ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ ЯНАО

ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ЯМАЛО-НЕНЕЦКОГО АВТОНОМНОГО ОКРУГА

«ЯМАЛЬСКИЙ МНОГОПРОФИЛЬНЫЙ КОЛЛЕДЖ»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

**ИНФОРМАЦИОННАЯ СИСТЕМА «РАСПИСАНИЕ ЗАНЯТИЙ» ДЛЯ ГБПОУ ЯНАО «****ЯМАЛЬСКИЙ МНОГОПРОФИЛЬНЫЙ КОЛЛЕДЖ»**

Выполнила:

студентка 21ИСиП группы

специальности 09.02.07

«Информационные системы

и программирование» Куйбина Е.В.

Руководитель:

преподаватель высшей

квалификационной категорииРыбин Ю.И.

(дата,подпись)

Защита состоялась

на заседании методического

объединения педагогических специальностей

(оценка,дата,подпись)

Салехард 2025

Оглавление

[ВВЕДЕНИЕ 3](#__RefHeading___Toc8300_1411721020)

[ГЛАВА I. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ 6](#__RefHeading___Toc8302_1411721020)

[1.1. Теоретические основы построения информационной системы 6](#__RefHeading___Toc8304_1411721020)

[1.2. Методы разработки АИС 8](#__RefHeading___Toc8306_1411721020)

[1.2. Описание предметной области 10](#__RefHeading___Toc8308_1411721020)

[1.3. Проектирование и принципы создания АИС 12](#__RefHeading___Toc8310_1411721020)

[1.5 . Этапы создания АИС 12](#__RefHeading___Toc8312_1411721020)

[1.6 . Внешний интерфейс 14](#__RefHeading___Toc8314_1411721020)

[1.7. Обзор и анализ существующих разработок, выбор технологии проектирования 14](#__RefHeading___Toc8338_1411721020)

[1.8. IDE и редакторы кода для разработчиков 16](#__RefHeading___Toc3096_3321077987)

[ГЛАВА II. Эскизный проект 20](#__RefHeading___Toc8316_1411721020)

[2.1 . Разработка информационной системы 20](#__RefHeading___Toc8318_1411721020)

[2.2 . Определение состава данных 23](#__RefHeading___Toc8320_1411721020)

[2.3 . Создание концептуальной модели 25](#__RefHeading___Toc8322_1411721020)

[2.5. Выявление информационных объектов 28](#__RefHeading___Toc8324_1411721020)

[2.6. Определение связей информационных объектов, построение ER-диаграммы 31](#__RefHeading___Toc8326_1411721020)

[2.7. Определение логической и физической моделей базы данных 33](#__RefHeading___Toc8328_1411721020)

[ЗАКЛЮЧЕНИЕ 35](#__RefHeading___Toc8330_1411721020)

[СПИСОК ЛИТЕРАТУРЫ 36](#__RefHeading___Toc8332_1411721020)

[ПРИЛОЖЕНИЕ А 39](#__RefHeading___Toc8334_1411721020)

[3.1. Техническое задание 39](#__RefHeading___Toc8336_1411721020)

# ВВЕДЕНИЕ

В настоящее время информация становится фактором, определяющим эффективность любой сферы деятельности. Существуют много разных отраслей производства, образовательных систем и государственных структур, где необходимо обрабатывать огромное количество данных. Большинство операций в настоящее время не могут быть выполнены вручную, поскольку для их выполнения необходимо использование электронно-вычислительных машин, в которых важную роль играют системы хранения и обработки информации.

Современная образовательная система требует оптимизации процесса обучения в колледже, наиболее актуальными становятся инновации в образовании. В этой связи особенно важно организовать оптимальное учебное расписание для эффективного планирования образовательного процесса. У нас в колледже нет единой системы составления расписания. Расписание составляется диспетчером вручную, заполняя таблицу в MC Excel, данный процесс является трудоемким. Для этого необходимо автоматизировать работу, чтобы облегчить не только сотрудникам должностные обязанности в выполнении монотонных действий, но и в планировании времени студентов. Уже существуют такие информационные и автоматизированные системы, которые направлены для преодоления сложностей, однако найти подходящую информационную систему очень сложно.

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

Поэтому актуальным является рассмотрение темы создания информационной системы при структурном анализе к программированию для информационной и программной совместимости по предметной области «Расписание занятий»

Цель исследования: создание информационной системы «Расписание занятий» для ГБПОУ ЯНАО «Ямальский многопрофильный колледж»

Задачи:

1. Изучить предметную область разрабатываемого проекта информационной системы «Расписание занятий»;
2. Проведение анализа понятия «информационная система» и технических и программных средств для ее создания;
3. Разработка и описание основных элементов модели информационной системы «Расписание занятий»
4. Составить техническое задание
5. Определить и создать состав данных изучаемой области
6. Создать UML-диаграммы последовательности, концептуальную и логическую модель
7. Изучение ресурсов для создания внешнего интерфейса программного обеспечения
8. Разработать базу данных (представить таблицы и схемы БД)
9. Обосновать выбор языка программирования используемый в создании продукта
10. Описать интегрированную среду разработки (IDE) для данного языка программирования

Объект исследования: информационная система

Предмет исследования: информационная система «Расписание занятий»

Методы исследования: моделирование, анализ,описание

Практическая значимость дипломного проекта состоит в создании информационной системы, которая позволит использовать современные цифровые технологии для работы с расписанием в деятельности учебного заведения, это позволит улучшить процесс планирования, доступность и обмен информацией, сократить затраты времени и ресурсов, а также повысить прозрачность и гибкость в управлении.

# ГЛАВА I. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

## Теоретические основы построения информационной системы

Информация — это некоторые сведения, знания об объектах и процессах реального мира.

Система — это комплекс взаимосвязанных средств, выступающих как единое целое. Каждая система характеризуется структурой, входными и выходными потоками, целью и ограничениями, законом функционирования.

Система охватывает комплекс взаимосвязанных элементов, действующих как единое целое в достижении поставленных целей.

Каждая система включает в себя компоненты:

1. Структура системы – это множество элементов системы и взаимосвязей между ними.
2. Функции каждого элемента системы.
3. Вход и выход каждого элемента и системы в целом.
4. Цели и ограничения системы и ее отдельных элементов.

База данных (БД) — совокупность взаимосвязанных данных, организованных по определенным правилам, предусматривающим общие принципы описания, хранения и обработки данных.

Информационной системой — называется взаимосвязанная совокупность средств, методов и персонала, используемых для хранения, обработки и выдачи информации в интересах достижения поставленной цели. Информационные системы обеспечивают сбор, хранение, обработку, поиск, выдачу информации, необходимой в процессе принятия решений задач из любой области. Они помогают анализировать проблемы и создавать новые продукты. В качестве основного технического средства переработки информации используют персональный компьютер (ПК).

Персональный компьютер (ПК) — это микро-ЭВМ (электронно-вычислительная машина), которая имеет эксплуатационные характеристики бытового прибора и широкий диапазон универсальных функциональных характеристик. Термин ПК на сегодняшний день означает персональный компьютер, который предназначается для индивидуальной работы и пользования.

Особую роль в информационных системах отводится человеку, т. к. техническое воплощение информационной системы само по себе ничего не будет значить, если не учтена роль человека, для которого предназначена производимая информация и без которого невозможно ее получение и представление.

Автоматизированная информационная система (АИС) — совокупность программно-аппаратных средств, предназначенных для автоматизации деятельности, связанной с хранением, передачей и обработкой информации.

