

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования

«Сибирский государственный индустриальный университет»

Кафедра прикладных информационных технологий и программирования

## КУРСОВОЙ ПРОЕКТ

по дисциплине «Проектная деятельность 3»

на тему «Разработка информационной системы для составления учебного  
расписания СибГИУ»

Выполнили: обучающиеся гр. ИСП-20-2

\_\_\_\_\_ Токмаков А.К.  
(дата) (подпись) (инициалы, фамилия)

\_\_\_\_\_ Перунов Я. А.  
(дата) (подпись) (инициалы, фамилия)

\_\_\_\_\_ Швецов К.А.  
(дата) (подпись) (инициалы, фамилия)

\_\_\_\_\_ Михайлов М.К.  
(дата) (подпись) (инициалы, фамилия)

Руководитель курсового проекта:

\_\_\_\_\_  
(уч. степень, звание, инициалы и фамилия)

\_\_\_\_\_ (оценка) \_\_\_\_\_ (дата) \_\_\_\_\_ (подпись)

Новокузнецк

2022г.

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Сибирский государственный индустриальный университет»  
Кафедра прикладных информационных технологий и программирования

УТВЕРЖДАЮ

Заведующий кафедрой

\_\_\_\_\_ Рыбенко И.А.  
(подпись) (инициалы, фамилия)  
«\_\_\_» \_\_\_\_\_ 2022г.

## ЗАДАНИЕ НА ВЫПОЛНЕНИЕ КУРСОВОГО ПРОЕКТА

по дисциплине «Проектная деятельность 3»  
на тему «Разработка информационной системы для составления учебного  
расписания СибГИУ»

проектной команды: Токмаков А.К., Перунов Я. А., Швецов К.А., Михайлов М.К.

1. Согласно структуре курсового проекта произвести инициацию и планирование проекта.

2. Разработать информационную систему для составления учебного расписания СибГИУ

3. Оформить отчёт по курсовому проектированию, подготовить презентацию по результатам проектирования и доклад.

Задание к курсовому проектированию приняли:

« 07 » _____ сентября _____ 2022г.	<u>Токмаков А.К.</u>	_____
« 07 » _____ сентября _____ 2022г.	<u>Перунов Я. А.</u>	_____
« 07 » _____ сентября _____ 2022г.	<u>Швецов К.А.</u>	_____
« 07 » _____ сентября _____ 2022г.	<u>Михайлов М.К.</u>	_____

Руководитель курсового проекта	<u>Гасымов Р.Р</u>	_____
--------------------------------	--------------------	-------

## Содержание

Введение.....	4
1. Инициация и планирования проекта .....	6
1.1. Анализ предметной области и целевой аудитории .....	6
1.2. Формирование цели и стратегии проекта.....	6
1.3. Устав проекта .....	7
1.4. Анализ рисков.....	11
1.5. Обоснование выбора программных и технических средств .....	12
1.6. Календарное планирование.....	19
2 Содержание проекта.....	20
2.1 Техническое задание.....	20
2.2 Разработка MVP .....	25
2.2.1 Описание по видам обеспечения.....	25
2.2.2 Анализ и доработка.....	45
Заключение .....	46
Библиографический список .....	48

## Введение

В современном мире информационные технологии играют огромную роль в различных сферах жизни, в том числе и в образовании. Составление учебного расписания является одной из важнейших задач в университете, и для ее решения необходимо использовать современные технологии. В этом контексте, разработка информационной системы для составления учебного расписания СибГИУ, будет направлена на автоматизацию отдела составления расписания. Это позволит значительно упростить и ускорить процесс составления расписания, а также обеспечить быстрый и удобный доступ к нему для студентов и преподавателей.

В современном образовании информационные технологии играют важную роль. В разных учебных заведениях есть разные предметные области, и в каждой из них можно применить информационные технологии для улучшения образовательного процесса. Создание информационной системы для расписания является одним из приоритетных направлений информатизации образования. В университете имеется проблема с составлением расписания вручную, что неудобно для студентов и работников университета. Тратится много времени на заполнение и обнаружение ошибок составления расписания на бумаге, а также расписание долго доходит до студентов, так как этим занимаются другие люди.

Информационная система для расписания позволит предотвратить эти проблемы на этапе составления расписания, так как она будет подсказывать наложения преподавателей и групп, так же в какое время свободны выбранные преподаватель или группа, для удобства составителей расписания и всех пользователей системы, можно будет сразу выложить расписание на сайт.

Одним из основных преимуществ информационных систем для составления учебного расписания и удобного просмотра является, возможность учета множества факторов, влияющих на составление расписания. Такие факторы могут включать в себя занятость преподавателей, наличие свободных аудиторий, наличие основных и альтернативных занятий. Автоматизация процедуры составления учебных занятий позволяет упростить методы формирования расписания и снизить к ним требования [1].

Создание информационной системы для составления учебного расписания СибГИУ и сайта для удобного просмотра, продемонстрирует навыки создания программных продуктов и позволит применить их на практике. Более того, проект имеет социальную значимость, так как позволит значительно улучшить процесс составления расписания и сделать его более доступным для студентов и преподавателей.

В дальнейшем планируется внедрение разработанного комплекса программного обеспечения в отдел составления расписания СибГИУ. Это позволит автоматизировать процесс составления расписания занятий вуза, что в свою очередь повысит качество обучения, экономическую эффективность обучения, комфортность учебы студентов и работы преподавательского состава.

## **1. Инициация и планирования проекта**

### **1.1. Анализ предметной области и целевой аудитории**

На данный момент в рассматриваемом учебном заведении не хватает учета множества факторов, влияющих на составление расписания. Такие факторы могут включать в себя занятость преподавателей, наличие свободных аудиторий.

Разработка информационной системы для составления учебного расписания СибГИУ, связана с автоматизацией составления расписания занятий в учебных заведениях.

Объектом предметной области является процесс составления расписания занятий, а предметом - все аспекты, связанные с этим процессом, включая организацию учебного процесса, занятость преподавателей, наличие свободных аудиторий, наличие основных и альтернативных занятий и т.д.

Целевой аудиторией данной информационной системы являются учебные заведения, которые нуждаются в автоматизации составления расписания занятий. Это могут быть как школы, так и высшие учебные заведения

Использование информационной системы позволяет оптимизировать учебный процесс, учитывать пожелания учеников и преподавателей с целью грамотного распределения нагрузки и ликвидации окон между занятиями. Также система позволяет создавать и вести аналитику по эксплуатации аудиторий и выявлению самых востребованных преподавателей.

### **1.2. Формирование цели и стратегии проекта**

Главная цель проекта – разработать и внедрить информационную систему составления и просмотр расписания в учебные заведения, с целью повышения качества и скорости составления расписания [2].

Стратегии достижения целей: увеличение эффективности работников в учебных заведениях, по средствам перехода от бумажных носителей информации на

цифровой вид составления расписания, тем самым способствуя повышению качества и освобождения составителя расписания от рутинной работы.

### **1.3. Устав проекта**

Наименование проекта: «Разработка информационной системы для составления учебного расписания СибГИУ»

Дата начала проекта: 23.09.2022

Дата окончания проекта: 30.05.2023

Цель проекта высшего уровня – разработать информационную систему для составления учебного расписания СибГИУ приложения и сайта также сайта для просмотра расписания.

#### **Задачи проекта**

1. Изучить конечного пользователя информационной системы и его предпочтения.
2. Изучить взаимодействие баз данных между собой, при разных условиях использования и выбрать наиболее подходящий.
3. Проектирование базы данных;
4. Разработка графического интерфейса пользователя;
5. Разработка приложения и сайта для просмотра расписания;
6. Подключение базы данных к приложению и сайту.

#### **Продукт проекта**

Продукт проекта – разработанная ИС, для составления расписания и его просмотра.

#### **Ограничение проекта**

Ограниченные финансовые ресурсы на разработку информационной системы для составления учебного расписания СибГИУ, сайта и сайта для просмотра расписания.

#### **Заинтересованные стороны проекта**

В таблице 1 указаны все заинтересованные стороны проекта, а также расписаны их интересы.

Таблица 1 – Заинтересованные стороны проекта

№ п/п	Заинтересованная сторона	Интерес
1	Участники проекта (студенты СибГИУ)	<ul style="list-style-type: none"> <li>1. Защита курсовой работы;</li> <li>2. Получение новых навыков работы с базами данных в ходе разработки программного продукта;</li> <li>3. Получения новых знаний в разработке информационных систем;</li> <li>4. Получения новых знаний в разработке пояснительной записки к проекту по ГОСТ.</li> </ul>
2	Руководитель проекта (Токмаков А.К.)	<ul style="list-style-type: none"> <li>1. Успешная разработка проекта;</li> <li>2. Внедрение программного продукта;</li> <li>3. Реализация востребованного продукта.</li> <li>4. Обеспечение удобного и интуитивно понятного интерфейса для пользователей, чтобы они могли легко и быстро находить нужную информацию.</li> </ul>
3	Заказчик (СибГИУ)	<ul style="list-style-type: none"> <li>1. Перевод составления расписания из бумажного вида в цифровой;</li> <li>2. Увеличение эффективности сотрудников;</li> <li>3. Автоматизация составления расписания занятий для учебных заведений.</li> <li>4. Удобный и интуитивно понятный интерфейс для пользователей.</li> <li>5. Безопасность данных и защита от несанкционированного доступа к информации.</li> </ul>



*Продолжение таблицы 1*

4	Пользователь (Учебный отдел)	1. Удобный просмотр расписания 2. Обеспечение безопасности данных и защиты от несанкционированного доступа к информации. 3. Возможность быстрого и удобного доступа к информации о расписании занятий в любое время и в любом месте. 4. Возможность получения уведомлений о изменениях в расписании занятий. 5. Возможность быстрого и удобного поиска информации о преподавателях и аудиториях.
---	---------------------------------	--

Ресурсы проекта

1. Временные ресурсы: 8 месяцев.
2. Материальные ресурсы: ПК, сеть интернет, Visual Studio 2019.
3. Трудовые ресурсы: 1 руководитель, 1 разработчик, 1 дизайнер, 1 тестировщик.

Последовательность работ по проекту

В таблице 2 описывается последовательность работ и этапы проекта по разработки информационной системы для составления учебного расписания СибГИУ

Таблица 2 – Последовательность работ по проекту

№ п/п	Этапы проекта
1	Инициация проекта
1.1	Разработка устава проекта
2	Планирование проекта
2.1	Определение последовательности работ
2.2	Составление диаграммы Ганта
2.3	Составление технического задания
3	Реализация проекта
3.1	Выбор платформы проекта
3.2	Выбор языка программирования

Продолжение таблицы 2

3.3	Выбор и обоснования методов реализации
3.4	Проведение исследования конечного пользователя продукта
3.5	Составление календаря созвонов и отчета проделанной работы
3.6	Разработка БД
3.7	Разработка front-end части ИС
3.8	Разработка back-end части ИС
3.9	Создание сайта для просмотра расписания
3.10	Подведение итогов работы по разработке информационной системы для составления учебного расписания СибГИУ, сайта и сайта для просмотра расписания
4	Завершение проекта
4.1	Создание отчета и презентации
4.2	Демонстрация и защита проекта

Команда проекта

Научный руководитель проекта: Гасымов Р.Р., Белавенцева Д.Ю.

Руководитель-разработчик проекта: Токмаков Антон Константинович.

Разработчик-дизайнер: Михайлов Максим Константинович

Дизайнер-разработчик: Швецов Кирилл Антонович

Тестировщик-документовед: Перунов Ярослав Алексеевич

Обязанности участников проекта

Полный перечень обязанностей участников проекта, а также их должности, прописаны в таблице 3.

Таблица 3 - Обязанности участников проекта

Должность	Участник	Обязанности
Руководитель проекта	Токмаков Антон Константинович	Руководство проектом
		Разработка базы данных
		Разработка back-end части ИС
Исполнитель проекта	Михайлов Максим Константинович	Разработка back-end части сайта
		Разработка документации проекта

*Продолжение таблицы 3*

Оценщик проекта	Перунов Ярослав Алексеевич	Разработка методов тестирования проекта
		Разработка front-end части сайта
Разведчик проекта	Швецов Кирилл Антонович	Разработка документации проекта
		Разработка back-end части сайта

Средства коммуникации участников проекта

Discord, ВК, личные встречи.

