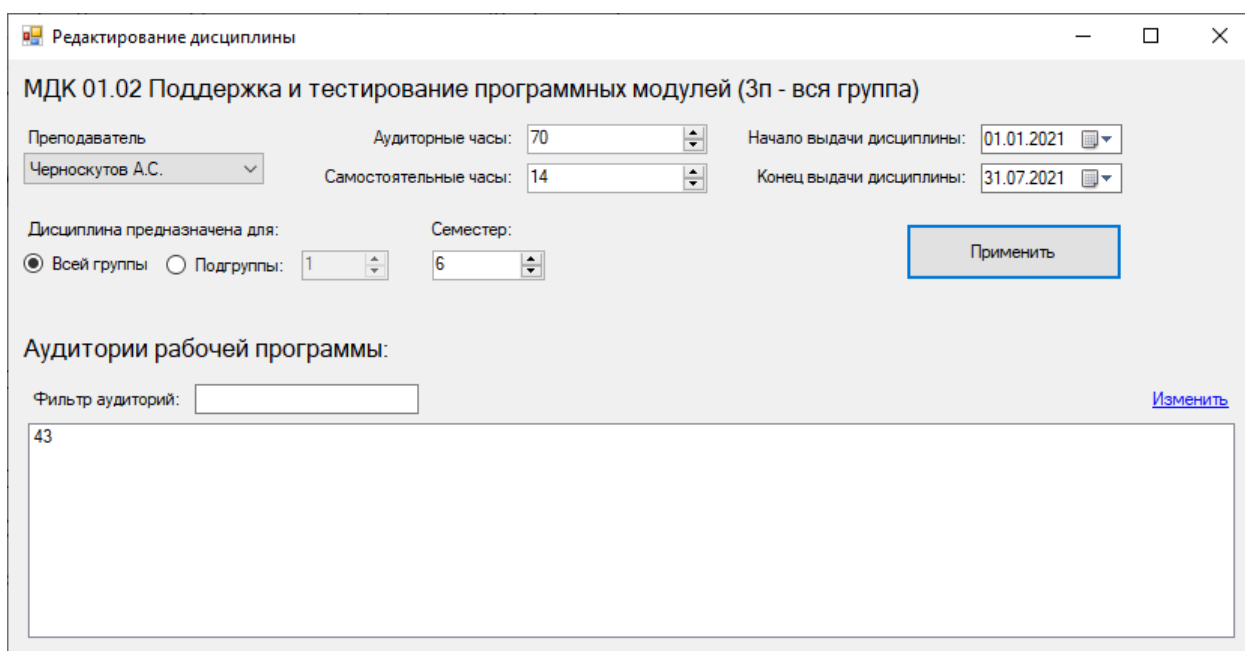


Рисунок 73. Диалоговое окно для редактирования дисциплины.

В этом окне также можно просмотреть список аудиторий, в которых допускается проведение данной дисциплины. Для редактирования этого списка необходимо нажать на ссылку «Изменить», которая расположена над списком аудиторий и прижата в правому краю окна. По нажатию по ссылке список аудиторий перейдет в режим редактирования (Рис. 74).

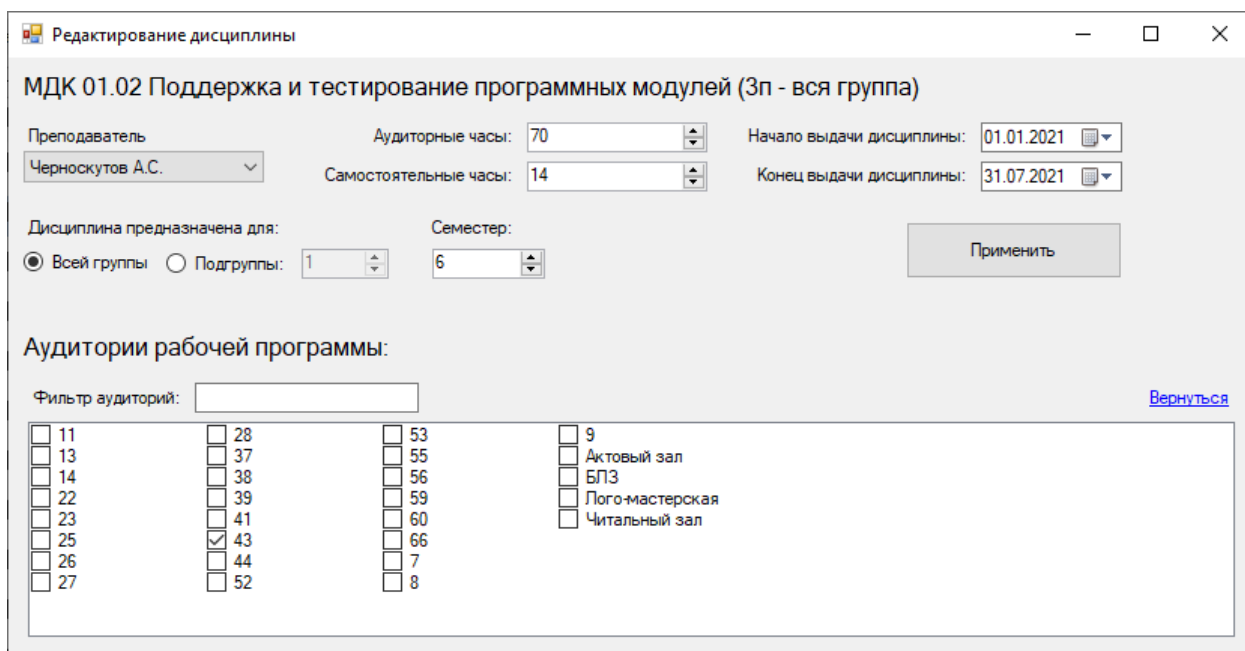


Рисунок 74. Демонстрация списка аудиторий в режиме редактирования.

В режиме редактирования в списке аудиторий будут отображаться все аудитории, рядом с каждой аудиторией находится элемент CheckVox. Чтобы допустить проведение занятий в некоторой аудитории необходимо чтобы элемент CheckVox рядом с требуемой аудиторией был в выбранном состоянии. Следовательно, для недопуска проведения занятий необходимо чтобы элемент CheckVox был в неактивном состоянии у требуемых аудиторий. При изменении состояния элементов CheckVox приложение будет сразу автоматически вносить изменения в базу данных.

Для удаления существующей дисциплины необходимо вызвать контекстное меню у требуемой записи в элементе ListView. В открывшемся контекстном меню нажать на «Удалить» (Рис. 75). При удалении приложение дополнительно запросит подтверждение действия (Рис. 76).