АИС являются, с одной стороны, разновидностью информационных систем (ИС), с другой — автоматизированных систем (АС), вследствие чего их часто называют ИС или АС.

Цель ИС: информационное обслуживание или обеспечение основной деятельности системы информационного обмена, а также повышение производительности и эффективности труда, улучшение качества информационной продукции и услуг, повышение сервиса и оперативности обслуживания.

Введение ИС в предметную область (расписание занятий) положительно скажется на эффективности его работы:

1. Удобство и эффективность организации расписания занятий. В учебных учреждениях важно иметь структурированное и обновляемое расписание. Это позволяет избежать путаницы, конфликтов и неопределенности при назначении и проведении занятий.

2. Увеличение эффективности использования ресурсов. С использованием информационной системы "Расписание занятий" можно оптимизировать распределение учебных аудиторий, преподавательского состава и других ресурсов. Это позволяет повысить эффективность использования учебных и рабочих мест, сократить время и затраты на организацию занятий.

3. Улучшение доступности информации. ИС позволяет предоставить участникам образовательного процесса или сотрудникам учреждений доступ к актуальной информации о расписании занятий. Это сокращает потребность в поиске.

4. Автоматизация рутинных задач. Использование ИС позволяет автоматизировать процессы составления расписания занятий. Это позволяет значительно сократить время и усилия, затрачиваемые на поддержание актуальности расписания, а также снизить вероятность ошибок при его создании и обновлении.

5. Соответствие современным требованиям. В современном информационном обществе информационные системы становятся неотъемлемой частью повседневной жизни. ИС позволяет соответствовать требованиям современного мира, где автоматизация, эффективность и доступность информации являются важными компонентами.

## 1.2. Методы разработки АИС

Разработка и проектирование АИС начинается с создания концептуальной модели использования системы. Прежде всего должна быть определена целесообразность создания системы, ее конкретные функции и подлежащие автоматизации задачи. Должна быть выполнена оценка не только целей, но и возможностей создания системы. Далее проводится анализ требований к АИС, детальное проектирование, взаимосвязь этапов, программирование и тестирование, минимизация потерь при переходе от одного уровня представления информации к другому, интеграция в существующую систему, внедрение и поддержка.

Существует три класса методологий проектирования АИС:

* концептуальное моделирование предметной области;
* выявление требований и спецификация информационной системы через ее макетирование;
* системная архитектура программных средств, поддерживаемая инструментальными средствами CASE-технологии (CASE — Computer Aided Software Engineering — технология создания и сопровождения ПО различных систем).

В базе данных отображается какая-то часть реального мира.

Естественно, что полнота ее описания будет зависеть от целей создаваемой информационной системы. Как указано выше, часть реального мира, представляющая интерес для данного исследования, называется предметной областью. Для того чтобы база данных адекватно отражала предметную область, проектировщик должен хорошо представлять себе все нюансы, присущие ей, и уметь отобразить их в базе данных.

Предметная область должна быть предварительно описана. Формализованное описание предметной области будем называть ее концептуальной моделью. Предметные области могут быть различными, и для их моделирования могут потребоваться специфические средства, соответствующие особенностям этих областей.

## Описание предметной области

Перед началом учебного периода в учебный отдел поступают рабочие программы от заместителя директора, ответственного за научно-методическую и исследовательскую работу. В программе указываются следующие данные:

• Название рабочей программы предмета, которое будет отображаться в расписании;

• Для какой группы/подгруппы предназначена программа;

• Количество аудиторных часов, которые необходимо провести для группы;

• Количество самостоятельных работ, которые необходимо выдать группе;

• Преподаватели, которые должны вести этот предмет и для какой группы\подгруппы;

• Сроки, к которым необходимо провести все академические часы (за год, за семестр, и т. д.);

• Дополнительные требования для аудиторий.

На основе этих данных необходимо составлять расписание (на неделю, на день) с учетом следующих требований:

• Все рабочие программы должны быть выданы в оговоренные сроки;

• У преподавателей не должно быть одновременно идущих пар;

• В одной аудитории не должно быть нескольких групп, и аудитория должна быть подготовлена для проведения занятий;

• Пары должны быть назначены преподавателю и группе в пределах их рабочего расписания;

• Также необходимо учитывать, что аудиторный час может быть выдан не группе, а подгруппе.

Длительность академического часа считается глобальным параметром, редко изменяющимся.

Каждый преподаватель может попросить учебный отдел не назначать занятий в определенный промежуток времени из-за форс-мажорных обстоятельств или изменить рабочее расписание.

Во время учебного процесса директор учебного заведения может запросить отчет о степени выдачи рабочих программ.

Также необходимо выдавать практику для студентов во время образования. Место практики не может быть определено заранее, однако заранее определяется только неделя, в которую необходимо выдать практику. Практика может быть учебной или производственной. Часть практик может занимать всю неделю или день, а другая часть может быть разбросана по дням и парам. Руководитель практики назначает практику и передает эту информацию в учебный отдел.

Расписание учебных занятий составляет диспетчер, подписывает заместитель директора по учебно-методической работе. Преподаватели, сформировав рабочую программу, заместителю учебной части для утверждения. Заместитель может вернуть преподавателю рабочую программу для доработки. Преподаватели ввиду форс-мажоров могут написать заявку для корректировки своего расписания. По запросу заведующего учебной частью формируется отчет выполнения учебных программ, в котором отражается соотношение количества выданных академических часов к общему числу, требуемых к выдаче.

Диспетчер должен вводить расписание в виде записей базы:

* День недели
* Время начала
* Аудитория
* Группа
* Дисциплина
* Преподаватель

Кроме того, информационная система должна выводить информацию по запросам расписаний:

* группы;
* преподаватели;
* аудитории.

## Проектирование и принципы создания АИС

Проектирование АИС - процесс создания и внедрения проектов комплексного решения экономических задач по новой технологии. В каждом подразделении организации должен быть назначен сотрудник, ответственный за проектирование и внедрение АИС и, который собирает нужную информацию, подбирает технику и программные средства, ведет обучение персонала, руководит внедрением и анализом функционирования ИС.

## . Этапы создания АИС

В основе создания и использования АИС лежит понятие жизненного цикла (ЖЦ).

Жизненный цикл является моделью создания и использования АИС, которая отражает различные состояния системы с момента возникновения в данном комплексе средств до момента его полного выхода из употребления.

Для АИС условно выделяют следующие основные этапы их жизненного цикла:

1. анализ — определение того, что должна делать система;
2. проектирование — определение того, как система будет функционировать: прежде всего спецификация подсистем, функциональных компонентов и способов их взаимодействия в системе;
3. разработку — создание функциональных компонентов и отдельных подсистем, соединение подсистем в единое целое;
4. тестирование — проверку функционального и параметрического соответствия системы показателям, определенным на этапе анализа;
5. внедрение — установку и ввод системы в действие;
6. сопровождение — обеспечение штатного процесса эксплуатации системы на предприятии заказчика.

Существующие модели жизненного цикла определяют порядок исполнения этапов в процессе создания системы, а также критерии перехода от этапа к этапу. Наибольшее распространение получили три следующие модели:

* каскадная модель (предполагает переход на следующий этап после полного завершения работ предыдущего этапа);
* поэтапная итерационная модель (предполагает наличие циклов обратной связи между этапами);
* спиральная модель (соответствует поэтапной модели создания фрагмента или версии системы, на нем уточняются цели и характеристики проекта, определяется его качество, планируются работы следующего этапы);

## . Внешний интерфейс

Интерфейс — это внешний вид продукта: кнопки, формы, иконки. Их можно создавать прямо в фигме и добавлять эффекты: создавать анимацию для баннеров, делать кнопки кликабельными.