#### **1.4. Анализ рисков**

1. Риск неправильного планирования и управления проектом, что может привести к просрочкам в сроках выполнения заданий по проекту. Минимизация, правильное планирование и использование современных методов управления проектами, а также контроль за выполнением работ.

2. Нехватка времени на реализацию поставленной задачи из-за плотного графика. Решение правильное планирование времени.

3. Невыполнения работ ответственным за его выполнения, преднамеренно или из-за неправильного распределения личного времени. Для их упразднения используются объяснительные.

4. Недостаточно знаний для реализации поставленных задач. Решение помощь одноклассников, интернет-курсы.

5. Нестабильная работа ИС. Решение информационная система будет покрыта различными тестами.

6. Аппаратные сбои, потеря важных данных по проекту [3]. Для их минимизации будем использовать систему контроля версий и облачные хранилища, которые обслуживаются крупными и надежными компаниями.

7. Риск неправильной работы с данными. Для минимизации этого риска необходимо обеспечить безопасность данных, а также проводить регулярное резервное копирование данных.

8. Другие риски, которые невозможно предсказать. Решение плановый срок сдачи проекта сделать раньше, тем самым предусмотреть данный риск.

### **1.5. Обоснование выбора программных и технических средств**

Для разработки информационной системы составление расписания нам понадобятся средства редактирования кода для того, чтобы написать проект. Так как в специализированных IDE присутствует удобная подсветка кода, которая помогает при чтении и редактировании большого объема кода. IDE ускорит и упростит разработку приложения в несколько раз.

Для проектирования интерфейса приложения понадобятся специализированные редакторы, которые помогут облегчить работу дизайнера. Дизайнер может сначала графически построить прототип будущего приложения легко изменяя его без кода. Такое средство проектирования может уменьшить будущую работу и способствует понять на раннем этапе проекта, требование конечного пользователя к дизайну приложения.

Для хранения информации о парах, преподавателях и подобной информации, нужно использовать базу данных. Для удобного взаимодействия с БД, необходимо выбрать СУБД которая будет удовлетворять всем потребностям разработчика. СУБД поможет разработчику быстро и без труда взаимодействовать с большим количеством таблиц в БД, не задумываясь о том, как их связывать и не ломать, так как за этим будет следить СУБД.

Так же для получения доступа к сайту обычному пользователю, необходимо иметь свой сервер или выложить его на хостинг в других компаниях. Так как свой сервер довольно сложно настраивать, а также у него могут быть частые свои и проблемы с безопасностью было принято решение использовать хостинг.

### 1.5.1 Средства разработки интерфейса

Для разработки интерфейса приложения будет использована одна из нижеперечисленных программ:

- Figma;
- Adobe Photoshop;
- CoreIDRAW;

Figma – программа для дизайна. С помощью нее можно создавать не только прототипы, но и сами интерфейсы (сайты, приложения, панель управления). Весомым преимуществом данной программы является ее простота.

Photoshop – многофункциональный графический редактор, разрабатываемый и распространяемый компанией Adobe Systems. В основном работает с растровыми изображениями, однако имеет некоторые векторные инструменты. Продукт является лидером рынка в области коммерческих средств редактирования растровых изображений и наиболее известной программой разработчика.

CoreIDRAW – графический редактор векторной графики, разработанный канадской корпорацией Corel. В пакет CorelDRAW также входит редактор растровой графики Corel Photo-Paint и другие программы — например, для захвата изображений с экрана — Corel Capture. Программа векторизации растровой графики Corel Trace до 12 версии входила в пакет как самостоятельная программа.

В таблице 4 прописаны сравнительные характеристики всех программных средств для разработки интерфейса.

Таблица 4 – выбор программного средства для разработки интерфейса

Средство разработки интерфейса	Доступность программного обеспечения	Опыт работы	Функциональность
Figma	+	+	+
Adobe Photoshop	-	-	+
CoreIDRAW	+	+	+

Для создания интерфейса проекта "Разработка информационной системы для составления учебного расписания СибГИУ" был выбран графический редактор Figma. Этот выбор обоснован тем, что Figma является мощным инструментом для создания интерфейсов и прототипов, который позволяет работать над проектом в режиме реального времени и совместно с другими участниками команды.

Figma обладает широким набором функций и инструментов, которые позволяют создавать высококачественные и профессиональные интерфейсы. Кроме того, Figma является онлайн-сервисом, что позволяет работать с проектом из любой точки мира и на любом устройстве.

В целом, выбор Figma для создания интерфейса проекта "Разработка информационной системы для составления учебного расписания СибГИУ" обоснован его функциональностью, удобством использования и возможностью совместной работы над проектом.

### **1.5.2 Средства редактирование кода**

Из средств для редактирования можно выделить 3 IDE:

- JetBrains Rider;
- Eclipse;
- Visual Studio.

JetBrains Rider – это кроссплатформенная .NET IDE, основанная на IntelliJ и ReSharper с помощью которой можно разрабатывать приложения .NET, ASP.NET, .NET Core, Xamarin и Unity в системах Windows, Linux и Mac.

Eclipse – это свободная интегрированная среда разработки модульных кроссплатформенных приложений. Развивается и поддерживается Eclipse Foundation. Наиболее известные приложения на основе Eclipse Platform - различные «Eclipse IDE» для разработки ПО на множестве языков включая C#.

Visual Studio – это интегрированная среда разработки, стартовая площадка для написания, отладки и сборки кода, а также последующей публикации приложений. Помимо стандартного редактора и отладчика, которые есть в большинстве сред IDE,

Visual Studio включает в себя компиляторы, средства авто завершения кода, графические конструкторы и многие другие функции для улучшения процесса разработки [4].

В таблице 5 прописаны сравнительные характеристики программного средства для редактирования кода.

Таблица 5 – выбор программного средства для редактирования кода

Программа для выбора	Доступность программного обеспечения	Стабильность системы	Удобства использования
JetBrains Rider	-	-	+
Eclipse	+	-	-
Visual Studio	+	+	+

Visual Studio была выбрана для создания проекта "Разработка информационной системы для составления учебного расписания СибГИУ" в связи с ее мощными возможностями для разработки desktop приложений на платформе .NET. Visual Studio предоставляет широкий набор инструментов для разработки, отладки и тестирования приложений, а также интегрированную среду разработки, которая позволяет удобно работать с кодом и проектом в целом.

Удобство использование является субъективным критерием, но для команды, Visual Studio является отличным IDE для разработки. Для проекта «Разработка информационной системы для составления учебного расписания СибГИУ» Visual Studio является лучшим решением, так как на C# и Visual Studio пишутся большинство информационных систем, так же данную среду разработки удобно интегрировать с MS SQL.

### 1.5.3 Платформа для хостинга сайта

При выборе хостинга были выделены три компании:

- Handyhost.ru;

- Fozzy.com;
- SmartApe.ru.

Nandyhost.ru – российский хостинг, оказывающий услуги shared-хостинга, VDS и аренды серверов с 2009 года. Часть серверов находится в Москве и Санкт-Петербурге, остальные размещены в Германии. Техническая поддержка доступна круглосуточно по эл. почте, тикетам из панели управления, онлайн-чата и бесплатному телефону.

Fozzy.com – это американская компания с российскими корнями. Сервера хостера расположены во многих локациях по всему миру. Фози отличает очень производительный хостинг за умеренные деньги.

SmartApe.ru – это российская компания, которая предоставляет услуги хостинга по приемлемым ценам. Хостинг SmartApe работает с 2012 года, обеспечивая безлимитные услуги по размещению сайтов, круглосуточную поддержку клиентов и защиту от DDOS атак.

В таблице 6 прописаны сравнительные характеристики платформ для хостинга, такие как Nandyhost.ru, Fozzy.com, SmartApe.ru.

Таблица 6 – выбор платформа для хостинга

Платформа для хостинга сайта	Пробный период	Доступность в приобретении	Быстродействие	Стабильность	Поддержка
Nandyhost.ru	+	+	+	+	+
Fozzy.com	-	+	+	+-	+
SmartApe.ru	-	-	+-	+-	+

В качестве хостинга для сайта будет выступать компания Nandyhost.ru, так как она имеет пробный период для проверки работоспособности проекта в отличие от Fozzy.com и SmartApe.ru. Стабильность работы и быстродействие сервера – это очень важный критерий для нашей информационной системы составления расписания, так как расписание должно быть всегда под рукой, здесь вперед вырывается так же Nandyhost.ru, так как у Fozzy.com и SmartApe.ru стабильность немного хуже, но с



быстродействием у Fozzy.com все в порядке, в отличии от SmartApe.ru. Поддержка у всех выделенных платформ для хостинга отличная.

#### 1.5.4 СУБД

MySQL – это система управления реляционными базами данных с открытым исходным кодом (RDBMS). Разработку и поддержку MySQL осуществляет корпорация Oracle. Чаще всего систему управления базами данных MySQL используют в следующих проектах: CMS (Content Management System). В системах управления контентом у MySQL (в связке с PHP/Apache) нет более сильного соперника.

LiteSQL – это встраиваемая кроссплатформенная БД, которая поддерживает достаточно полный набор команд SQL и доступна в исходных кодах (на языке C). Исходные коды SQLite находятся в public domain, то есть никаких ограничений на использование нет. В 2005 году проект получил награду Google-O'Reilly Open Source Awards.

PostgreSQL – одна из самых популярных реляционных СУБД в мире. Непрерывной доработкой и выпуском новых версий PostgreSQL как опенсорсного проекта занимаются сотни разработчиков по всему миру. У СУБД нет владельца или управляющей компании, каждый пользователь может предложить изменения или включиться в процесс разработки [5].

В таблице 7 прописаны сравнительные характеристики выбранных СУБД, таких как MySQL, LiteSQL, PostgreSQL.

Таблица №7 – выбор СУБД

СУБД	Простота освоения	Скорость работы	опыт работы	Работа с сервером
MySQL	+-	+	+	+
LiteSQL	+-	+	+	-
PostgreSQL	+-	+	-	+

При проведении анализа и сравнения СУБД была выбрана MySQL для проекта "Разработка информационной системы для составления учебного расписания СибГИУ" в связи с ее мощными возможностями и широкой поддержкой. MySQL является одной из самых популярных СУБД в мире и обладает широким набором функций и инструментов для работы с данными. Кроме того, MySQL является бесплатной и открытой СУБД, что позволяет существенно снизить затраты на разработку проекта.

MySQL также обладает высокой производительностью и масштабируемостью, что позволяет обрабатывать большие объемы данных и поддерживать высокую нагрузку. MySQL является достаточно простой в освоении, у неё высокая скорость работы с большими данными, так же, как и у PostgreSQL, имеет возможность работы с сервером в отличие от LiteSQL [6]. Так как база данных находится на сервере, не иметь возможности работать с сервером важный критерий для информационной системы составления расписания.

MySQL пользуется достаточно большим спросом, так как эту СУБД поддерживает Oracle, компания, которая в базах данных удерживает первое место. Так же MySQL поддерживает множество языков программирования, один из которых C#, требующейся для нашего проекта.

В целом, выбор MySQL для проекта "Разработка информационной системы для составления учебного расписания СибГИУ" обоснован ее мощными возможностями, широкой поддержкой и бесплатностью.

## 1.6 Календарное планирование

### Диаграмма Ганта

На рисунке 1 представлена диаграмма Ганта, на ней представлены даты начала и завершения заданий, контрольные точки, зависимости между заданиями и исполнителями.