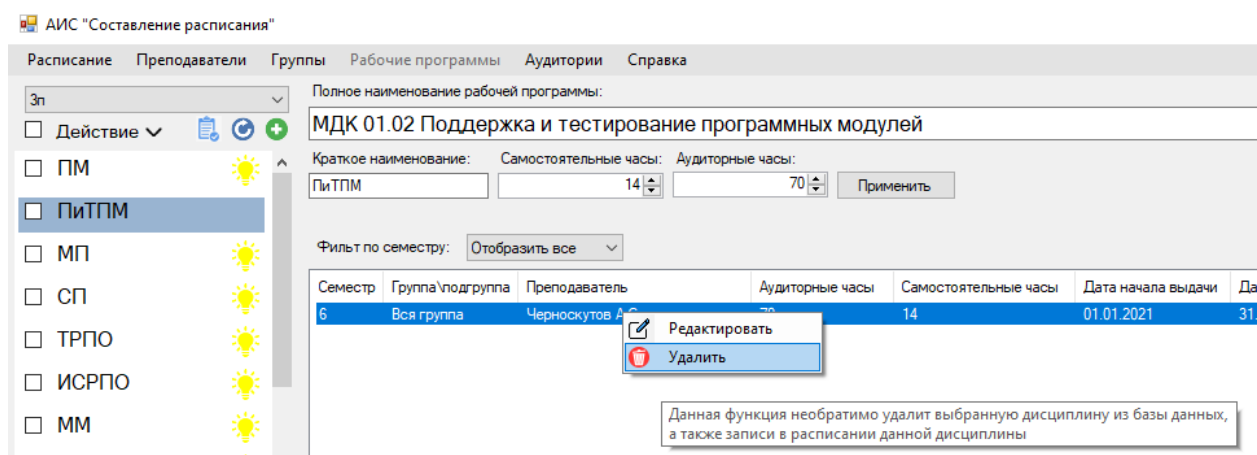


Рисунок 75. Демонстрация функции удаления дисциплины.

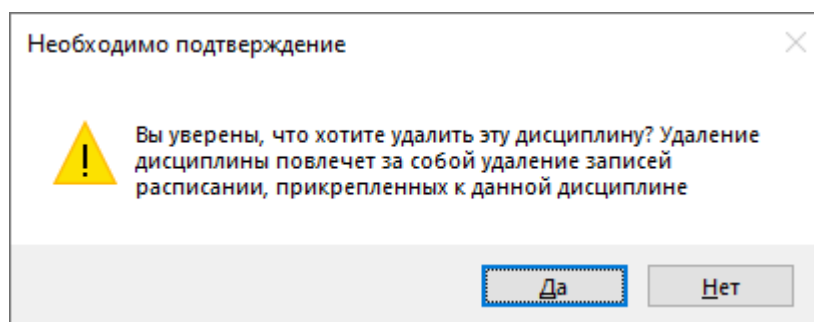


Рисунок 76. Подтверждение удаления дисциплины.

#### 4.2.2.5. Аудитории

Данный раздел также является тематическим, т.е. предназначен для управления информационными объектами, в данном случае — это аудитории. Для этого в левой части окна представлен список всех аудиторий, отсортированных по наименованию аудитории (Рис. 77).

Кликнув по наименованию аудитории, можно более подробно рассмотреть информацию о ней (Рис. 78).

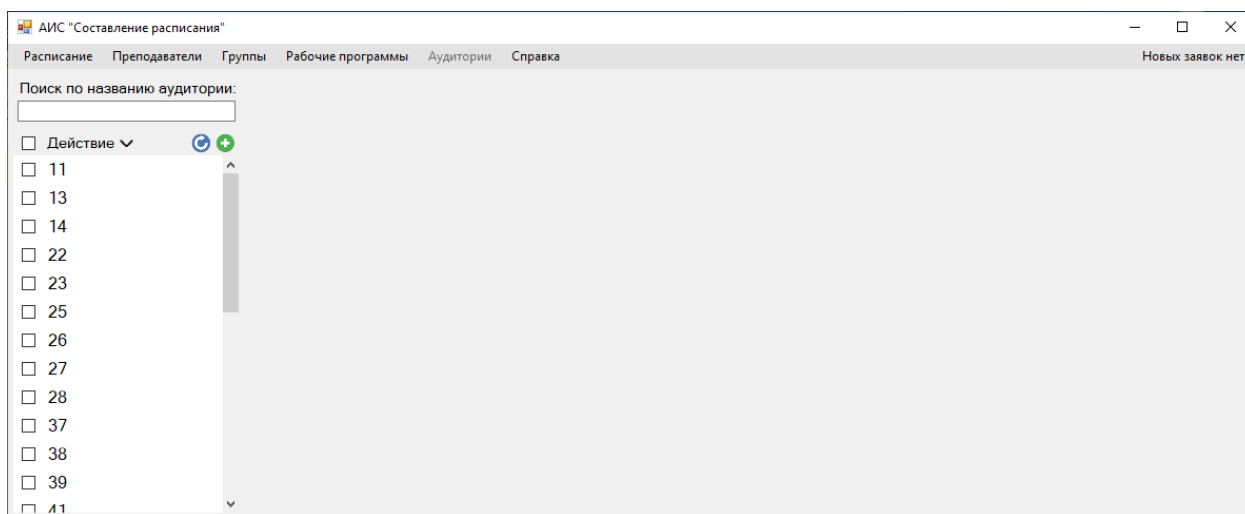


Рисунок 77. Интерфейс раздела аудиторий.

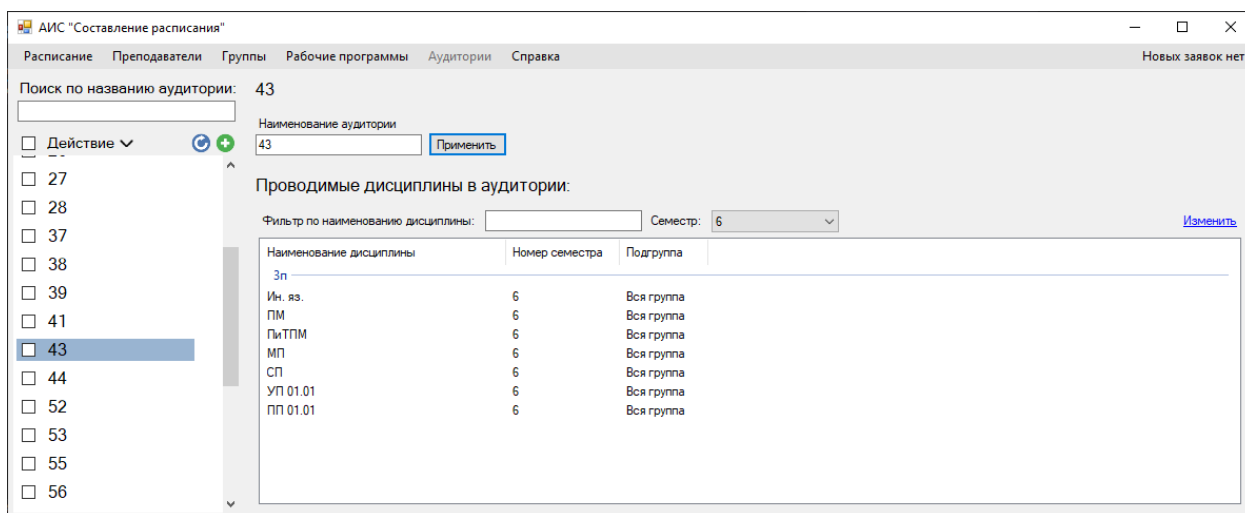


Рисунок 78. Интерфейс раздела аудиторий с выбранной аудиторией.