Фигма (Figma) — графический редактор для создания прототипов сайтов и приложений. Я выбрала именно его, потому что считаю его самым оптимальным вариантом. Он удобен в работе.

Figma можно пользоваться бесплатно, но с ограничениями для командной работы:

* Ваши личные файлы можете редактировать только вы. При этом личных файлов может быть сколько угодно;

Для командной работы доступны только три файла и один проект.

## 1.7. Обзор и анализ существующих разработок, выбор технологии проектирования

Обзор и анализ существующих разработок и выбор технологии проектирования являются важными этапами в процессе разработки информационных систем (ИС). Они помогают определить наилучший подход к проектированию и выбрать соответствующие технологии для реализации системы. Для того, чтобы приступить к выбору существующих разработок определимся и выберем среду разработки.

Краткий обзор некоторых популярных средств разработки:

– Java является одним из самых популярных языков программирования и платформой разработки. Он широко используется для создания масштабируемых и надежных приложений, особенно в предметно-ориентированном программировании и разработке веб-приложений. К достоинствам можно отнести объектную ориентированность и архитектурную независимость. К недостаткам: повышенные требования к объему оперативной памяти, относительно невысокое быстродействие.

– Python – это высокоуровневый язык программирования, который известен своей простотой и читаемостью. Он широко используется для разработки веб-приложений, научных вычислений, анализа данных и

автоматизации задач. К достоинствам можно отнести принципы ООП,

относительная компактность и простота кода. Минусы: относительно

невысокая скорость выполнения программ, ограниченность средств для работы с базами данными.

– C# – это язык программирования, разработанный Microsoft, который

используется для создания приложений на платформе .NET. Он

обеспечивает широкие возможности для разработки различных типов

приложений, включая веб-приложения, настольные приложения и

игры. К достоинствам можно отнести архитектурную независимость,управление памятью в автоматическом режиме, строгая типизация. К недостаткам можно отнести невысокую скорость, проблемы безопасности.

– PHP – это язык программирования, который специально разработан для веб-разработки. Он широко используется для создания динамических веб-сайтов и веб-приложений, и он интегрируется с HTML. К достоинствам можно отнести принципы ОПП, удобство и гибкость программирования, совместимость с большим количеством СУБД, кроссплатформенность. К минусам можно отнести его открытость, что снижает безопасность, низкая модульность приводит к сложностям при разработке больших приложений.

– язык программирования 1С – это встроенный язык платформы 1С

Предприятие, представляющий инфраструктуру и инструменты для быстрой разработки бизнес-приложений. Разработка осуществляется по методологии объектно-ориентированного программирования в концептуальной модели (с некоторыми ограничениями) и процедурном стиле. К плюсам можно отнести самодостаточную платформу, российское происхождение и присутствие в реестре отечественного ПО, гибкость и адаптированность под наиболее известные СУБД и

операционные системы. К недостаткам можно отнести ресурсоемкость, закрытость интерфейса к изменениям.

Каждый из этих инструментов имеет свои особенности, преимущества и области применения. Выбор наиболее подходящего средства разработки зависит от требований проекта, целевой платформы, опыта команды разработчиков и других факторов.

Разработка на языке программирования Python имеет свои особенности и может быть предпочтительной в определенных ситуациях. Вот несколько аргументов, которые могут обосновывать мой выбор разработки на Python:

* его достоинства: простота синтаксиса и визуального восприятия кода, интерпретируемость, объектно ориентированность и динамическая типизация;
* изучение этого языка в образовательной программе натолкнули меня применить мои знания на практике в создании информационной системы;
* подходит к требованиям для разработки данного проекта

## **1.8. IDE и редакторы кода для разработчиков**

IDE (Integrated Development Environment) и редактор кода – это виды программного обеспечения, предназначенные для работы над приложениями, их разработки и тестирования. Возможности этих инструментов отличаются.

Редактор кода является программой, работающей как текстовый редактор, только более подходящий для написания кода.

IDE – это ПО, которое объединяет инструменты для разработки приложений и их тестирования в едином интерфейсе. Может делать все то, что и текстовый редактор, и даже больше. В принципе этим IDE и отличается от него, она предоставляет более расширенные функции. Среды разработки дают возможность создавать крупные проекты, а также подключать Git. Как правило, IDE состоит из:

* самого текстового редактора для написания и редактирования кода;
* компилятора – инструмента, позволяющего перевести текст, написанный на языке программирования, в набор машинных кодов;
* отладчика, проверяющего код и устраняющего в нем ошибки;
* инструментов для автоматизации сборки кода, ускоряющих процесс разработки.

Многие IDE позволяют настраивать дополнительные фреймворки и библиотеки для расширения возможностей разработки.

При выборе подходящей среды разработчики ориентируются на различные факторы, поддерживает ли IDE необходимые языки программирования, есть ли возможность совместной разработки для командной работы, какие дополнительные функции доступны, можно ли устанавливать дополнительные расширения, нужна ли подписка или можно обойтись бесплатной версией. Ниже привела описание и основные преимущества и недостатки некоторых инструментов, чтобы можно было понять, какой инструмент подходит больше под мои задачи.

PyCharm

PyCharm – среда для работы с Python. Разработчики из компании JetBrains создали серию IDE с ориентированием на разные языки: PyCharm, PHPStorm, CLion, IntelliJ IDEA. К сожалению, компания-разработчик этого и другого ПО, ушли с российского рынка, и оплата платных подписок сейчас недоступна. Однако PyCharm обладает бесплатной опенсорсной версией – Community, которая предоставляет инструменты для программирования на Python.

Преимущества:

* автозавершение кода, автопоиск и исправление ошибок в коде;
* поддержка сред для веб-разработки Django, Flask, Google App Engine, Pyramid и web2py;
* интеграция с системами контроля версий;
* интеграция с Jupyter Notebook, Anaconda и другими библиотеками;
* отладчик, профилировщик Python, встроенные терминал и инструменты баз данных.

Недостатки:

* версия Community не поддерживает многие инструменты, нет профилировщика Python и поддержки баз данных;
* отсутствует возможность оплаты Professional версии в России.

IntelliJ IDEA

IDE от JetBrains – IntelliJ IDEA. Изначально была разработана для работы с языком Java. Сегодня же она позволяет работать с кодом и на других языках, которые можно добавить с помощью расширений: HTML, CSS, PHP, Python, SQL и другие.

Преимущества:

* автозавершение и исправление кода;
* мультиязычность;
* удобная рабочая среда, которая интуитивно понимает ваши действия и автоматически вызывает нужные инструменты;
* интеграция с системами контроля версий;
* анализ кода в реальном времени и надежные рефакторинги.

Недостатки:

* отсутствие в Community-версии инструментов профилирования, баз данных;
* отсутствует возможность удаленной разработки;
* сложная платформа для начинающих разработчиков.

Visual Studio Code

Visual Studio Code – это бесплатный редактор кода, который включает в себя инструменты запуска и отладки, интеграцию с системой контроля версий и дает возможность писать код на многих языках программирования: C++, CSS, JavaScript, JSON, PHP, Python и другие.

Преимущества:

• кроссплатформенность: поддержка на Windows, macOS и Linux;

• встроенная технология автодополнения кода IntelliSense;

• множество бесплатных расширений в маркетплейсе;

• полностью бесплатный редактор.