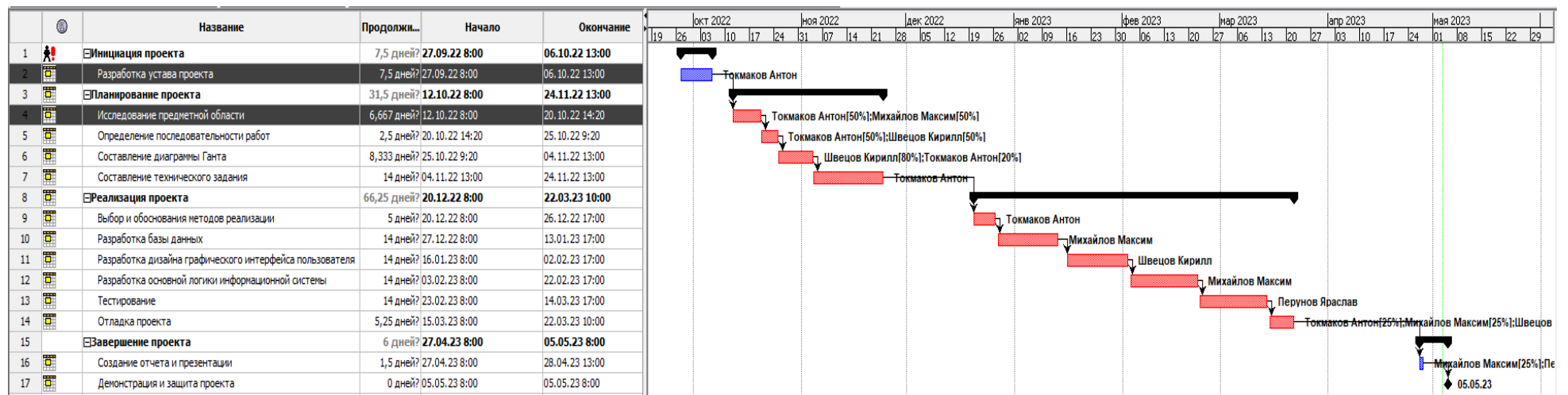


Рисунок 1 – диаграмма Ганта

## **2 Содержание проекта**

### **2.1 Техническое задание**

#### **Введение**

Разработать информационную систему для составления учебного расписания СибГИУ и сайта для удобного просмотра расписания. В дальнейшем внедрить данный комплекс ПО в СибГИУ.

#### **Основания для разработки**

Документ, на основании которого ведется разработка Устав проекта по разработке информационной системы для составления учебного расписания СибГИУ и сайта для расписания, дата 23.09.2022.

Тема проекта: Разработка информационной системы для составления учебного расписания СибГИУ.

Отсутствие информационной системы составление расписания в СибГИУ, а также не удобное его расположение для ежедневного просмотра.

#### **Назначение разработки**

Демоверсия информационной системы является первым этапом реализации сложного комплекса системы составление расписания, предназначенного для внедрения и использования в учебных заведениях.

Назначение системы – система должна удовлетворят требованиям учебных заведений по электронному составлению расписания, ссылаясь на законы в области образования. Способствует оптимизации рабочего времени составителя расписания, так как ИС сразу будет предупреждать о наложении пар у преподавателей либо групп,

а также быстрому исправлению расписания, за счет ИС [7].

### **Требования к программе или программному изделию.**

### **Требования к функциональным характеристикам.**

- ИС должна быть проста в использовании для рядового сотрудника;
- База данных выполняет свою главную функцию хранения информации расписания занятий;
- ИС должна стабильно работать;
- Хранение базы данных пользователей, имеющих доступ к системе и обеспечивать аутентификацию пользователей согласно имеющимся записей;
- Составление расписания;
- Просмотр расписания;
- Выбор групп и недели для просмотра расписания;
- Переход по ссылкам на курс в системе управления образовательными электронными курсами Moodle (далее Moodle);
- Изменение профиля.

### **Требование к надежности**

Надёжность системы в целом зависит от надежности используемой операционной системы. Серверная часть должна обслуживать без сбоев одновременное подключение и работу до 1000 пользователей. Обе части должны без потерь передавать информацию по каналу связи между клиентом и сервером.

### **Условия эксплуатации**

Стандартные условия эксплуатации программных продуктов. Необходимые сотрудники для обслуживания серверной части системы – системный администратор

для обслуживания удаленного сервера, группа составителей расписания, а также группа лиц, которые будут заполнять данные о парах, преподавателях и остальной информации. Численность групп может меняться в зависимости от учебного заведения и его потребностям.

Для управления приложением учебного расписания СибГИУ пользователю достаточно уметь пользоваться компьютером на базовом уровне.

### **Требования к составу и параметрам технических средств.**

Для нормальной работы приложения необходимо:

- Компьютер с процессором Intel i3 7100.
- Оперативная память не менее 4 Gb.
- Жесткий диск объемом не менее 128 Gb.
- Наличие адаптера подключения к сети (сетевой карты и т.п.).
- Установленная ОС Windows 10 и выше.

### **Требования к информационной и программной совместимости.**

Модель системы должна работать под управлением ОС Windows 10, 11, поэтому требуется совместимость исполняемого модуля и библиотек динамического подключения стандартам, используемым этими ОС на платформе IBM PC. Модель должна использовать свой протокол передачи данных высокого уровня как надстройку над TCP/IP. Для хранения информации требуется использование баз данных формата SQL.

Для доступа к базе данных необходима сетевая карта и доступ в интернет.

### **Требования к маркировке и упаковке.**

Не предъявляются.

### **Требования к транспортированию и хранению.**

Не предъявляются.

### **Специальные требования.**

Не предъявляются.

### **Требования к программной документации.**

Программной документацией к разрабатываемой информационной системы составление расписания является расчётно-пояснительная записка, которая будет приложена к программе.

### **Стадии и этапы разработки.**

В таблице 8 прописаны все стадии и этапы разработки проекта информационной системы для составления учебного расписания СибГИУ и сайта для удобного просмотра расписания. Так же прописаны сроки выполнения задач проекта и на каждую задачу назначен исполнитель. Общее количество этапов, на которые разделен проект равняется 13

Таблица 8 – Стадии и этапы разработки

№	Содержание работы	Срок	Исполнитель этапы разработки
1	Разработка устава проекта	7 дней	Токмаков Антон;
2	Исследование предметной области	6 дней	Токмаков Антон; Михайлов Максим;
3	Определение последовательности работ	2 дня	Токмаков Антон; Швецов Кирилл;
4	Составление диаграммы Ганта	8 дней	Швецов Кирилл; Токмаков Антон;

*Продолжение таблицы 8*

5	Составление технического задания	14 дней	Токмаков Антон;
6	Выбор и обоснования методов реализации	5 дней	Токмаков Антон;
7	Разработка базы данных	14 дней	Михайлов Максим
8	Разработка дизайна графического интерфейса пользователя	14 дней	Швецов Кирилл;
9	Разработка основной логики информационной системы	14 дней	Михайлов Максим;
10	Тестирование	14 дней	Перунов Ярослав;
11	Отладка проекта	5 дней	Токмаков Антон; Михайлов Максим; Швецов Кирилл; Перунов Ярослав
12	Создание отчета и презентации	1 день	Токмаков Антон; Михайлов Максим; Швецов Кирилл; Перунов Ярослав
13	Демонстрация и защита проекта	1 день	Токмаков Антон; Михайлов Максим; Швецов Кирилл; Перунов Ярослав

**Порядок контроля и приемки**

Испытание представленной модели и контроль качества ее работы провести на базе компьютерного класса кафедры. Во время испытаний проверить работу системы по следующим позициям:

- Запуск серверной и клиентской частей системы;
- Соединение пользователя с БД;



- Оценка удобства интерфейса;
- Проверка корректной записи информации о парах в БД;
- Корректное получение результатов из БД;
- Корректное составление расписание;
- Проверка создания экзамена;
- Проверка отображения экзамена;
- Просмотр расписания всех видов;
- Корректность добавления информации в справочники;
- Общее функционирование системы.

## **2.2 Разработка MVP**

### **2.2.1 Описание по видам обеспечения**

#### **2.2.1.1 Информационное обеспечение**

##### **Use-case**

На рисунке 2 показан use-case информационной системы. Гость в ИС сможет зарегистрироваться и посмотреть расписание, так же регистрация в системе включает в себя редактирование профиля.

Пользователь в ИС сможет просматривать расписание студента, /преподавателя/свободных кабинетов, печатать расписание и просматривать предстоящие экзамены, так же студент может редактировать свой профиль.

Составитель расписание будет наследовать все функции от Студента, а также имеет свои, такие как Составление и редактирование расписание и экзаменов.

Так же в системе присутствует Администратор, который будет наследовать все функции от Пользователя, а также будет иметь такие функции как удаления пользователя и изменение статуса пользователя, добавление пользователя.

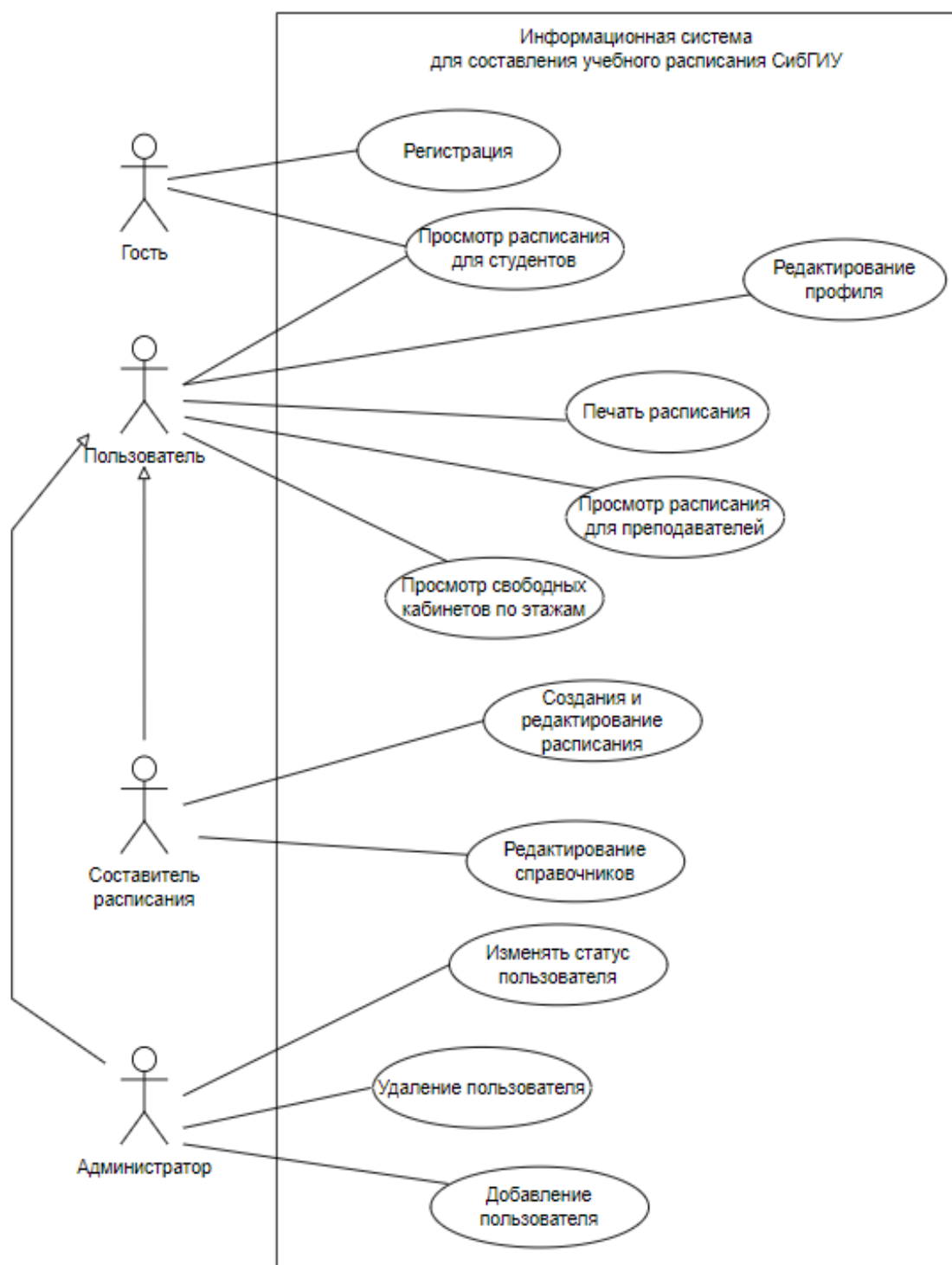


Рисунок 2 – use-case

На рисунке 3 представлена схема, в которой, на нулевом уровне диаграммы в информационную систему в управление входит Внутренние правила СибГИУ и Учебный план, которому надо придерживаться для составления расписания.