Для быстрого поиска необходимой аудитории в списке существует функция поиска, для этого необходимо ввести в текстовое поле над списком некоторую часть наименования необходимой аудитории, и программа автоматически будет фильтровать список по мере ввода текста, совершая поиск подстроки в наименовании аудитории (Рис. 79, регистр не учитывается, но требуется вводить поисковую подстроку без ошибок).

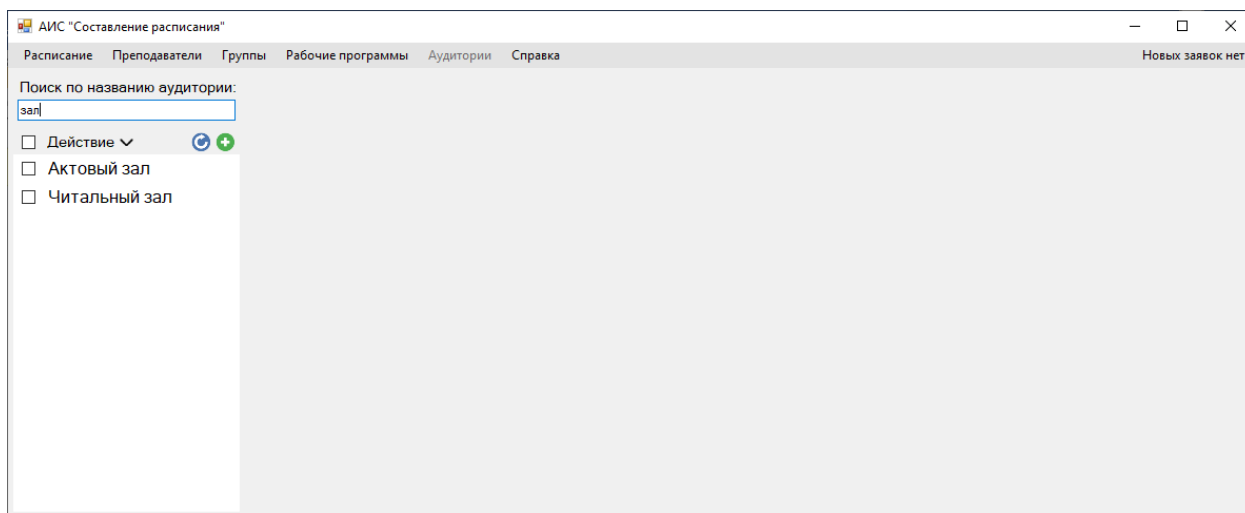


Рисунок 79. Демонстрация функции поиска аудитории.

Для добавления новой аудитории в базу данных существует кнопка-иконка над списком в виде зеленого круга с белым плюсиком, наведя курсор на нее можно увидеть текстовую подсказку, которая говорит за что отвечает данная кнопка (Рис. 80).

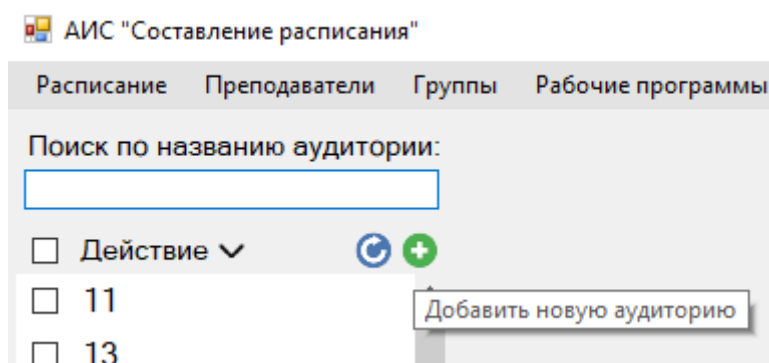


Рисунок 80. Текстовая подсказка у кнопки добавления новой аудитории.



Щелкнув по данной кнопке, откроется диалоговое окно (Рис. 81), в котором можно ввести наименование новой аудитории, после чего, нажав по кнопке «продолжить», в базе данных появится новая аудитория. Данный процесс можно отменить, нажав на кнопку «Отмена», или нажав на крестик диалогового окна.

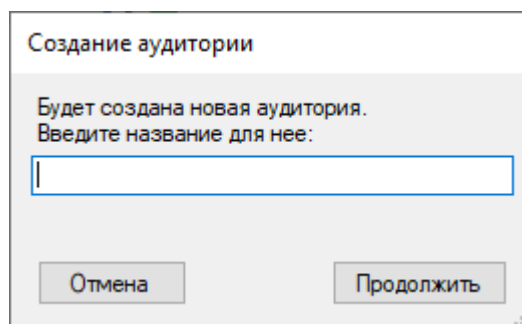


Рисунок 81. Диалоговое окно добавления новой аудитории в базу данных.

Для удаления аудитории из базы данных существует 2 способа:

- Удаление конкретной аудитории (Рис. 82) – Для этого необходимо вызвать контекстное меню у аудитории в списке, которую необходимо удалить и нажать по соответствующему пункту контекстного меню. При удалении программа дополнительно запросит подтверждение на удалении аудитории (Рис 83).

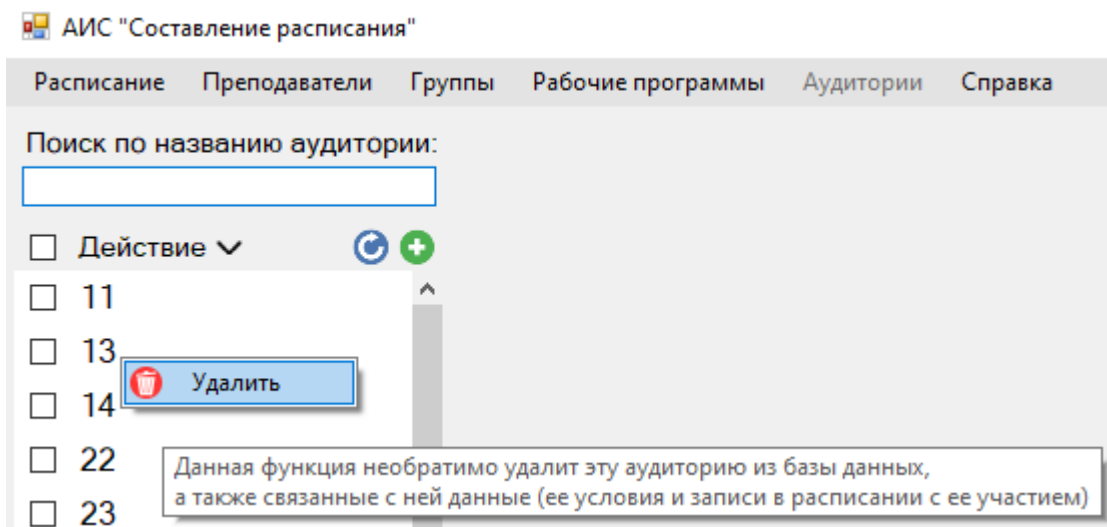


Рисунок 82. Демонстрация функции удаления конкретной аудитории.