Недостатки:

• в Visual Studio Code доступно меньше инструментов, чем в полноценной IDE от Microsoft;

• отсутствует поддержка split screen.

Анализировав интегрированные среды разработки (IDE) для данного языка программирования Python

# ГЛАВА II. Эскизный проект

## . Разработка информационной системы

Этап проектирования является одним из наиболее важных этапов разработки информационной системы. Данный этап подразумевает «проектирование объектов данных, которые будут реализованы в базе данных; проектирование программ, экранных форм, отчетов, которые будут обеспечивать выполнение запросов к данным; учет конкретной среды или технологии, а именно: топологии сети, конфигурации аппаратных средств, используемой архитектуры (файл-сервер или клиент-сервер), параллельной обработки, распределенной обработки данных и т.п.»

В качестве средства проектирования, определения целей и задач создания информационной системы является, без сомнения, унифицированный язык моделирования – UML. Проанализировав особенности предметной области, для формирования требований к информационной системе «Расписание занятий» создала следующие диаграммы вариантов использования (см. рис. 1.1 и рис. 1.2).

Изображение выглядит как текст, круг, линия, иллюстрация

Автоматически созданное описаниеРис. 1.1 Диаграммы вариантов использования

Изображение выглядит как текст, диаграмма

Автоматически созданное описаниеРис. 1.2 Диаграммы вариантов использования

Для дальнейшего проектирования я предлагаю следующие диаграммы последовательности, которые, на мой взгляд, лучше всего отражают потоки событий в рамках соответствующих вариантов использования.

Диаграмма последовательности действий преподавателя отличается от данной диаграммы только запрашиваемой информацией – для преподавателя важно именно его расписание, а не расписание по какой-либо студенческой группе. Данные пользователи информационной системы будут получать данные с сайта колледжа со страницы, написанной на языке программирования.

Так, диаграмма последовательности действий студента изображена на рис. 2.

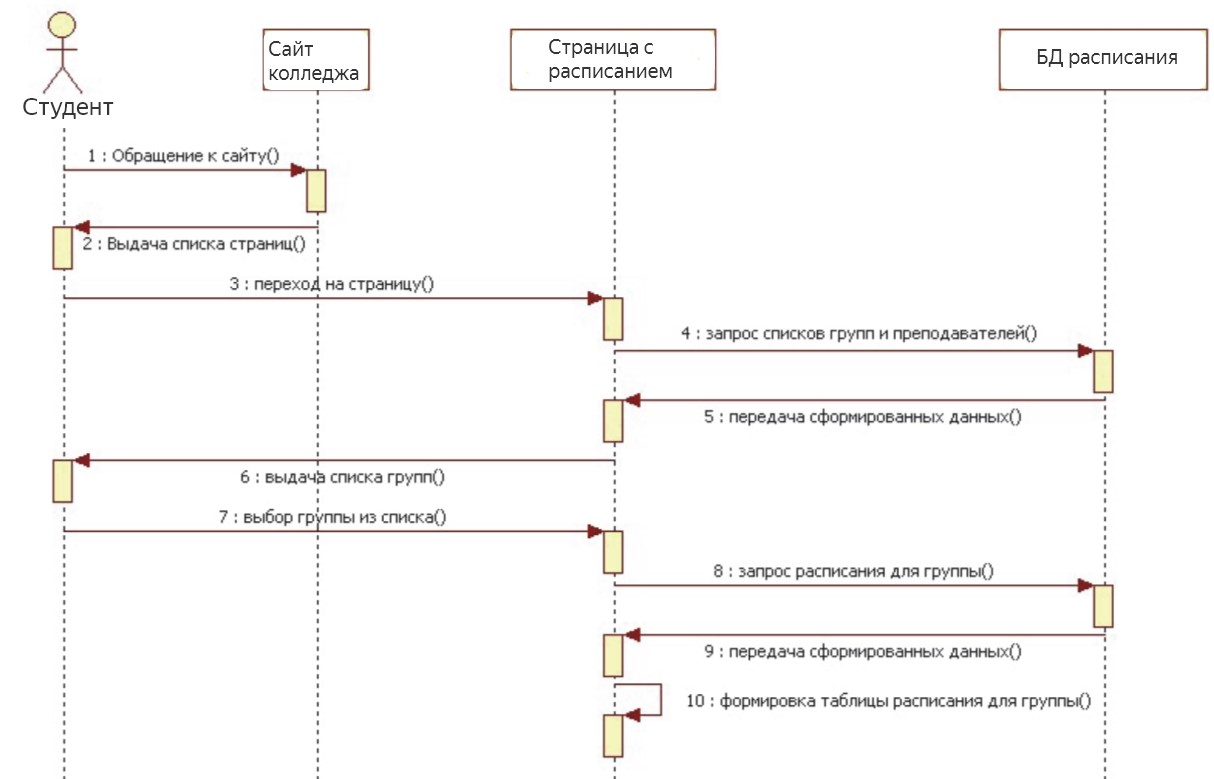
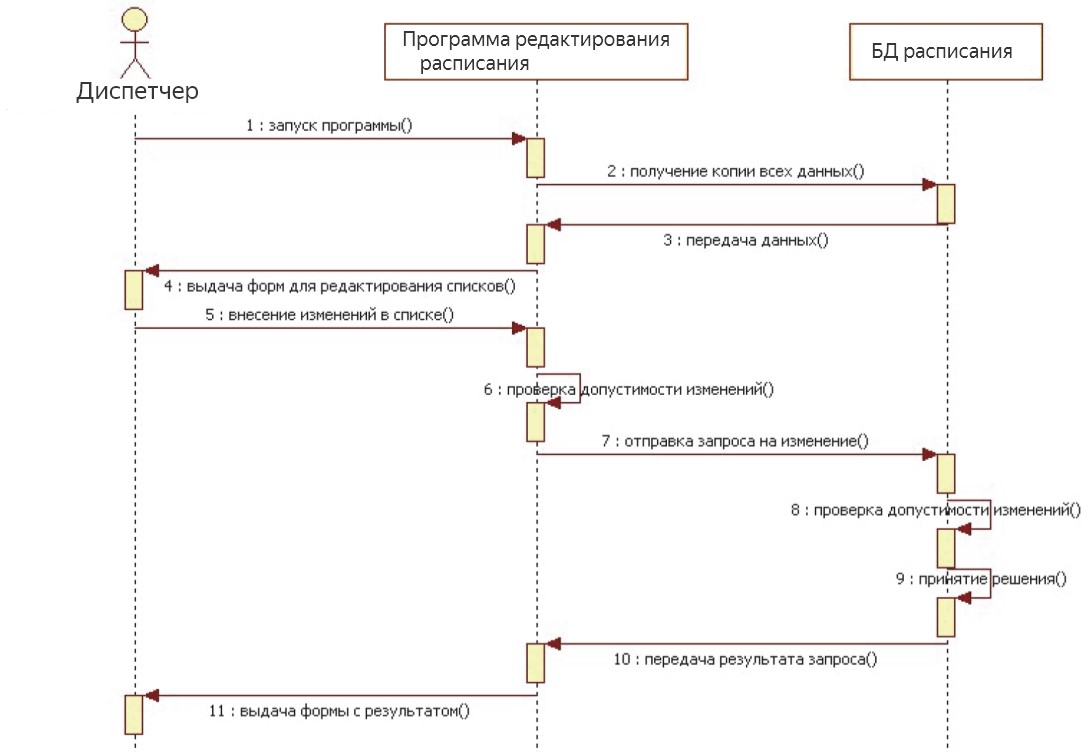
Рис. 2. Диаграмма последовательности действий информационной системы и студента

Диаграмма последовательности действий диспетчера, занимающегося заполнением и редактированием расписания, изображена на рис. 3.