На вход будет подаваться данные по дисциплинам, далее эти данные ИС будет обрабатывать, чтобы на выходе получить Расписание в удобной форме для студента, преподавателя и инженера кафедры.

В роли механизма будут выступать, составители расписания будут вводить расписание в ИС.

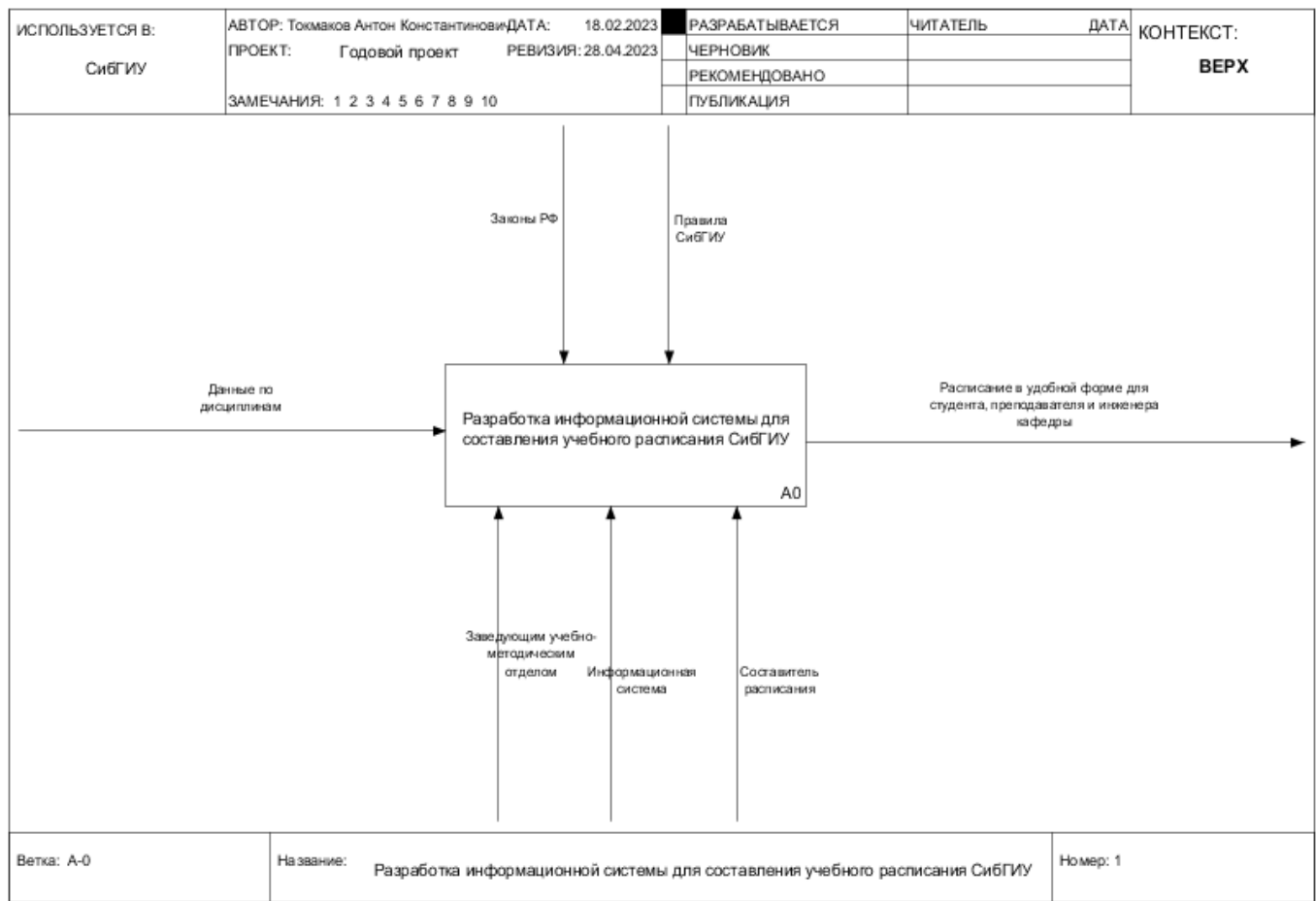


Рисунок 3 – первый уровень диаграммы A0

На рисунке 4 показана контекстная диаграмма первого уровня. На ней выделили 3 основных процесса: Ручной ввод расписания, Проверка ошибок ввода и Формирование удобного отражения.

Управление приходит в блоки Ручной ввод и Формирование удобного отображения.

Вход приходит в первый процесс Ручной ввод, далее первый процесс идет на вход в процесс Проверка ошибок ввода, на выходе же мы получаем либо подсказки к ручным корректировкам, то есть делаем управление над первым процессом, а после

успешной проверки переход в следующий процесс, Формирование удобного отображения, где происходит поэтапное формирование удобного отображения для разных пользователей [8].

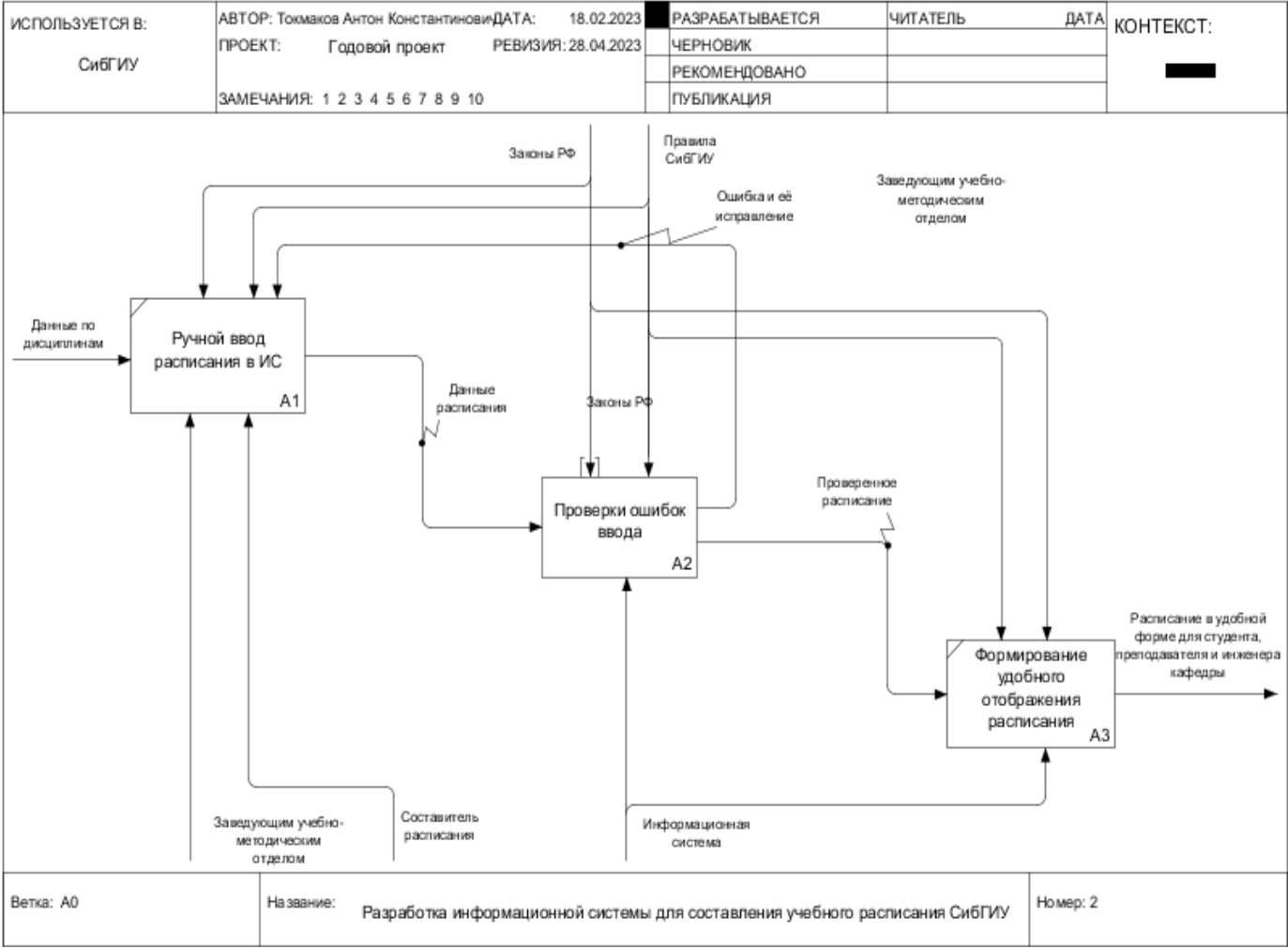


Рисунок 4 – Первый уровень контекстной диаграммы A1

На рисунке 5 декомпозируем процесс Проверок ошибок ввода, на котором выделили 3 процесса. Таких как проверка накладок преподавателей и групп, проверка оснащения кабинетов и запись в БД. От процесса проверка накладок преподавателей и групп идет стрелка к процессу проверка оснащения кабинетов, под названием проверенные данные. От того ж процесса идет стрелка на выход из блока, ошибка и её исправление, таким же образом идет стрелки от процесса оснащения кабинетов. После записи в базу данных на выход идет проверенное расписание.