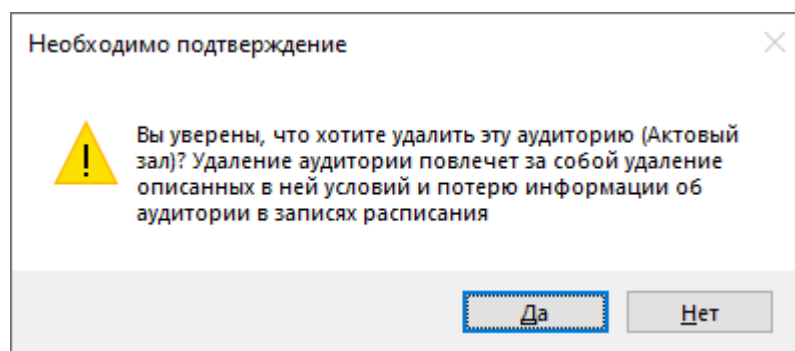


Рисунок 83. Запрос подтверждения удаления конкретной аудитории.

- Удаление набора аудиторий (Рис. 84) – Для этого необходимо сначала определить какие аудитории будут входить в набор для удаления, чтобы выбрать или наоборот снять выделение со всех аудиторий можно нажать на элемент CheckBox, который находится над списком, также существует возможность точечного выделения аудиторий, путем щелчков по CheckBox находящихся рядом с именами аудиторий. Выделив необходимые аудитории, можно приступить к удалению выделенных аудиторий, для этого необходимо раскрыть список действий над списком и выбрать соответствующий пункт раскрывшегося меню. Программа дополнительно запросит подтверждение перед удалением аудиторий (Рис. 85).

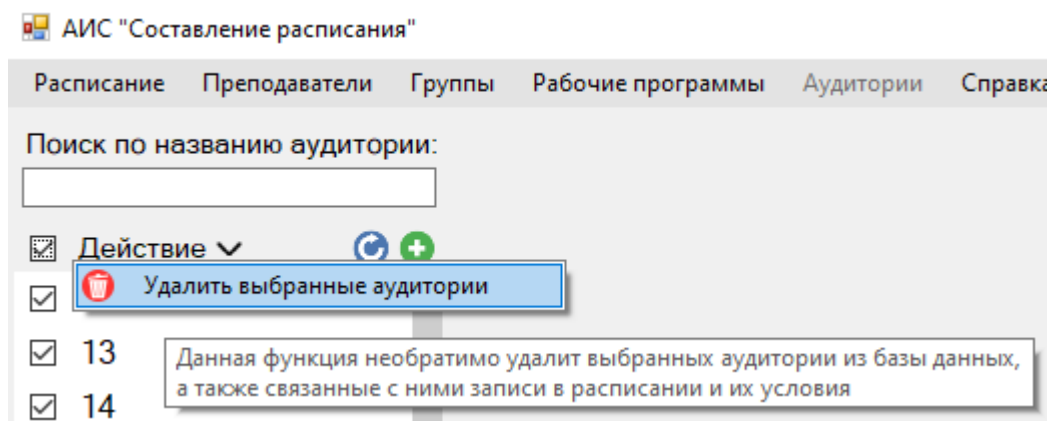


Рисунок 84. Демонстрация функции удаления набора аудиторий.

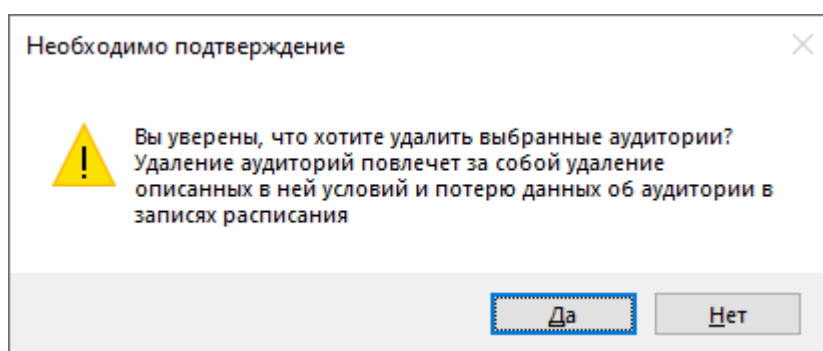


Рисунок 85. Демонстрация окна подтверждения удаления набора аудиторий.

Щелкнув по имени аудитории в списке, в остальной части окна появиться интерфейс для взаимодействия с данной аудиторией.

В верхней части можно свободно изменить наименование аудитории (Рис. 86).

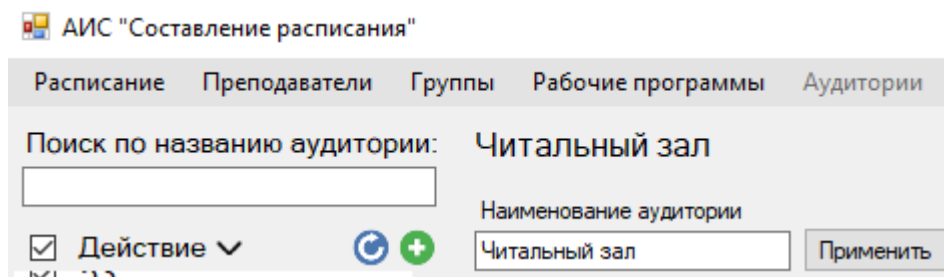


Рисунок 86. Демонстрация свободного изменения наименования аудитории.

Оставшуюся часть интерфейса занимает список дисциплин, которым допускается проведение занятий в данной аудитории. Над списком

предусмотрено 2 фильтра: по наименованию рабочей программы (регистр не учитывается) и по семестру (Рис. 87).

Проводимые дисциплины в аудитории:

Фильтр по наименованию дисциплины:  Семестр: 6 [Изменить](#)

Наименование дисциплины	Номер семестра	Подгруппа
<b>3п</b>		
Ин. яз.	6	Вся группа
ПМ	6	Вся группа
ПлТПМ	6	Вся группа
МП	6	Вся группа
СП	6	Вся группа
УП 01.01	6	Вся группа
ПП 01.01	6	Вся группа

Рисунок 87. Список дисциплина, проводимых в данной аудитории.

Здесь же можно отредактировать этот список дисциплин, нажав на ссылку над списком «Изменить». При нажатии ссылки список перейдет в режим редактирования (Рис. 88).