Рис. 3. Диаграмма последовательности действий информационной системы и диспетчера

Эти сотрудники будут работать с единой базой данных расписания колледжа, хранящейся на сервере (серверах) при помощи программы-клиента, написанной на языке программирования. При этом очень важно исключить возможные коллизии одновременного доступа к общей базе данных разными сотрудниками, такие как взаимные блокировки, дедлоки и т. п. Возможные варианты решения — это использование политик безопасности системы управления базами данных или встроенных механизмов выбранного языка программирования. Выбор конкретного решения можно сделать на этапе реализации информационной системы «Расписание занятий» для ГБПОУ ЯНАО «ЯМК».

При изменении расписания важно учитывать тот факт, что множественный ввод некорректных значений в случае проверки только на сервере может увеличить нагрузку на сеть. Однако проверка на соответствие в пределах программы не может гарантировать отсутствие конфликтов в расписании в виду того, что информация, полученная с сервера, устаревает. Как видно из рис. 3, планируется совершать две последовательные проверки.

## . Определение состава данных

Для обеспечения пользователя указанной информацией в БД должны храниться справочные данные о группах, аудиториях и преподавателях. В результате анализа предметной области выявляются документы, содержащие справочную информацию: «Аудитории» (рис.3.1), «Группы» (рис.3.2), «Преподаватели» (рис.3.3), «Расписание звонков» (рис.3.4), «Дисциплины» (рис.3.5), «Дни недели» (рис.3.6), «Расписание» (рис.3.7), «Учебный план» (рис.3.8).Ниже приведены формы документов.

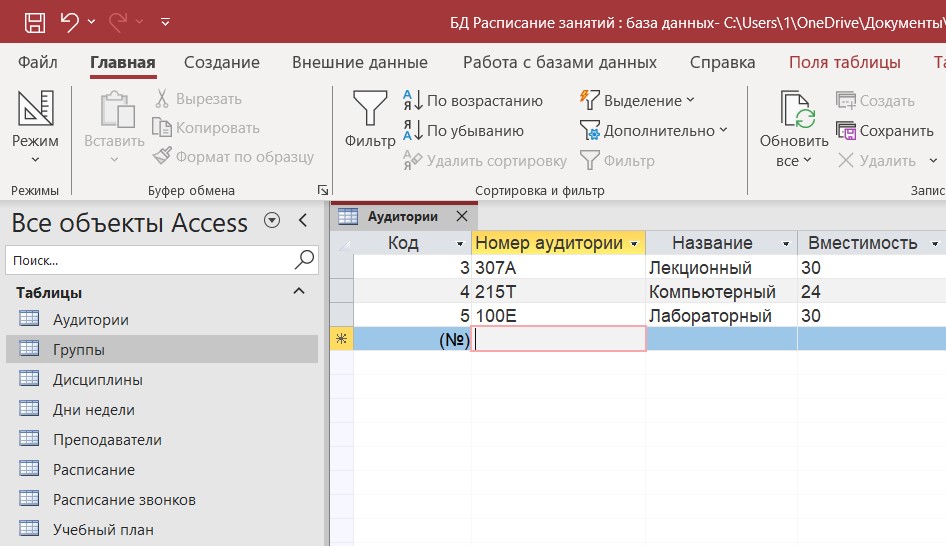


Рис.3.1 Форма справочного документа «Аудитории»



Рис.3.2 Форма справочного документа «Группы»



Рис.3.3 Форма справочного документа «Преподаватели»



Рис.3.4 Форма справочного документа «Расписание звонков»

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рис.3.5 Форма справочного документа «Дисциплины»



Рис.3.6 Форма справочного документа «Дни недели»

Рис.3.7 Форма справочного документа «Расписание»

Рис.3.8 Форма справочного документа «Учебный план»

На основе анализа предметной области можно выявить логические взаимосвязи данных, которые определяют структуру данных предметной области. В результате должны быть выявлены информационные объекты и установлены связи между ними.

## . Создание концептуальной модели

Концептуальная модель является представлением точки зрения пользователя на предметную область и не зависит ни от программного обеспечения СУБД, ни от технических решений.

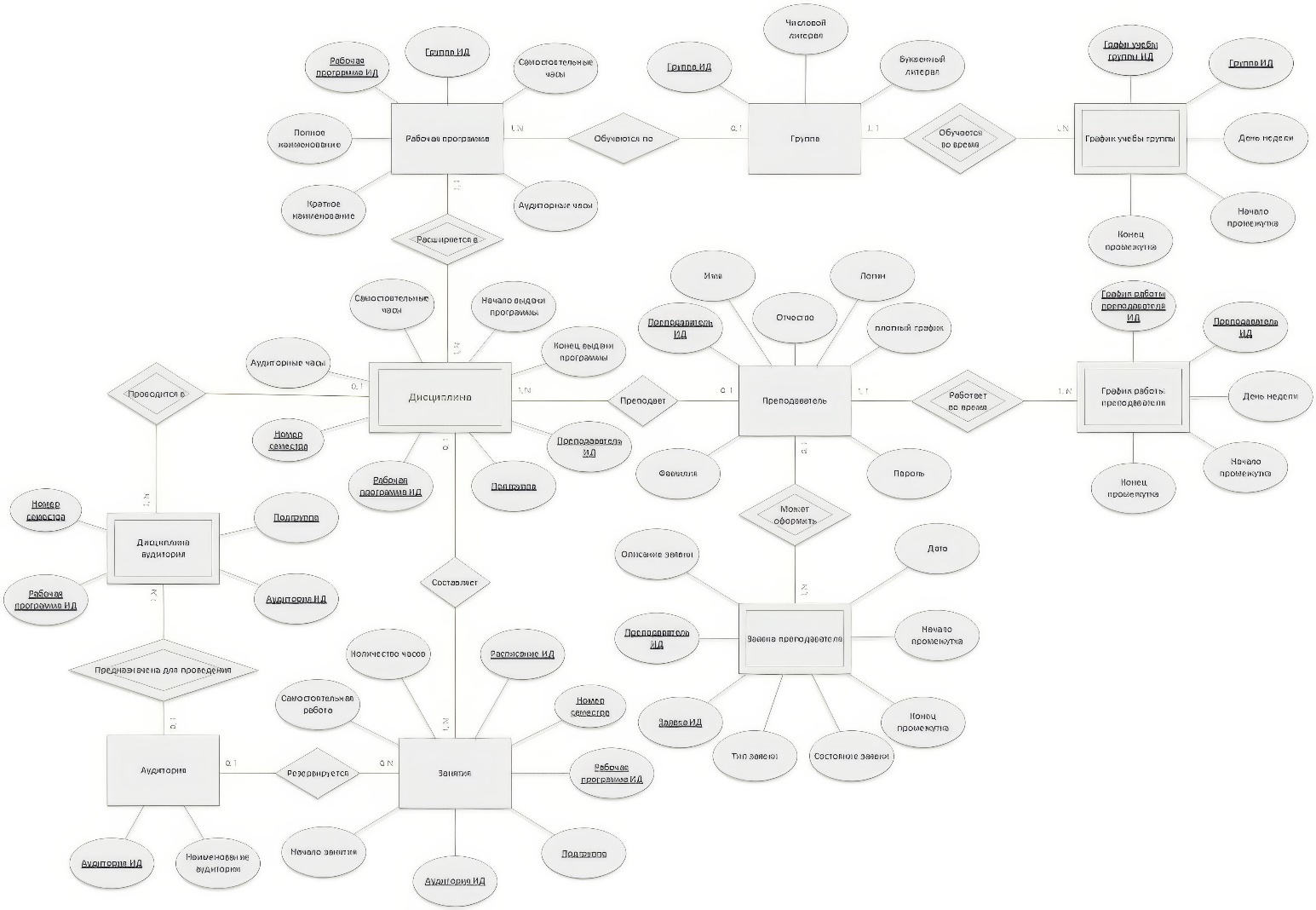
Концептуальная модель должна быть стабильной.