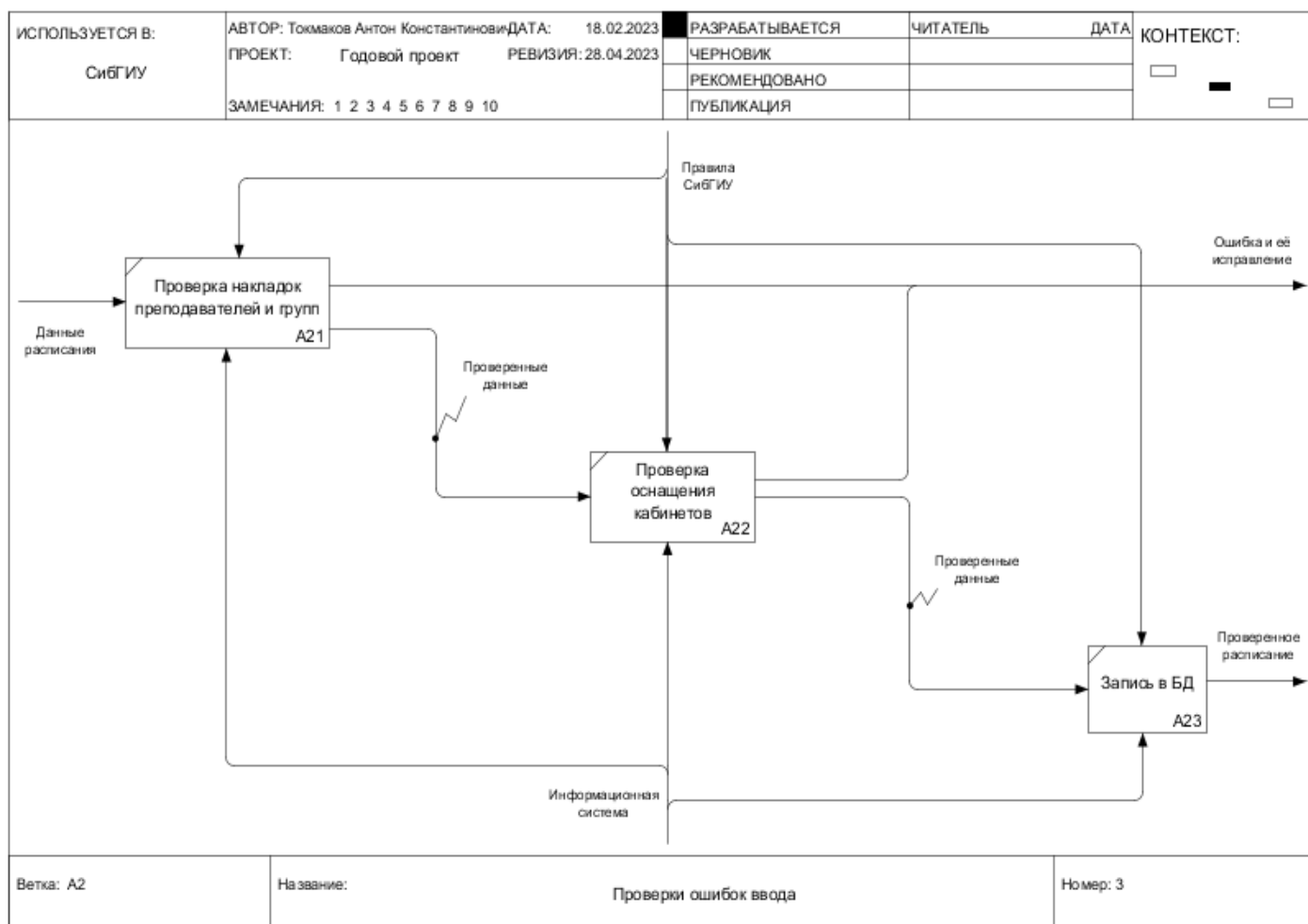


Рисунок 5 – Второй уровень декомпозиции A2

## Схема базы данных

На рисунке 6 представлена схема базы данных, которая будет использоваться в проекте. Результирующей сущностью будет Информация о паре, данная сущность будет хранить всю подробную информацию о конкретной паре, такую как Группа преподавателей, Дисциплина, День недели (Понедельник/Вторник/Среда/Четверг/Пятница/Суббота), Номер пары (Первая/Вторая и тд), Кабинет, Группу, Тип дисциплины (Лекция/Практика).

Так же можно заметить большое количество справочников, таких как День недели, Номер пары, Неделя (Четная/Нечетная), Тип дисциплин (Лекция/Практика), Ссылка на курс в Moodle, Тип дисциплины и Кабинет, это сделано для обеспечения

защиты целостности, хранящейся в базе данных информации. Что бы снизить риск ошибки пользователей при работе с БД.

В схеме БД представлена сущность Преподаватель. Данная сущность служит для уникального определения преподавателя с его определенным качеством, у Преподавателя может быть Контактные данные, такие как номер телефона или почта, он может указать любое количество контактных данных, а также может не указывать их вовсе.

Каждый преподаватель должен относиться к определенной кафедре для того, чтобы в дальнейшем можно было собрать разную аналитику по данным кафедр. Преподаватель может относиться только к одной кафедре, так как в предметной области неважно к какой точно кафедре относиться преподаватель.

На каждое назначение на пару может указываться несколько преподавателей, это регулирует сущность под название Преподаватель – Назначение на пару. Что бы показать, что преподаватели будут вместе вести пару, достаточно указать у них одинаковый номер группы.

Так же для уникального определения Группы студентов, создана сущность с названием Группа студентов. Она определяет качества каждой группы, такие как Краткое наименование группы, Полное наименование группы, Количество студентов в группе и Специальность, последний атрибут будет подтягиваться из справочника Специальность, который в свою очередь будет зависеть от справочника Кафедра, для снижения рисков ошибок пользователей и сохранение информационной целостности БД.

Так как на Назначение на пару может быть несколько групп, в случае если это лекция, для этого случая создана связывающая сущность Группа – Назначение на пару. Она может по определенному коду группы хранить несколько групп, которые в дальнейшем и будут принадлежать назначению на пару.

Рассмотрим сущность Дисциплина, у неё имеются такие атрибуты как Кафедра, информация подтягивается из справочника, Наименование пары, Сокр. Наименование пары, ссылка на курс в Moodle, количество практик и лекций в семестре, для отслеживания, когда должны закончиться пары этого предмета.

Так же есть сущность под названием Кафедра, она нужна для уникального определения кафедр с их отличительными особенностями, такие как Наименование и краткое наименование, Номер кабинета, Профессия и Контактные данные, которых может быть неограниченное количество.

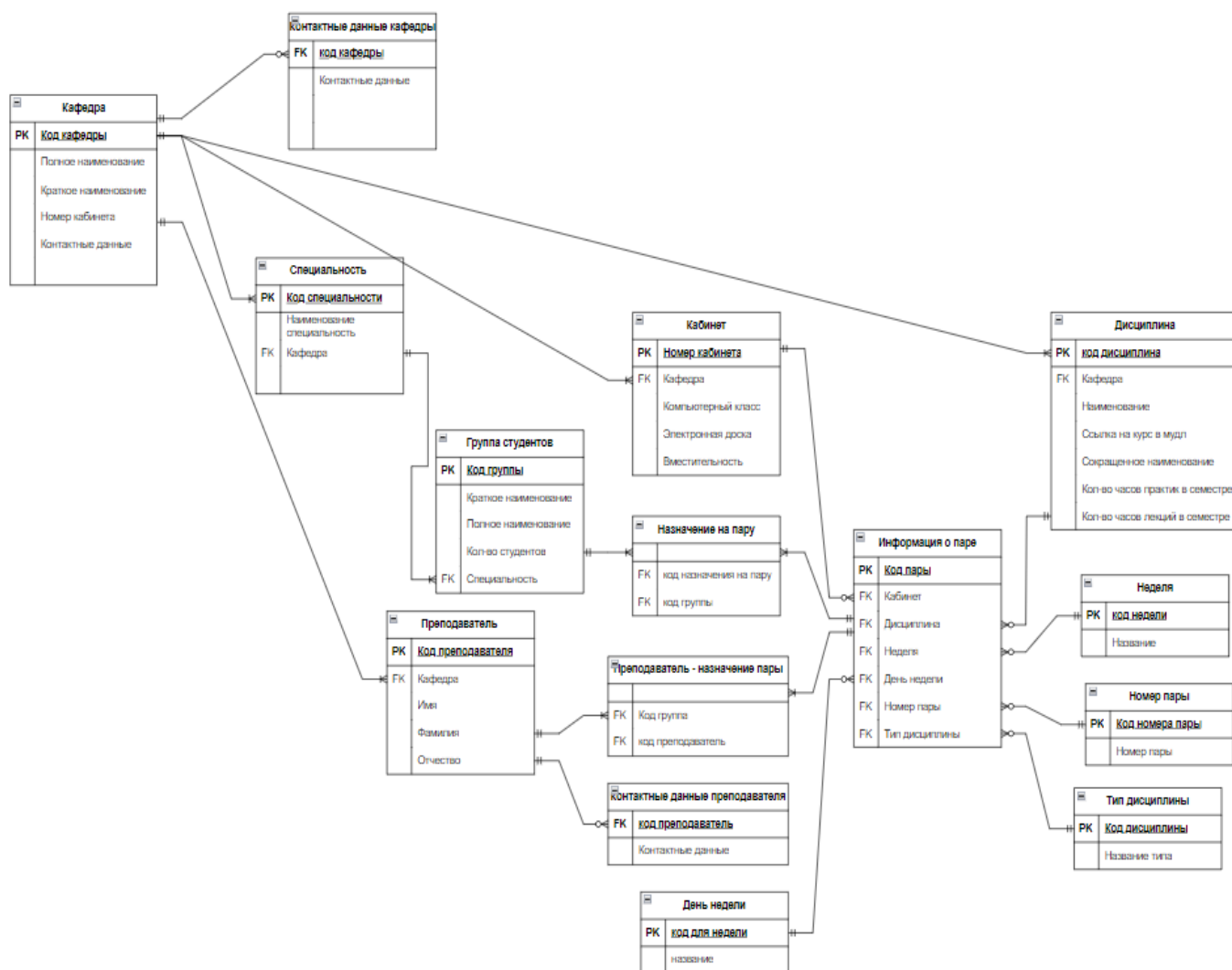


Рисунок 6 – схема базы данных

### 2.2.1.2 Алгоритмическое обеспечение

На рисунке 7, представлен алгоритм работы приложения, сначала происходит авторизация пользователя, которая рассмотрена в отдельном блоке, далее если пользователь выбирает страницу Расписание, то он видит расписание студента, преподавателя и инженера кафедры. Если будет выбрана страница экзамены, то пользователю отобразятся экзамены. Если же пользователь выберет вкладку составитель расписания, то его перебросят на другой алгоритм, рассмотренный на рисунке 19.

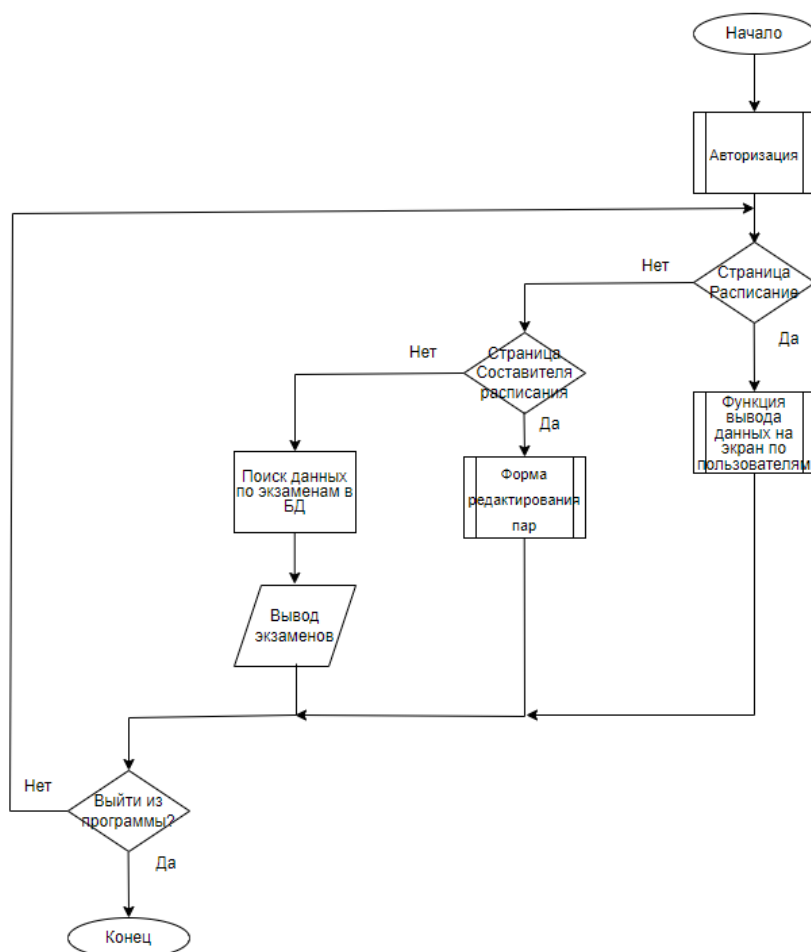


Рисунок 7 – алгоритм работы приложения

На рисунке 8 изображен алгоритм редактирования пар, который начинается с проверки на составителя расписания, если пользователь таковым не является, то дальше он пройти не сможет [9].



Если пользователь является составителем расписания, то ему надо выбрать неделю, группу и карточку дня, далее эта карточка дня передает данные окну редактирования, и составитель расписания может изменять данные карточки из окна редактирования.

Если пользователь захочет выйти ему будет предложено остаться, если пользователь все-таки нажмет выйти, программа его предупредит, что данные будут потеряны, если пользователь подтвердит это, то карточка закроется без сохранения, если же пользователь откажется от потери данных, то ему будет предложено сохранить данные, если он согласится данные запишутся в БД, иначе будет продолжено редактирование карточек.