АИС "Составление расписания"

Расписание Преподаватели Группы Рабочие программы Аудитории Справка

Поиск по названию аудитории: 43

Наименование аудитории: 43

Действие  23  26  27  28  37  38  39  41  43  44  52  53  55  56  59  60  66  7  8  9  Актовый зал  БЛЗ  Лого-мастерская  Читальный зал

Проводимые дисциплины в аудитории:

Фильтр по наименованию дисциплины:  Семестр: 6 [Сохранить](#)

Наименование дисциплины	Номер семестра	Подгруппа
<b>1кл</b>		
<input type="checkbox"/> Теор.мат.развит	6	Вся группа
<input type="checkbox"/> Осн.лог.по звуко	6	Вся группа
<input type="checkbox"/> Орг.лог.подгото	6	Вся группа
<input type="checkbox"/> Сов.вед.тех.обр	6	Вся группа
<input type="checkbox"/> Об.деял.дет.	6	Вся группа
<input type="checkbox"/> Орг.образ.деял	6	Вся группа
<input type="checkbox"/> Основ.воспит.род	6	Вся группа
<input type="checkbox"/> План.взаим.семей	6	Вся группа
<input type="checkbox"/> Осн.проект.исл.дея	6	Вся группа
<b>1п</b>		
<input type="checkbox"/> дет.лит.	6	Вся группа
<input type="checkbox"/> ТО	6	Вся группа
<input type="checkbox"/> методика	6	Вся группа
<input type="checkbox"/> ТИМФВ	6	Вся группа
<input type="checkbox"/> Т и Муз.	6	Вся группа
<b>3п</b>		
<input checked="" type="checkbox"/> Ин. яз.	6	Вся группа
<input type="checkbox"/> Физ-ра	6	Вся группа
<input checked="" type="checkbox"/> ПМ	6	Вся группа
<input checked="" type="checkbox"/> ПлТПМ	6	Вся группа
<input checked="" type="checkbox"/> МП	6	Вся группа
<input checked="" type="checkbox"/> СП	6	Вся группа
<input checked="" type="checkbox"/> УП 01.01	6	Вся группа
<input checked="" type="checkbox"/> ПП 01.01	6	Вся группа

Рисунок 88. Список дисциплин в режиме редактирования.

В режиме редактирования в списке дисциплин будут отображаться все дисциплины, удовлетворяющие фильтрам, рядом с каждой дисциплиной находиться элемент CheckBox. Чтобы допустить проведение занятий

некоторой дисциплины в этой аудитории необходимо чтобы элемент CheckBox рядом с требуемой дисциплиной был в выбранном состоянии. Следовательно, для недопуска проведения занятий некоторой дисциплины необходимо чтобы элемент CheckBox был в неактивном состоянии у требуемой дисциплины. Для сохранения изменений необходимо нажать на ссылку «Сохранить». При нажатии на ссылку список выйдет из режима редактирования и все изменения сохраняться.

#### **4.2.2.6. Справка**

Данный раздел является информационным, он не привносит никакого функционала. Здесь можно ознакомиться с приложением: о том, как с ним работать, какие существуют в нем функции и т.д.

Все руководство пользователя разделено на 6 разделов: Общие сведения (Рис. 89) и 5 других разделов, в которых описано руководство пользователя для соответствующего раздела приложения (Рис. 90).

Разделы руководства расположены в ином порядке, в отличии от главного меню приложения. Они расположены в таком порядке для постепенного изучения приложения, т. о. для составления расписания потребуется изучить все остальные разделы, т.к. без, например, преподавателей невозможно добавить занятие в расписании.

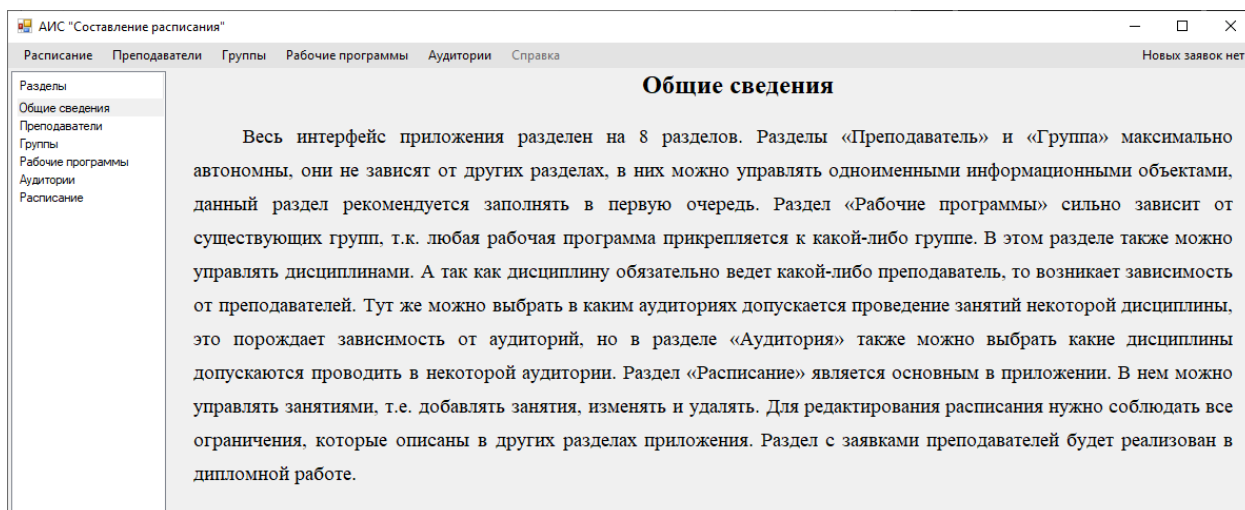


Рисунок 89. Раздел справки.

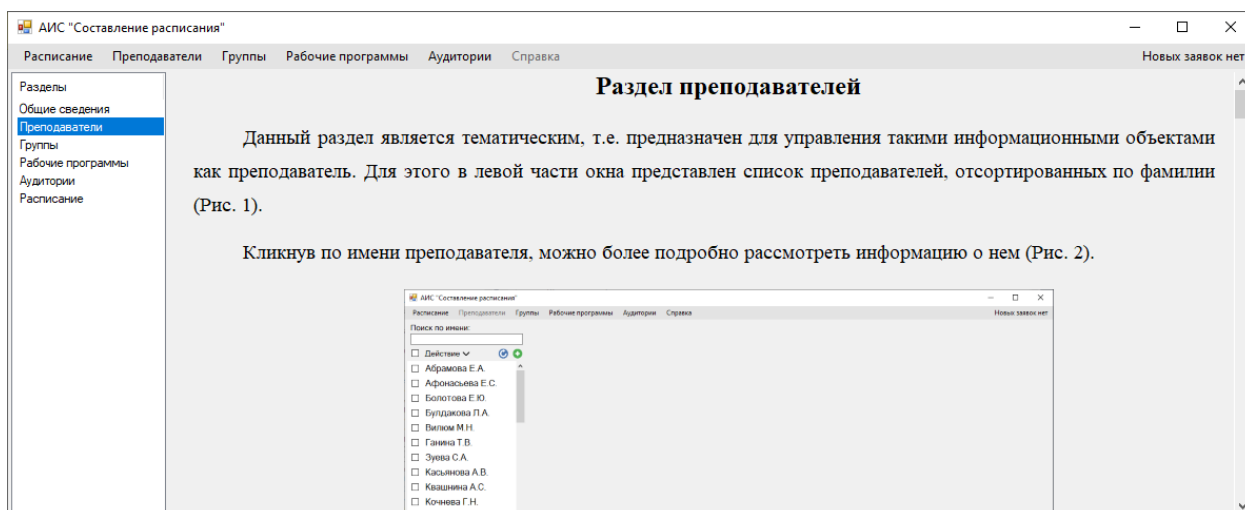


Рисунок 90. Раздел справки преподавателей.