Одной из распространенных моделей концептуальной схемы является модель «сущность - связь». Под сущностью понимают основное содержание объекта предметной области, о котором собирают информацию. В качестве сущности могут выступать место, вещь, личность, явление.

Экземпляр сущности - конкретный объект.

Сущность принято определять атрибутами – поименованными характеристиками.

Мне предстояло создать концептуальную модель моей будущей базы данных. Для этого я сначала для моделирования выбрала нотацию Чена, после приступила к моделированию самой модели (Рис. 4).

Рис. 4 Концептуальная модель базы данных по нотации Чена

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 учебной части. 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 со 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 ввела сущность «Дисциплина аудитория», которая сама по себе является лишь таблицей соответствий, показывающая в каких аудиториях можно проводить занятия некоторой дисциплины.

## 2.5. Выявление информационных объектов

Необходимо проанализировать каждый реквизит на наличие взаимосвязей с другими реквизитами. Реквизит приобретает смысл только тогда, когда он связан с другими реквизитами, обладающими смысловым единством. Так, код техники, наименование техники, количество отражают сведения об изделии; код объекта, наименование объекта, начальник объекта отражают сведения об объектах.

Среди реквизитов, описывающих сущность, можно выделить один или несколько реквизитов, которые однозначно определяют экземпляр сущности. Такой реквизит является ключом. Между ключом и другими реквизитами существует функциональная зависимость.

Реквизиты каждого информационного объекта канонической модели данных должны отвечать требованиям, соответствующим третьей нормальной форме реляционной модели данных:

* информационный объект должен содержать уникальный идентификатор: первичный ключ;
* все описательные реквизиты должны быть взаимно независимы, то есть между ними не должно быть функциональных зависимостей;
* все реквизиты, входящие в составной ключ, также должны быть взаимно независимы;
* каждый описательный реквизит должен функционально полно зависеть от ключа, то есть каждому значению ключа должно соответствовать только одно значение описательного реквизита, а при составном ключе описательные реквизиты должны зависеть целиком от всей совокупности реквизитов, образующих ключ;
* каждый описательный реквизит должен зависеть от ключа не транзитивно, то есть не должен зависеть через другой промежуточный реквизит.

Реквизитный состав первичных документов (см. таблицу 1):

Таблица 1. Объекты справочной информации

|  |  |  |  |
| --- | --- | --- | --- |
| **Информационный**  **объект** | **Наименование реквизита** | **Обозначение** | **Признак ключа** |
| Группы | Код группы | КОД\_ГР | П, У |
| Количество | КОЛ |  |
| Специальность | СПЕЦ |  |

Продолжение таблицы 1.

|  |  |  |  |
| --- | --- | --- | --- |
| **Информационный**  **объект** | **Наименование реквизита** | **Обозначение** | **Признак ключа** |
| Аудитории | Номер аудитории | НОМ\_АУД | П, У |
| Название | НАЗВ |  |
| Вместимость | ВМЕСТ |  |
| Дисциплины | Код дисциплины | КОД\_ДИСЦ | П, У |
| Дисциплина | ДИСЦ |  |
| Дни недели | Код дня недели | КОД\_НЕД | П, У |
| Название | НАЗВ |  |
| Преподаватели | Табельный номер | ТАБ\_НОМ | П, У |
| Фамилия | ФАМ |  |
| Имя | ИМ |  |
| Отчество | ОТЧ |  |
| Дисциплина | ДИСЦ |  |
| Расписание | Код дня недели | КОД\_НЕД | С |
| Номер пары | НОМ\_ПАР |
| Номер аудитории | НОМ\_АУД |
| Код дисциплины | КОД\_ДИСЦ |
| Код группы | КОД\_ГР |
| Табельный номер  преподавателя | ТАБ\_НОМ |
| Расписание звонков | Номер пары | НОМ\_ПАР | П, У |
| Начало | НАЧ |  |
| Конец | КОН |  |
| Учебный план | Дисциплина | ДИСЦ | П, У |
| Лекция | ЛЕК |  |
| Учебная практика | УЧ\_ПР |  |
| Производственная практика | ПР\_ПР |  |
| Курсовая работа | КУРС |  |
| Дипломная работа | ДИПЛ |  |
| Контрольная | КОНТР |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Информационный**  **объект** | **Наименование реквизита** | **Обозначение** | **Признак ключа** |
|  | Зачёт | ЗАЧ |  |
| Консультация | КОНС |  |
| Экзамен | ЭКЗ |  |
| Итого часов | ИТ\_Ч |  |

Обозначения: У – уникальный, П – простой, С - составной.

Полученные информационные объекты отвечают требованиям нормализации, т. е. находятся в третьей нормальной форме.

## 2.6. Определение связей информационных объектов, построение ER-диаграммы

Существуют следующие типы отношений между объектами:

* Один-к-одному (1:1). Каждому экземпляру первого информационного объекта соответствует только один экземпляр второго информационного объекта
* Один-ко-многим (1:М). Каждому экземпляру одного информационного объекта соответствует несколько экземпляров другого информационного объекта, а каждому экземпляру второго информационного объекта соответствовать не более одного экземпляра первого информационного объекта.
* Многие-ко-многим (М:М). Каждому экземпляру одного информационного объекта соответствует несколько экземпляров другого информационного объекта и каждому экземпляру второго информационного объекта может соответствовать несколько экземпляров первого.

В результате анализа получили 8 информационных объектов: ДНИ НЕДЕЛИ, АУДИТОРИИ, РАСПИСАНИЕ ЗВОНКОВ, УЧЕБНЫЙ ПЛАН, ГРУППЫ, ДИСЦИПЛИНЫ, РАСПИСАНИЕ, ПРЕПОДАВАТЕЛИ.

Проведем попарный анализ связей между ними:

Дни недели (1) Расписание (7). Тип связи 1:М. Связь между этими объектами обеспечивается с помощью реквизита Код дня недели.

Аудитории (2) Расписание (7). Тип связи 1:М. Связь между этими объектами обеспечивается с помощью реквизита Номер аудитории.

Расписание звонков (3) Расписание (7). Тип связи 1:М. Связь между этими объектами обеспечивается с помощью реквизита Номер.

Учебный план (4) Дисциплины (6). Тип связи 1:М. Связь между этими объектами обеспечивается с помощью реквизита Дисциплина.

Группы (5) Расписание (7). Тип связи 1:М. Связь между этими объектами обеспечивается с помощью реквизита Код группы.

Дисциплины (6) Расписание (7). Тип связи 1:М. Связь между этими объектами обеспечивается с помощью реквизита Код дисциплины.

Преподаватели (8) Расписание (7). Тип связи 1:М. Связь между этими объектами обеспечивается с помощью реквизита Табельный номер преподавателя.