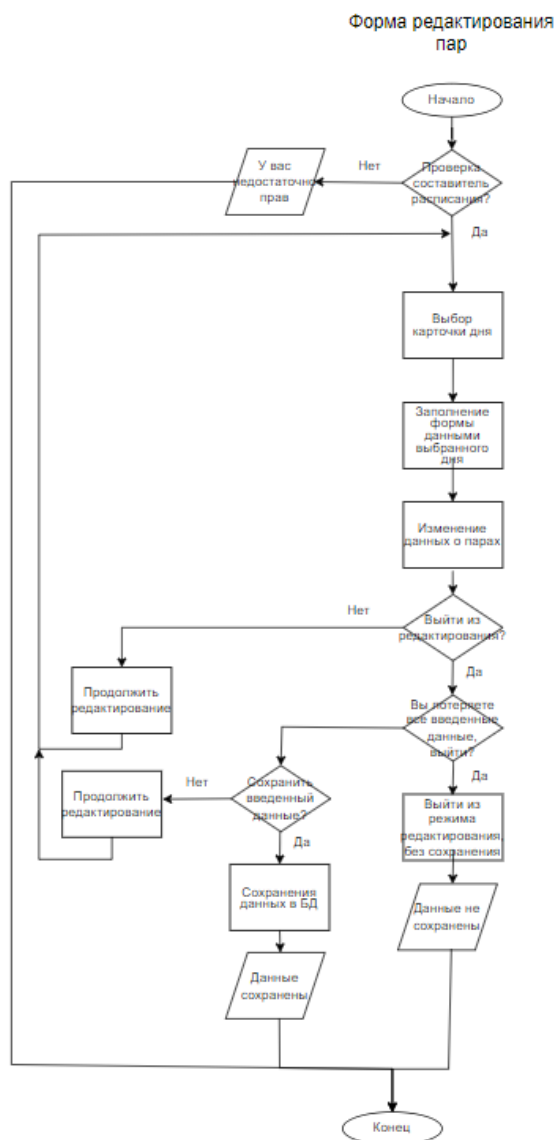


Рисунок 8 – алгоритм редактирования пар

На рисунке 9 функция выводит на экран расписание пользователя если у него правильно заполнен профиль. И передает информацию из профиля в фильтры.

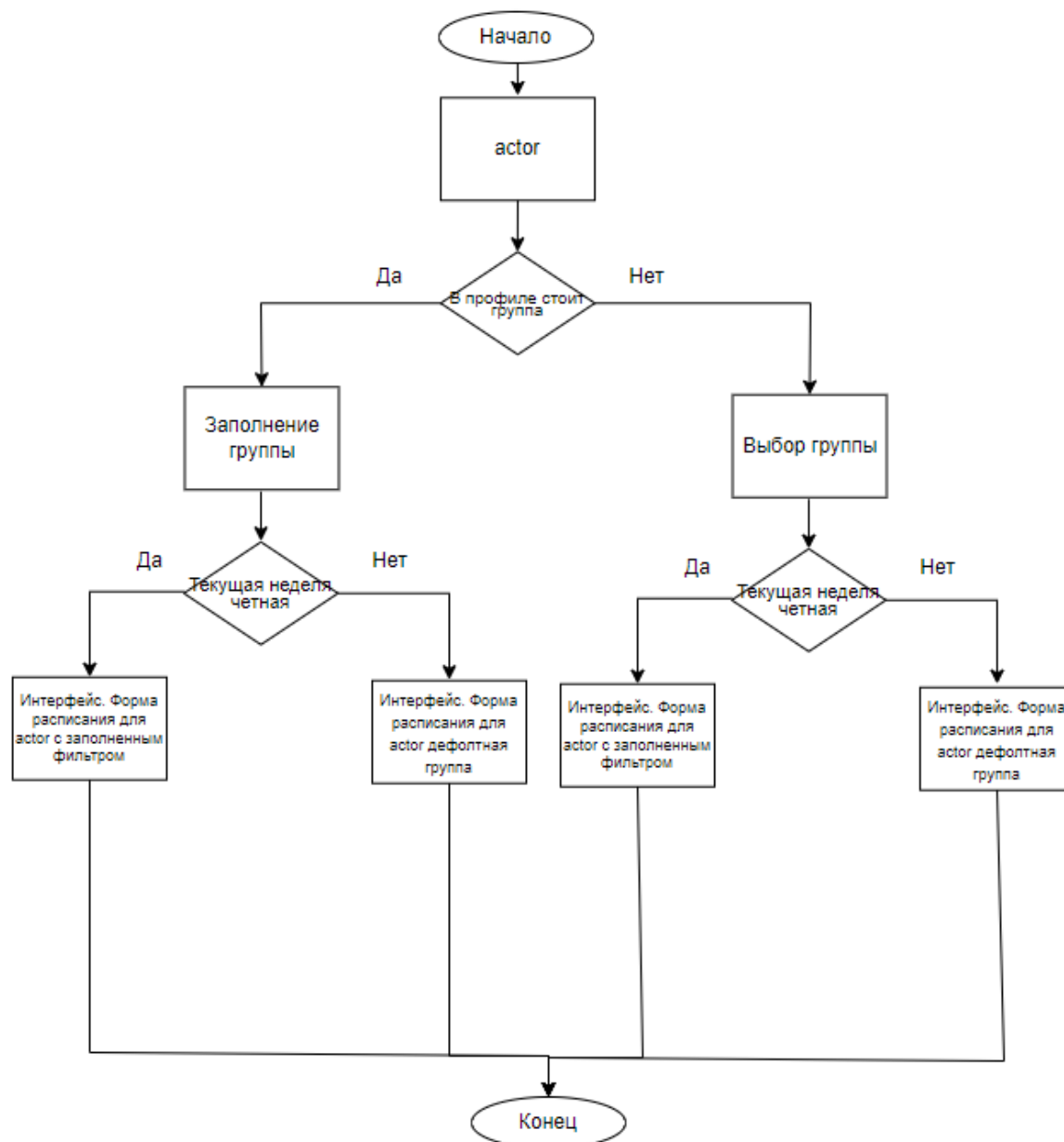


Рисунок 9 – функция вывода данных

А в рисунке 10 представлен блок авторизации. Данные на вход принимается логин и пароль, потом этот пароль сверяется с паролем и логинов в БД в случае успешной проверки, можно будет пройти далее и увидеть интерфейс либо пользователя, либо составителя расписания, либо администратора, иначе программа попросит повторить вход в программу.

## Авторизация

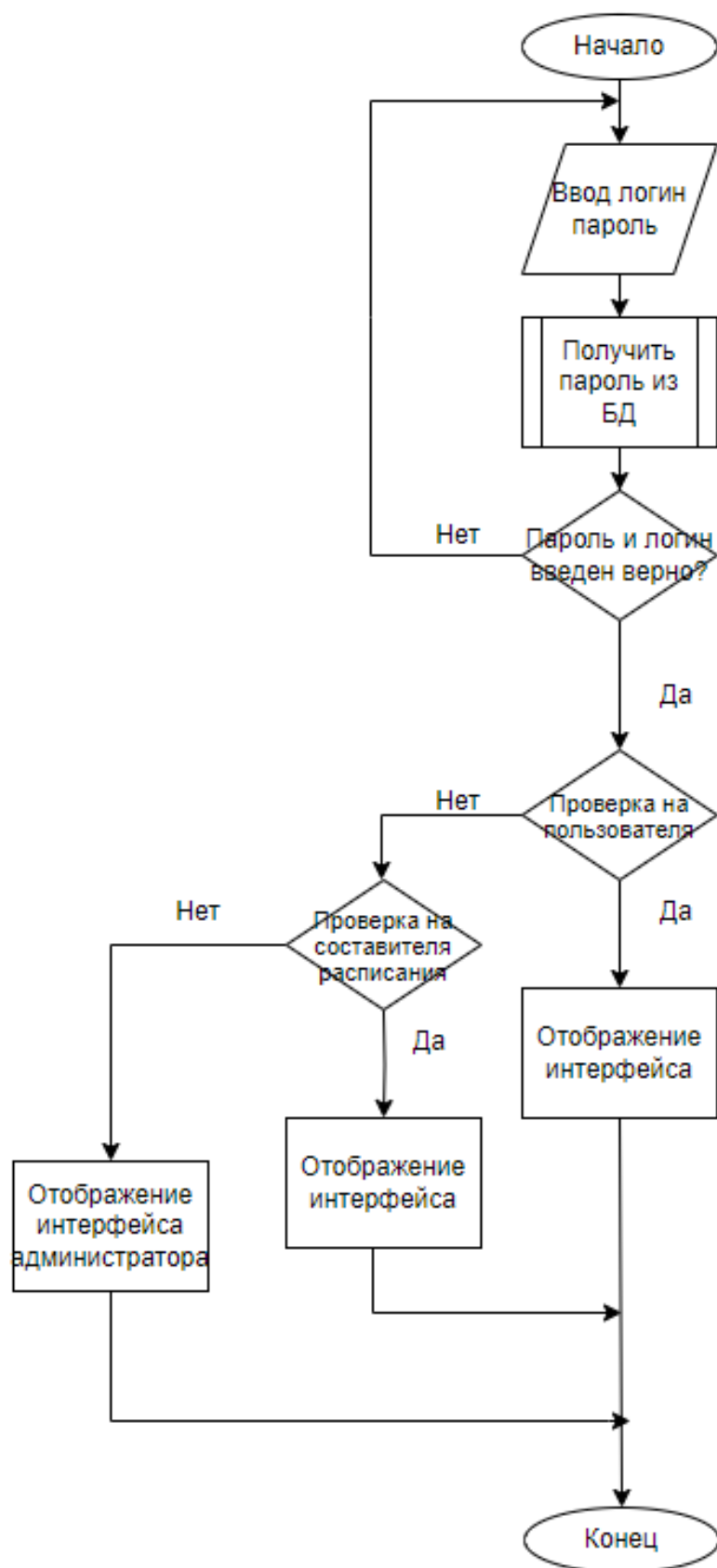


Рисунок 10 – проверка авторизации

### 2.2.1.3 Программное обеспечение

#### Эскизный интерфейс

Акцентными цветами в эскизном дизайне приложения был синий и желтый цвета. На рисунке 11 показан экран составителя расписания, который включает в себя слева меню, в нем имеется разный функционал приложения. В центре изображено само расписание по дням на группу, которую выбирает составитель расписания.

Справа можно увидеть основной элемент составителя расписания, это панель для редактирования. На ней расписаны пары на день, преподаватель, кабинет, а также сама пара. Снизу этой панели можно выбрать группу и неделю (четная, не четная). В нижней панели можно выбрать прошлые семестры и посмотреть их расписание.

Расписание

Меню

Настройки

Экзамены

Преподаватели

Выход

Группа

Понедельник

Вторник

Среда

Четверг

Пятница

Суббота

ПН ВТ СР ЧТ ПТ СБ

Первая пара

Вторая пара

Третья пара

Четвёртая пара

Пятая пара

Шестая пара

Преподаватель

Кабинет

Группа

Чётная

Первый семестр

Второй семестр

Третий семестр

Четвёртый семестр

Пятый семестр

Рисунок 11 – эскизный интерфейс приложения составителя расписания

На рисунке 12 показан экран пользователя. На нем отображается информация о парах, так же при наведении на определенную пару, она отображает преподавателя, который ведет пару.

Слева есть меню с функционалом приложения, присутствуют такие кнопки как меню, настройки, экзамены, которая переходит на вкладку экзамены



Рисунок 12 – эскизный интерфейс, экран пользователя

На рисунке 13 изображен экран входа, на котором надо ввести логин и пароль, ниже находится кнопка входа.