## ТЕСТИРОВАНИЕ ПРОГРАММНОГО ПРОДУКТА

### 5.1. Тестирование по классам эквивалентности входных данных

Рисунок 91. Форма "Создание преподавателя", подвергаемая тестированию

Входные условия	Классы эквивалентности	
	Правильные	Неправильные
Значение поля «Фамилия» в форме «Создания пользователя» для указания фамилии добавляемого преподавателя	текстовое значение длиной от 1 до 50 символов (1);	не введено (2); текстовое значение длиной более 50 символов (3);
Значение поля «Имя» в форме «Создания пользователя» для указания имени добавляемого преподавателя	текстовое значение длиной от 1 до 50 символов (4);	не введено (5); текстовое значение длиной более 50 символов (6);
Значение поля «Отчество» в форме	текстовое значение длиной от 1 до 50	не введено (8); текстовое значение

«Создания пользователя» для указания отчества добавляемого преподавателя	символов (7);	длиной более 50 символов (9);
Значение поля «Логин» в форме «Создания пользователя» для указания уникального логина добавляемого преподавателя	Уникальное текстовое значение длиной от 8 до 64 символов (10);	текстовое значение длиной менее 8 символов (11); текстовое значение длиной более 64 символов (12);
Значение поля «Пароль» в форме «Создания пользователя» для указания пароля добавляемого преподавателя	текстовое значение длиной от 8 до 128 символов (13);	текстовое значение длиной менее 8 символов (14); текстовое значение длиной более 128 символов (15);
Значение поля «Повторите пароль» в форме «Создания пользователя» для проверки введенного пароля	Текстовое значение совпадает с текстовым значением поля «Пароль» (16);	Несовпадающие текстовые значения полей «Пароль» и «Повторите пароль» (17);

### **Тест для правильных классов эквивалентности**

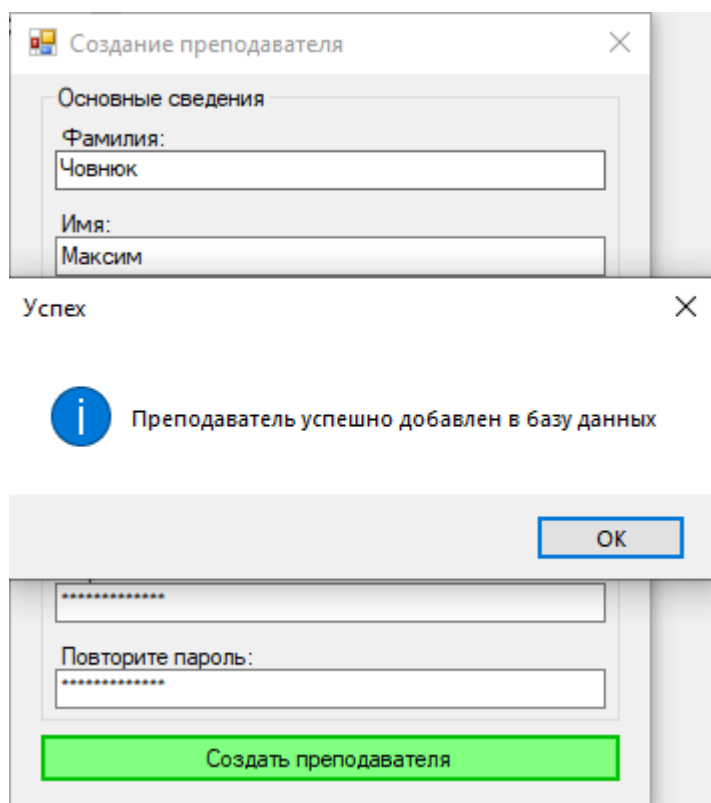
Заполнение полей формы «Создания преподавателя» правильными значениями (охватывает классы 1, 4, 7, 19, 13, 16):

- Фамилия: Човнюк
- Имя: Максим
- Отчество: Андреевич
- Логин: chovnuk\_m\_a
- Пароль: superpassword
- Повторите пароль: superpassword

Результаты: При нажатии кнопки «Создать преподавателя» введенные данные образуют нового преподавателя в базе данных и будет выведено



сообщение об успехе, а также новый преподаватель появится в списке преподавателей.



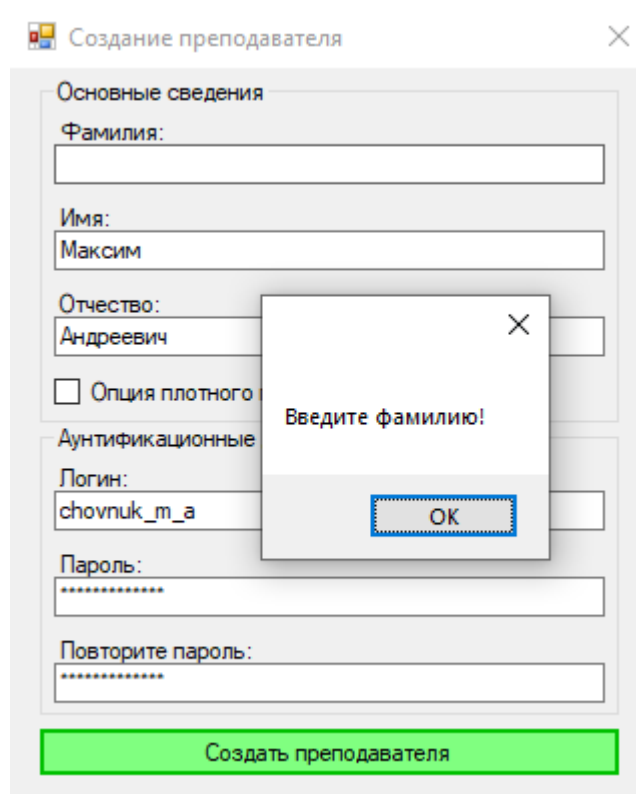
### Тесты для неправильных классов эквивалентности

Для формы «Создание преподавателя» поле «Фамилия» осталось незаполненным (класс 2):

- Фамилия:
- Имя: Максим
- Отчество: Андреевич
- Логин: chovnuk\_m\_a
- Пароль: superpassword
- Повторите пароль: superpassword

Результаты: При нажатии на кнопку «Создать преподавателя» выведется сообщение об ошибке «Введите фамилию!».

При закрытии сообщения все введенные данные сохраняются, при закрытии формы «Создание преподавателя» все введенные данные удаляются.



Для формы «Создание преподавателя» поле «Имя» осталось незаполненным (класс 5):

- Фамилия: Човнюк
- Имя:
- Отчество: Андреевич
- Логин: chovnik\_m\_a
- Пароль: superpassword
- Повторите пароль: superpassword

Результаты: При нажатии на кнопку «Создать преподавателя» выведется сообщение об ошибке «Введите имя!».