Я построила ER-диаграмма предметной области «Расписание»

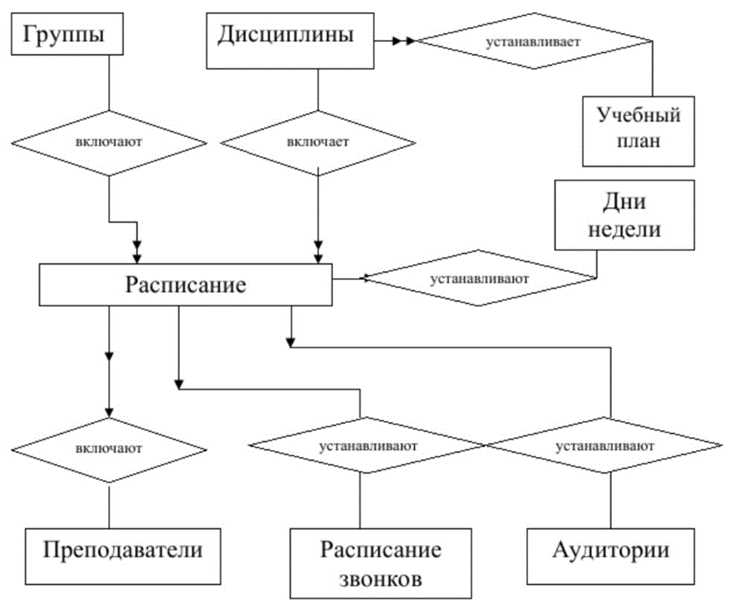


Рис.4.1 ER-диаграмма предметной области «Расписание»

Диаграммы отношений между таблицами помогают наглядно представить проект базы данных. Диаграммы объект-отношение (Entity-relation - ER), которые также называют диаграммами объект-атрибут-отношение (entity-atribute-relation – EAR), представляют собой один из наиболее широко используемых методов для изображения отношений между таблицами базы данных. Эллипсы над прямоугольниками таблиц обозначают классы атрибутов (поля), входящие в отношение. Ромбы, соединяющие пары прямоугольников таблиц и эллипсы полей, представляющих отношение между полями.

## 2.7. Определение логической и физической моделей базы данных

Логическая модель реляционной базы данных является адекватным отображением полученной информационно-логической модели предметной области. Для канонической модели не требуется дополнительных преобразований. Каждый информационный объект модели данных отображается соответствующей реляционной таблицей. Структура реляционной таблицы определяется реквизитным составом соответствующего информационного объекта, где каждый столбец (поле) соответствует одному из реквизитов. Ключевые реквизиты образуют уникальный ключ реляционной таблицы. Для каждого столбца таблицы задается тип, размер данных и другие свойства. Топология проекта схемы данных практически совпадает с топологией информационно – логической модели.

Логическая модель базы данных приведена ниже (рис. 4.2):

Изображение выглядит как текст, снимок экрана, диаграмма, Шрифт

Автоматически созданное описание

Рис.4.2 Логическая модель БД «Расписание занятий»

# ЗАКЛЮЧЕНИЕ

В ходе выполнения курсовой работы были продемонстрированы теоретические и практические навыки и умения, полученные в рамках обучения. Подводя её итоги, можно выделить следующие наиболее значимые достигнутые результаты:

* Проведение анализа предметной области сферы применения разрабатываемого, согласно теме работы, программного продукта, с выявлением основных параметров и функций проектируемой программы, представленное в аналитической части работы;
* Разработка модели программного продукта, включающая в себя: написание технического задания, разработку UML, ER диаграммы, концептуальная модель, а также возможных сценариев взаимодействия, с ней пользователя показанных в главе посвящённой проектной части работы;

В заключение можно сказать, что поставленная передо мной цель, а именно разработать эскизный проект информационной системы «Расписание занятий» для ГБПОУ ЯНАО «Ямальский многопрофильный колледж» была достигнута, все задачи выполнены.

В будущем создание АИС в готовом виде система должна избавить диспетчера от бумажной работы, предоставить возможность анализа ошибок в реальном времени и обеспечить формирование готовых листов с расписанием в цифровом графическом формате, удобном для студентов и преподавателей.

# СПИСОК ЛИТЕРАТУРЫ

**Ресурсы удалённого доступа**

Информационные системы - определение, цель создания, структура. - Текст: электронный // Учебный файл. – URL: <https://studfile.net/preview/4545719/> (дата обращения 12.11.2023).

Курс «Проектирование информационных систем» - основные понятия технологии проектирования информационных систем (ИС). Текст: электронный // НОУ Интуит. – URL: <https://www.intuit.ru/studies/courses/2195/55/lecture/1618?page=3> (дата обращения 16.12.2023).

Информационная система - задачи и функции ИС. - Текст: электронный // Студопедия. – URL: [https://st](https://st/)[udopedia](https://studopedia.ru/3_9102_lektsiya--ponyatie-informatsionnoy-sistemi-zadachi-i-funktsii-is.html).ru/3\_9102\_lektsiya--ponyatie-informatsionnoy-sistemi-zadachi-i-funktsii-is.html (дата обращения 12.11.2023).

Персональный компьютер – термин, виды и типы. – Текст: электронный // Топсто. – URL: [https://to](https://to/)[psto](https://topsto-crimea.ru/chto-takoe-personalnyy-kompyuter-pk/)-crimea.ru/chto-takoe-personalnyy-kompyuter-pk/ (дата обращения 29.11.2023).

Аганина, Д. А. Проблемы автоматизированного расписания образовательного процесса / Д. А. Аганина. – Текст: электронный. // Молодой ученый. — 2018. — № 42 (228). – С. 42-43. – URL: <https://moluch.ru/archive/228/53149/> (дата обращения: 05.01.2024).

Семенюта, И.С. Методика анализа информационной структуры базы данных автоматизированной системы составления расписаний / И.С. Семенюта. – Текст: электронный. // Научный журнал КубГАУ. – 2011. – № 09. – С.3-4. – URL: <http://ej.kubagro.ru/2011/09/pdf/06.pdf> (дата обращения: 23.01.2024).

ГОСТ 19.201-78 ЕСПД – техническое задание;требования к содержанию и оформлению. – Текст: электронный // Государственный стандарт, 1980. – URL: <https://www.swrit.ru/doc/espd/19.201-78.pdf> (дата обращения: 28.03.2024).

ГОСТ 34.602-89 Информационная технология – комплекс стандартов на автоматизированные системы; техническое задание на создание автоматизированной системы. – Текст: электронный // Москва Стандартинформ, 2009. – URL: <https://files.stroyinf.ru/Data2/1/4294850/4294850134.pdf?ysclid=lw4hjeml7h804382643> (дата обращения: 01.04.2024).

Каюмова, А.В. Визуальное моделирование систем в StarUML: учебное пособие / А.В. Каюмова. – Текст: электронный // Учебное пособие КФУ, 2013. – URL: <https://kpfu.ru/docs/F1686173080/Kajumova.pdf> (дата обращения: 16.04.2024).

Холод И. И. Опыт автоматизации процесса составления расписания в вузе , И. И. Холод, В. С. Иванов, И. С. Григорьев. – Текст: электронный. // Учебное пособие СПбГЭТУ «ЛЭТИ», 2020. – URL: <https://www.researchgate.net/publication/361562174_Opyt_avtomatizacii_processa_sostavlenia_raspisania_v_vuze> (дата обращения: 22.04.2024).