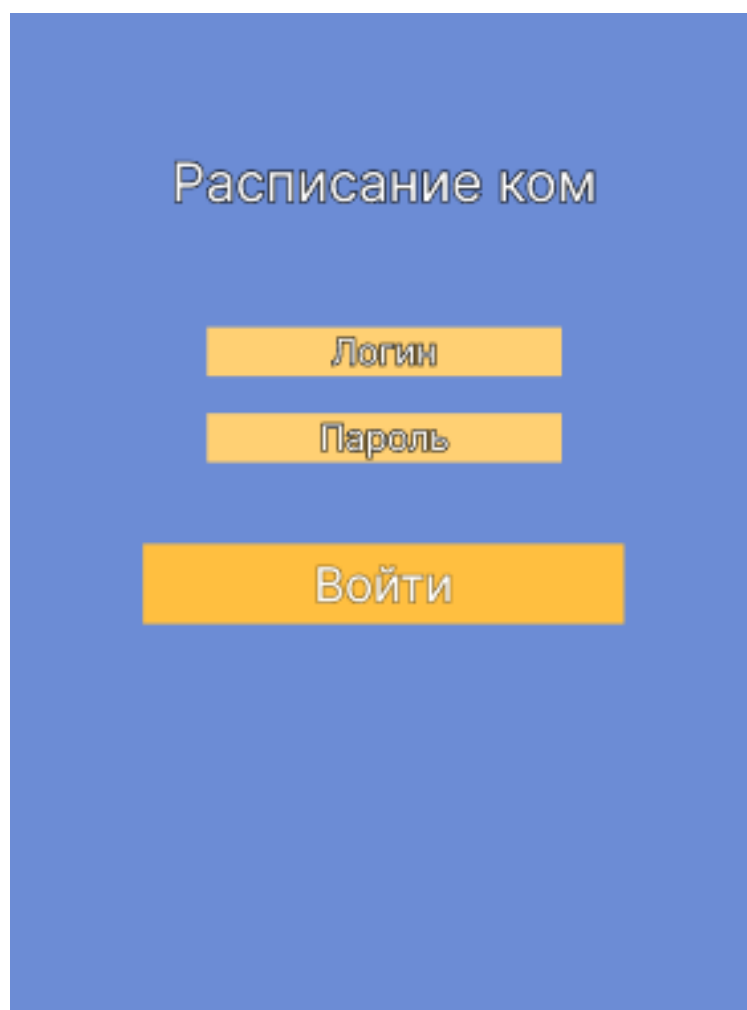


Рисунок 13 – эскизный интерфейс, авторизация

### **Финальный интерфейс**

Финальный дизайн приложения по составлению расписания.

На рисунке 15 изображен финальный интерфейс приложения, основные цвета, которые используются это темно-синий и зелёный.

Так же можно заметить карточки дней, углы которых закруглены для эстетичности, рядом с названием пары можно заметить, небольшую кнопку, при нажатии которой будет открываться Moodle на странице с этим курсом.

В самом верху видно экраны приложения, по которым можно переключаться, такие как, главная, расписание, настройки, составление расписания, а также кнопка входа.

Слева в углу можно заметить логотип СибГИУ, чтобы показать принадлежность проекта к университету.

Чуть ниже бара навигации находится бар инструментов. На нем будут находиться печать пар, смена недели, выбор группы, а также выбор отображения расписания (студента, преподавателя, инженера кафедры).

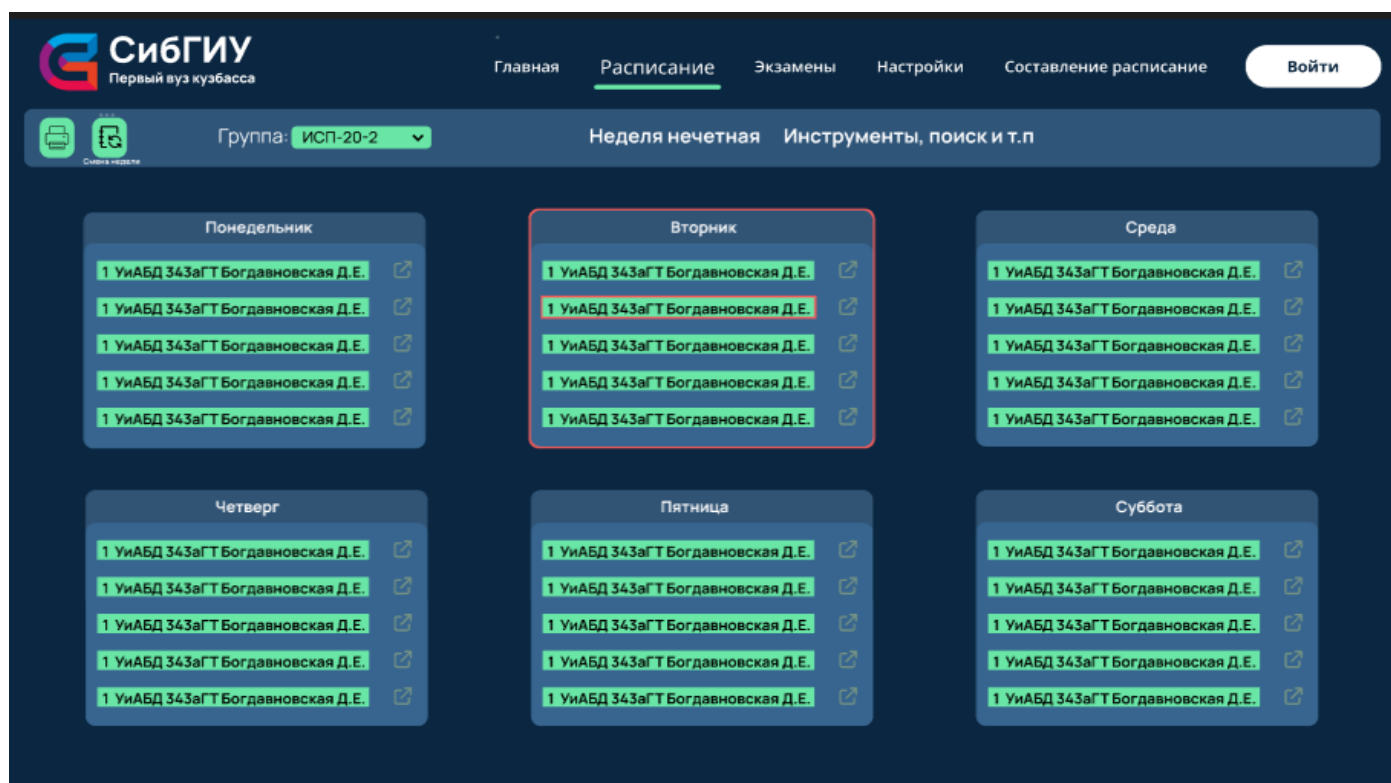


Рисунок 15 – финальный интерфейс, экран пользователя

На рисунке 16 показан экран составителя расписания, слева так же, как и у пользователя карточки дни с параметрами, на которых отображается наименование пары, кабинет в котором будет проходить пара, преподаватели, которые будут вести пару, а также ссылка на курс в Moodle, а слева основной элемент составителя расписания, панель редактирования расписания.

На панели редактирования находится наименование пары, преподаватели, которые ведут эту пару, чтобы их добавить надо нажать на кнопку, стоящую рядом, тогда откроется форма, для выбора преподавателей, кабинет в котором будет проходить эта пара, а также отметка какой это тип пар (лекция, практика).

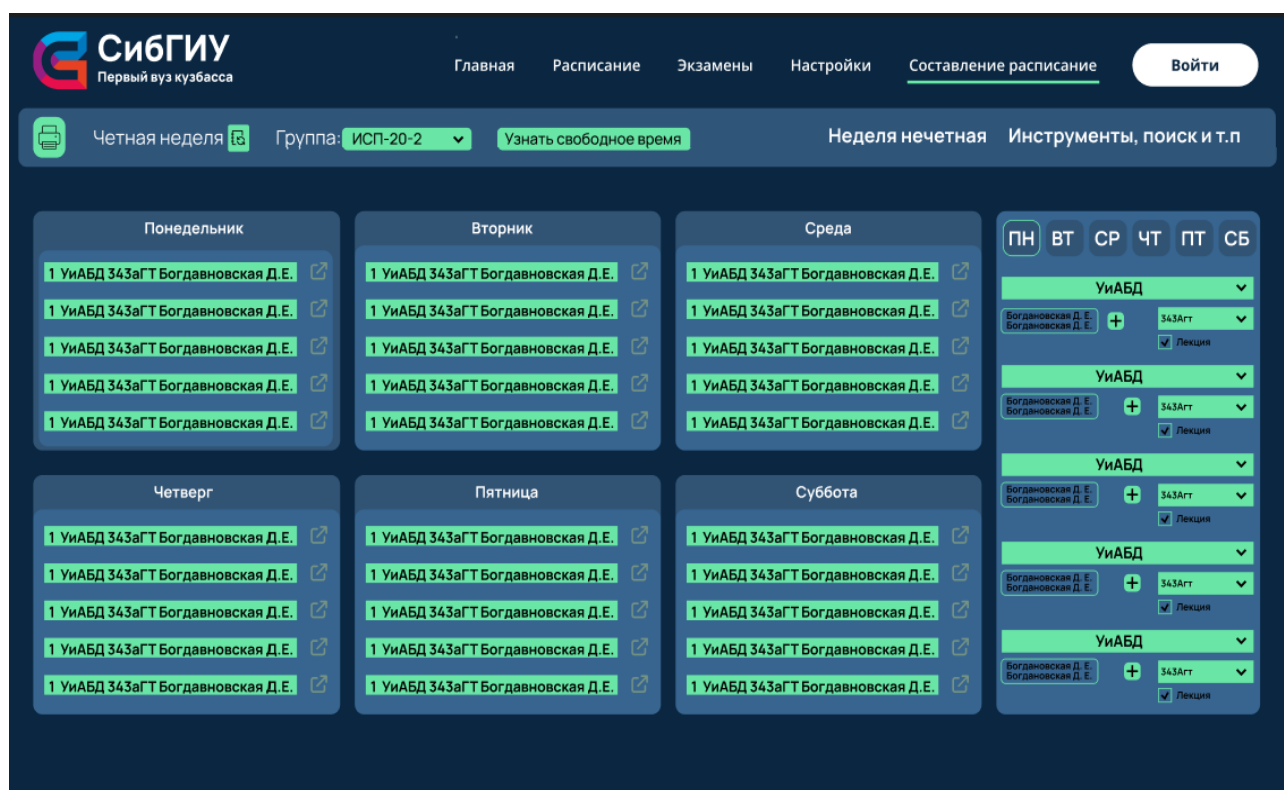


Рисунок 16 – финальный интерфейс, экран составителя приложения

На рисунке 17 изображена форма авторизации, на ней отображаются 2 текстовых поля, в которые можно ввести логин и пароль созданной учетной записи, а далее нажать кнопку войти, что бы произошел вход в систему.

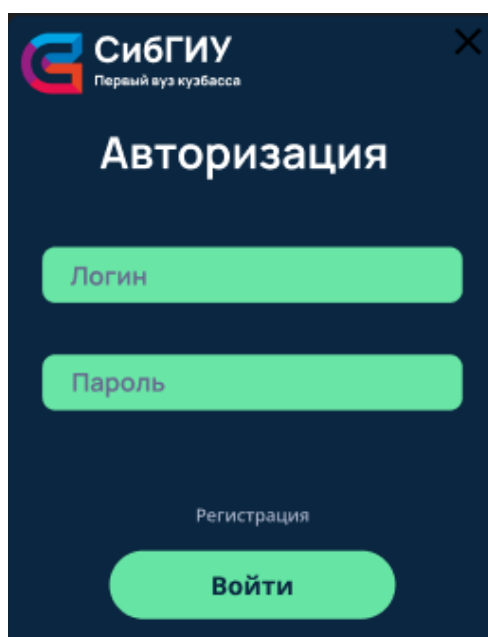


Рисунок 17 – финальный интерфейс, экран авторизации



## Итоговая реализация интерфейса разработанной системы

На рисунке 18 показан итоговый интерфейс программы для просмотра расписания пользователям.