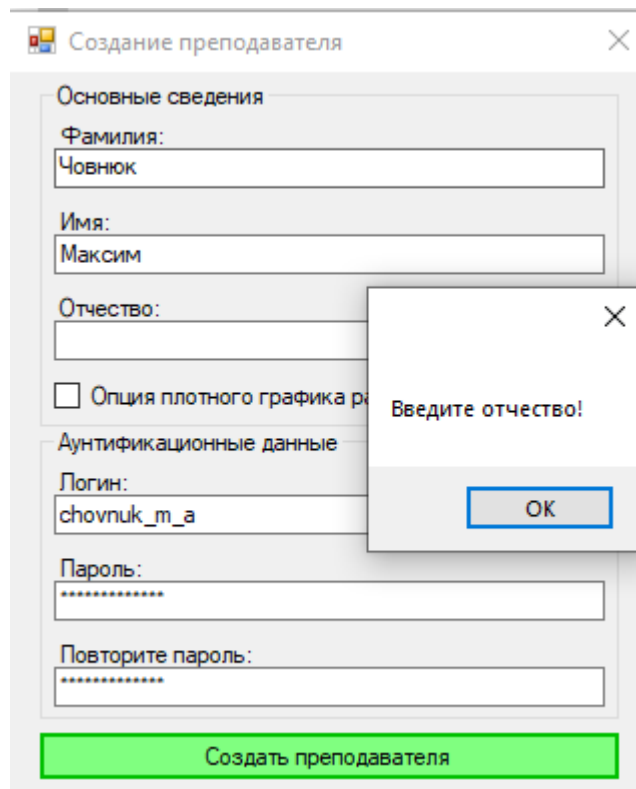
При закрытии сообщения все введенные данные сохраняются, при закрытии формы «Создание преподавателя» все введенные данные удаляются.

Для формы «Создание преподавателя» поле «Отчество» осталось пустым (класс 8):

- Фамилия: Човнюк
- Имя: Максим
- Отчество:
- Логин: chovnik\_m\_a
- Пароль: superpassword
- Повторите пароль: superpassword

Результаты: При нажатии на кнопку «Создать преподавателя» выведется сообщение об ошибке «Введите отчество!».

При закрытии сообщения все введенные данные сохраняются, при закрытии формы «Создание преподавателя» все введенные данные удаляются.

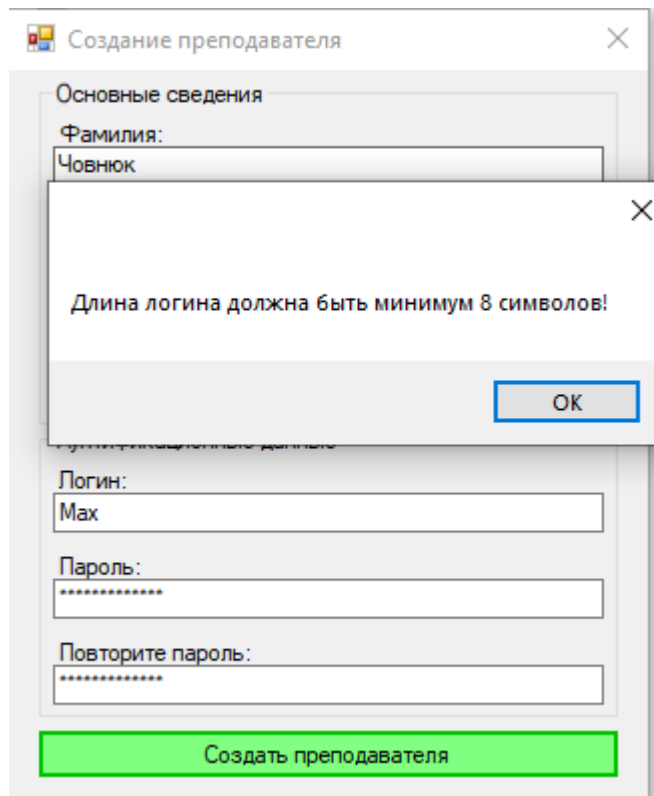


Для формы «Создание преподавателя» значение поля «Логин» имеет менее 8 символов (класс 11):

- Фамилия: Човнюк
- Имя: Максим
- Отчество: Андреевич
- Логин: Мах
- Пароль: superpassword
- Повторите пароль: superpassword

Результаты: При нажатии на кнопку «Создать преподавателя» выведется сообщение об ошибке «Длина логина должна быть минимум 8 символов!».

При закрытии сообщения все введенные данные сохраняются, при закрытии формы «Создание преподавателя» все введенные данные удаляются.

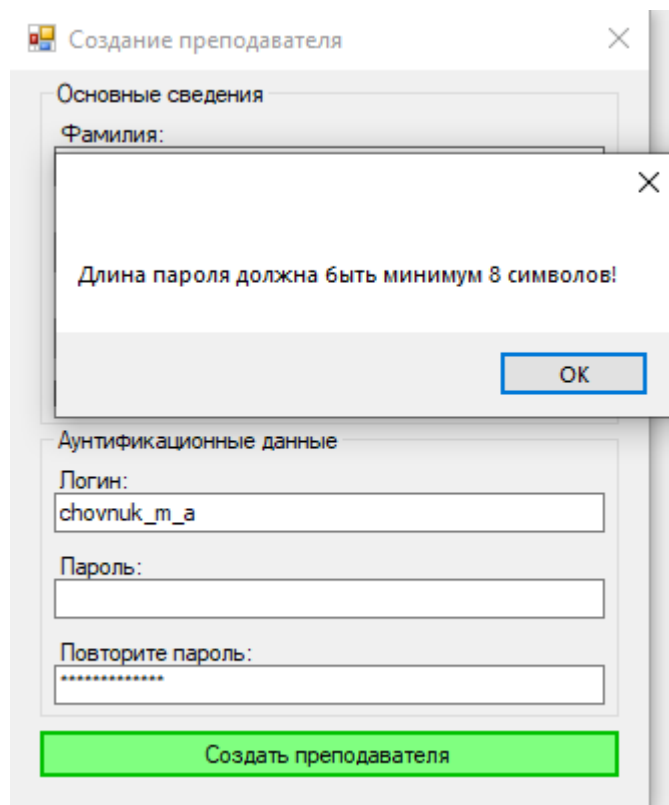


Для формы «Создание преподавателя» поле «Пароль» осталось пустым, т.е. введено менее 8 символов (класс 14):

- Фамилия: Човнюк
- Имя: Максим
- Отчество: Андреевич
- Логин: chovnuk\_m\_a
- Пароль:
- Повторите пароль: superpassword

Результаты: При нажатии на кнопку «Создать преподавателя» выведется сообщение об ошибке «Длина пароля должна быть минимум 8 символов!».

При закрытии сообщения все введенные данные сохраняются, при закрытии формы «Создание преподавателя» все введенные данные удаляются.