Черушева, Т.В. Проектирование программного обеспечения: учебное пособие / Т.В.Черушева. – Текст: электронный // Пенза: Издательство ПГУ, 2014. – 170с. – URL: <https://dep_vipm.pnzgu.ru/files/dep_vipm.pnzgu.ru/books/cherusheva_proektirovanie_programmnogo_obespecheniya.pdf?ysclid=lw4jrvyfy5647890602> (дата обращения: 21.04.2024).

Гриченко, Н.Н. Базы данных; разработка клиентских приложений на платформе Net : учебное пособие / Н. Н. Гриченко, А.Ю. Громов, А.В. Благодаров, . –М.: КУРС, 2018. - 288 с.– Текст: непосредственный.

Богатырев, М. Ю. Генетические алгоритмы: принципы работы, моделирование, применение: монография / М. Ю. Богатырев. - Тула : Экбсон, 2013. - 152 с. – Текст: непосредственный.

Перлова, О.Н. Проектирование и разработка информационных систем: учебник / О.Н. Перлова. - М.: Академия, 2018. - 272 c. – Текст: непосредственный.

Абухания Амер, Ю. А . Модели, алгоритмы и программные средства обработки информации и принятия решений при составлении расписаний занятий на основе эволюционных методов / Ю. А Абухания Амер – Новочеркасск, 2016. – 19 с.– Текст : непосредственный.

Радыгин, В.Ю. Базы данных и СУБД: учебно-методическое пособие / В. Ю. Радыгин . – М.: МГИУ, 2011. - 70 c. – Текст: непосредственный.

Федорова, Г.Н. 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. Текст: непосредственный.

# ПРИЛОЖЕНИЕ А

## 3.1. Техническое задание

Создание автоматизированной информационной системы – процесс трудоемкий. Важно учесть все моменты. Это позволить избежать появления ошибок в процессе работы готовой системы после ее внедрения. Именно поэтому сначала нужно составить подробное техническое задание на создание.

Техническое задание (ТЗ) - документ, утверждённый в соответствующем порядке. Он определяет требования, цели и основные исходные данные, которые необходимы для разработки аналитической информационной системы и содержит предварительную оценку эффективности системы. Оно разрабатывается заказчиком при непосредственном участии разработчика.

Кроме требований к функционалу системы в ТЗ прописываются условия эксплуатации, безопасности, надежности ИС и поэтапный порядок ее разработки.

1.Техническое задание

1.1Общие сведения

1. Наименование системы

Наименование системы: разработка эскизного проекта информационной системы "Расписание занятий" для автоматизации деятельности ГБПОУ ЯНАО «Ямальского многопрофильного колледжа».

1.1.2Основные понятия, определения и сокращения

Расписание занятий – группа документов, содержащих данные для учебных групп и преподавателей.

Пользовательский интерфейс – объект, обеспечивающий высокую информативность выводимой на экран информации, организующий удобство ее вывода и обработки пользователем автоматизированной системы.

1.1.3 Порядок оформления и предъявления заказчику результатов работ

Работы по созданию системы выполняются в два этапа:

* разработка проекта (продолжительность – 16 недель).
* тестирование и сдача проекта заказчику (продолжительность – 10 недель).

Общий срок работ по созданию расписания составляет 26 недель начало работ 01.11.2023 г., окончание работ 08.05.2025 г.

Календарный план работы по проектированию ИС «Расписание занятий» для предоставления заказчику.

Таблица 1 – Календарный план работы по проектированию ИС «Расписание занятий»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | Наименование работ | Исполнитель | Сроки исполнения | Отметка о выполнении |
| 1 | Разработка проекта | Куйбина Е.В. | 01.11.2024г.-02.03.2025г. |  |
| 2 | Тестирование проекта | Куйбина Е.В. | 09.03.2025г.-08.05.2025г. |  |
| 3 | Сдача проекта ИС | Куйбина Е.В. |  |  |

По результатам приёмо-сдаточных испытаний составляется акт сдачи-приёмки, в котором описывается испытуемый проект, её технические характеристики и их соответствие техническому заданию на задачу.

1.2 Назначение и цели создания системы

1.2.1 Назначение системы

Создание инструмента для планирования, отображения и управления расписанием занятий в колледже, с возможностью автоматизации процессов и уменьшения затрат временных и человеческих.

1.2.2 Цели создания системы

С целью автоматизации планирования было разработано решение, упрощающее процесс создания электронного расписания на основе анализа имеющихся учебных планов специальностей, позволяющий анализировать структуру нагрузки.

Общие требования, предъявляемые к системе, следующие:

* использование норм времени для расчета объемов учебной нагрузки;
* использование информации из учебных планов специальностей;
* автоматической генерации расписания на основе заданных параметров;
* внесения изменений в расписание в реальном времени;
* формирование отчетных форм;
* удобный пользовательский интерфейс.

1.3 Характеристика объектов автоматизации

Объектом автоматизации является учебная часть колледжа, отвечающей за составление расписания. Задача предназначена для представления расписания занятий студентов в электронном виде, позволяющем внести дополнительный уровень контроля и исключения ошибок при расстановке предметов по времени, по аудиториям, исключению постановки разных групп в одну аудиторию.

При составлении расписания занятий необходимо учитывать:

* распределение учебной нагрузки по дням недели;
* распределение учебной нагрузки в течение каждого дня отдельно;
* чередование предметов разных видов деятельности;
* учебная нагрузка и занятость преподавателей, работающих в группе.

1.4 Требования к системе

1.4.1 Требования к системе в целом

Требования к структуре и функционированию системы

Функциональное назначение информационной системы:

– обеспечение всех участников учебного процесса данными о расписании занятий, занятости преподавателей, учебных помещений, изменениях в расписании.

Информационная система должна обеспечить выполнение следующих функций:

* подготовка расписания занятий с учетом требований учебных планов, пожеланий преподавателей и возможностей учебных помещений;
* подготовка и коррекция расписания должна проводиться предварительно с указанием сроков начала и окончания действия изменений;
* возможность добавления, удаления и редактирования учебных групп;
* просмотр расписания для студентов и преподавателей с учётом индивидуальных параметров;
* уведомления о изменениях в расписании через электронную почту или уведомления в приложении;
* возможность автоматической генерации расписания с учетом доступных ресурсов (аудиторий, преподавателей);
* возможность управления списком преподавателей, их расписанием и доступностью в определенные дни и часы;
* поддержка административной панели для управления системой и данными.

Не функциональные требования:

* интерфейс должен быть интуитивно понятным и легко использоваться как студентами, так и персоналом колледжа;
* информационная система должна обеспечивать безопасное хранение данных и защиту информации;
* время отклика системы должно быть минимальным, чтобы обеспечить быстрый доступ к информации.

Требования к численности и квалификации персонала системы и режиму его работы

Пользователями расписания являются:

1. Администрация (администратор, директор, завуч и другие заинтересованные лица);
2. Преподаватели;
3. Студенты;
4. Сторонние потребители (в том числе проверяющие и контролирующие организации).

Пользователи Расписания должны:

– иметь навыки работы на ПК в качестве пользователя;

– знать принципы работы с ОС Windows 2000/XP/7/8;

– пройти обучение для работы с ИС «Расписание занятий

Администратор ИС «Расписание занятий» должен иметь образование со специализацией в области разработки информационных систем и баз данных, обладать навыками администрирования современных SQL-серверов и серверов приложений, пройти обучение основам работы с ИС «Расписание занятий» в объеме технической (эксплуатационной) документации (Руководство администратора) на систему.