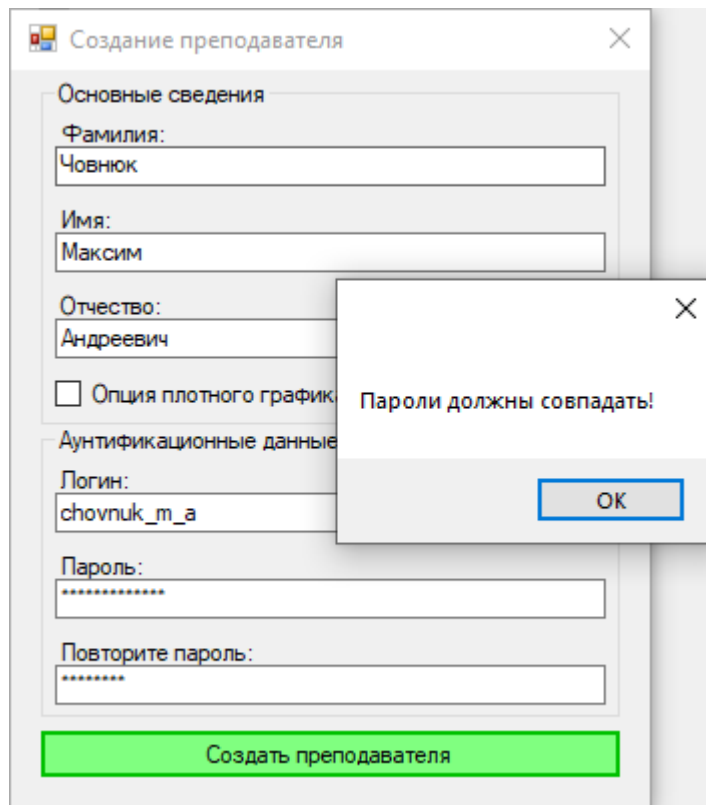


Для формы «Создание преподавателя» поле «Повторите пароль» не совпадает с полем «Пароль» (класс 17):

- Фамилия: Човнюк
- Имя: Максим
- Отчество: Андреевич
- Логин: chovnik\_m\_a
- Пароль: superpassword
- Повторите пароль: password

Результаты: При нажатии на кнопку «Создать преподавателя» выведется сообщение об ошибке «Пароли должны совпадать!».

При закрытии сообщения все введенные данные сохраняются, при закрытии формы «Создание преподавателя» все введенные данные удаляются.



## ПРАКТИЧЕСКАЯ ЧАСТЬ

Исходный код проекта был разделен на логические части – на формы и пользовательские элементы управления. Они в свою очередь были сгруппированы в каталоги по разделу интерфейса, к которому были разработаны (Рис. 92). Скомпилированные файлы проекта весят менее 52 Мбайт, допускается перемещение каталога программы. Для работы программы требуется постоянный доступ к серверу базы данных, .NET Framework 4.7.2,

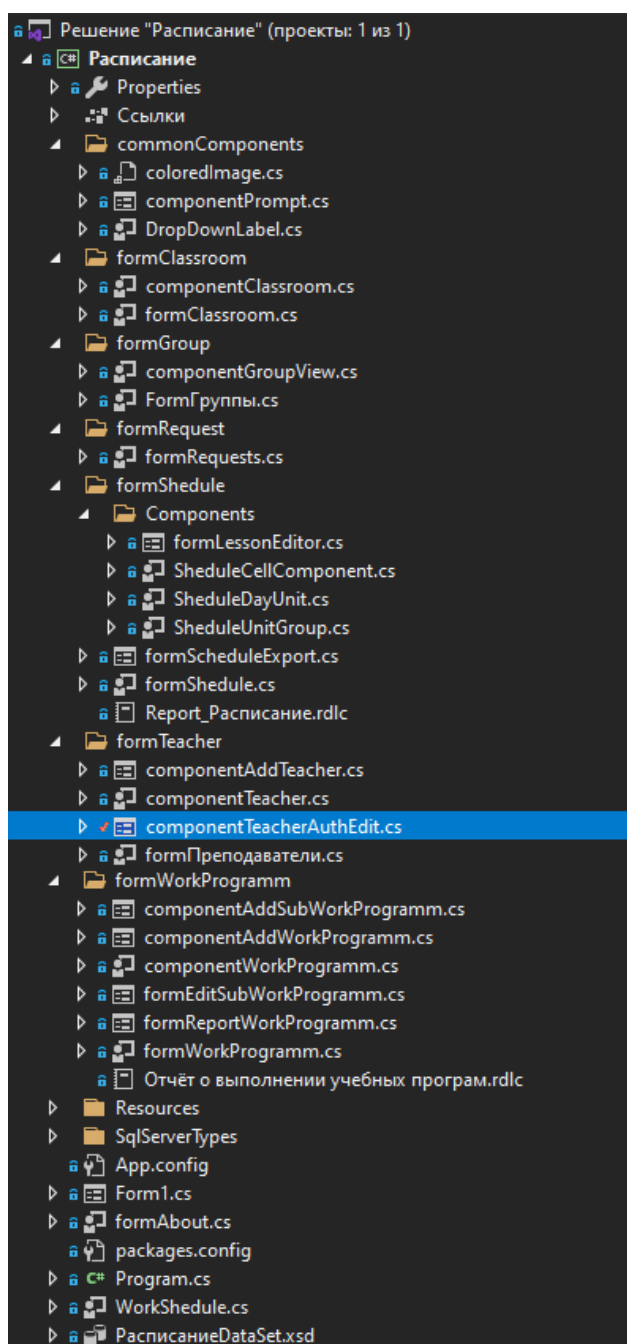


Рисунок 92. Структура файлов в проекте.



## ЗАКЛЮЧЕНИЕ

В ходе курсовой работы удалось реализовать только лишь приложение для руководителя очным отделением. В этом приложении удалось реализовать полноценный интерфейс для управления преподавателями (добавление, редактирование и удаление преподавателей), группами (добавление, редактирование, удаление и перевод групп на следующий курс), аудиториями (добавление, редактирование и удаление аудиторий). Удалось реализовать удобный и практичный интерфейс для управления рабочими программами и дисциплинами: добавлять, редактировать, удалять, просматривать неточности между рабочей программой и дисциплинами, приписанными к рабочей программе, и формировать отчет о выполнении учебного плана. Также реализован редактор графика работы преподавателей и графика учёбы групп. Также реализована интерфейс для редактирования условий проведения дисциплин в аудиториях, причем в 2 версиях: в первой выбираются аудитории, в которых позволено проводить некоторую дисциплину, а во второй выбираются дисциплины, которым позволено проводить занятия в некоторой аудитории.

В конце концов был реализован редактор расписания, в котором можно добавлять, редактировать и удалять занятия. При добавлении и редактировании занятия предусмотрено управление ограничениями, т.е. можно, например, отключить проверку соответствия дисциплины и аудиторий. Также реализован экспорт расписания в удобной, привычной и выравненной по времени таблице, которую можно распечатать на бумагу или сохранить в виде файлов Excel, Word, PDF.

Остается реализовать веб-приложение, в котором можно в удобном графическом режиме просматривать расписание студентами и преподавателями, где преподаватели смогут формировать заявки для корректировки расписания, а директор самостоятельно формировать отчет о выполнении учебного плана в пару кликов.

## СПИСОК ИСТОЧНИКОВ И ЛИТЕРАТУРЫ

1. Основы проектирование баз данных : учебник для студ. учреждений сред. проф. образования / Г.Н. Федорова. – 2-е изд., стер. – М. : Издательский центр «Академия», 2018. 224 с.
2. Базы данных. Разработка клиентских приложений на платформе Net : учебник / Н.Н. Гриченко, А.Ю. Громов, А.В. Благодаров, . – М.: КУРС, 2018. - 288 с.
3. Аганина, Д. А. Проблемы автоматизированного расписания образовательного процесса / Д. А. Аганина. — Текст : непосредственный // Молодой ученый. — 2018. — № 42 (228). — С. 42-43. — URL: <https://moluch.ru/archive/228/53149/> (дата обращения: 16.12.2020).