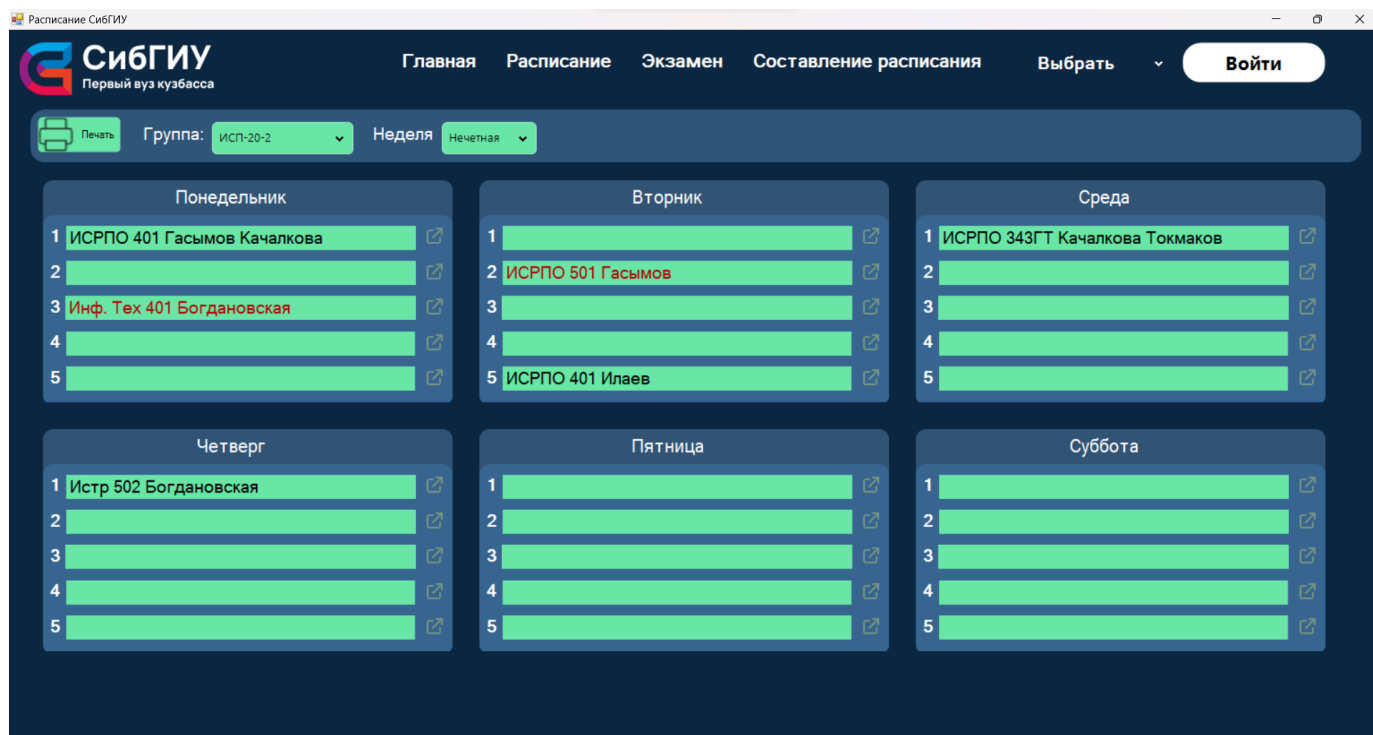


Рисунок 18 – интерфейс готовой программы, экран пользователя

На рисунке 19 представлена форма составления расписания, на неё может зайти только пользователь с правами составителя расписания. На верхней панели видны выбор пар и недель, по центру размещена информация о парах по карточкам дней в которой отображается наименование пары, номер кабинета, а также преподаватели, которые ведут пару.

Справа основной элемент составителя расписания, по двойному нажатию на любой из карточки дня данные из карточки передаются на панель редактирования, с неё составитель заполняет или изменяет данные о паре.

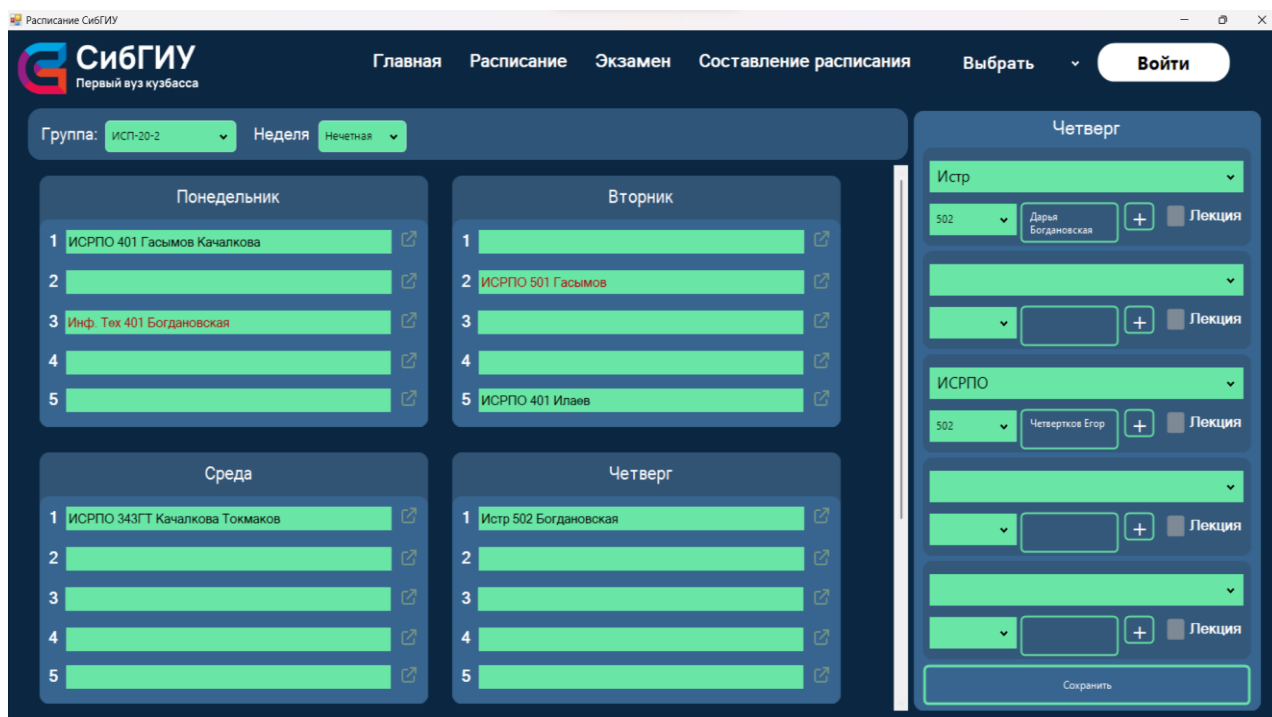


Рисунок 19 – интерфейс готовой программы, экран составления расписания

На рисунке 20 показан интерфейс формы справочника преподаватели. На панели есть три по нажатию на кнопку добавления будет всплывать новое окно, в котором можно добавить преподавателя, для изменения информации о преподавателе следует нажать кнопку изменить, а для удаления кнопку удалить.

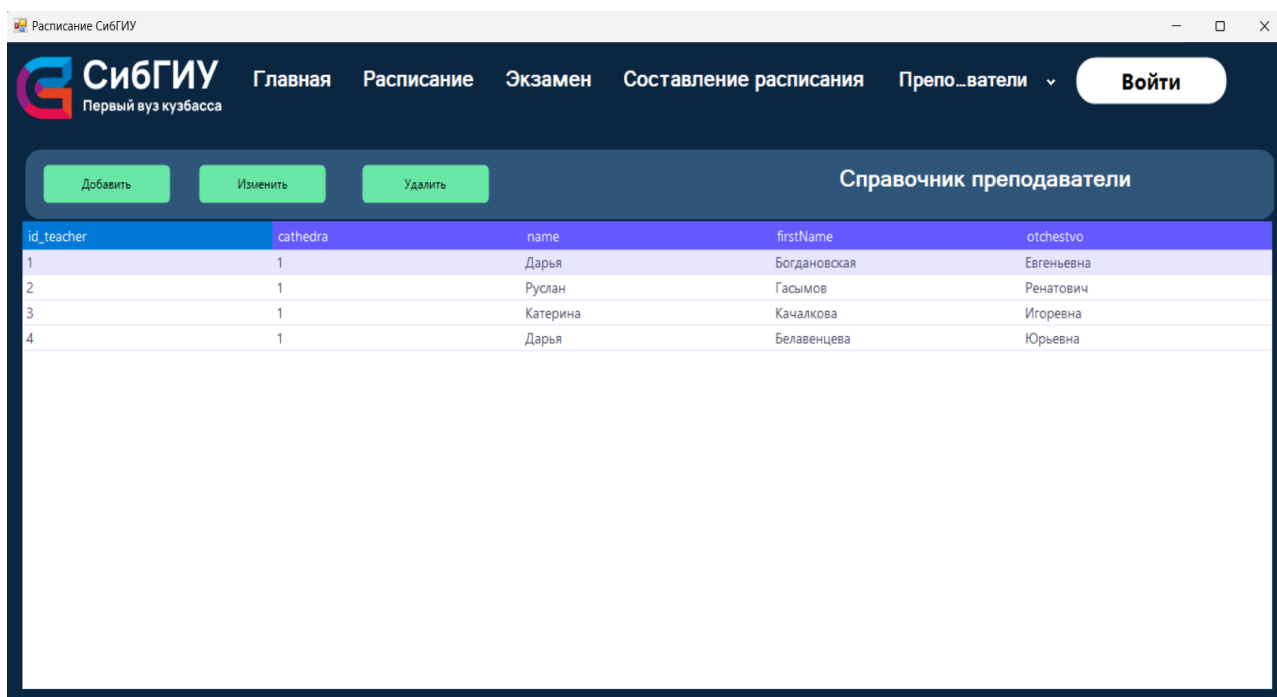


Рисунок 20 – интерфейс готовой программы, экран справочник преподаватели

#### 2.2.1.4 Лингвистическое обеспечение

В информационной системе для составления учебного расписания СибГИУ и сайта для удобного просмотра расписания применяется один естественный язык – русский. В процессе жизненного цикла программного обеспечения, количество естественных языков не увеличивается.

Для вывода текста на естественных языках был применен шрифт семейства Times New Roman. На различных экранных формах кегль шрифта и прочие настройки отличаются. По умолчанию используются следующие настройки: кегль – 12, межстрочный интервал – 1, выравнивание – по левому краю, абзацный отступ – 1.

Текст хранится в бинарных файлах и файлах исходного кода.

При проектировании программного продукта для описания структуры, информационных потоков и другого использовались специальные языки: UML, блок-схемы.

Программный продукт построен с использованием языка программирования C# и интегрированной среды разработки Microsoft Visual Studio 2019. СУБД, использованная при разработке Microsoft SQL Server.

#### 2.2.1.5 Техническое обеспечение

Минимальные и рекомендованные системные требования клиентского компьютера для запуска информационной системы для составления учебного расписания СибГИУ и сайта для удобного просмотра расписания представлена в таблице 9.

Таблица 9 – Требование к ИС

Минимальные системные требования	
Операционная система	Windows 10
Процессор	Intel i3 7100
Свободное место	не менее 128 Gb

*Продолжение таблицы 9*

Объем оперативной памяти	не менее 4 Gb
Видеокарта	Nvidia GeForce G210
Рекомендуемые системные требования	
Операционная система	Windows 10
Процессор	Intel i3 9100
Свободное место	256 Gb
Объем оперативной памяти	16 Gb
Видеокарта	Nvidia GeForce 1050

Для реализации информационной системы, также требуется серверный компьютер со следующими минимальными или рекомендованными системными требованиями, представленными в таблице 10.

Таблица 10 – Требование к серверу

Минимальные системные требования	
Операционная система	Windows 10
Процессор	Intel i3 9100
Свободное место	512 Гб
Объем оперативной памяти	8 Гб
Видеокарта	Nvidia GeForce G210
Рекомендованные системные требования	
Операционная система	Windows 10
Процессор	Intel i3 10100
Свободное место	1 Тб
Объем оперативной памяти	16 Гб
Видеокарта	Nvidia GeForce GT1030 или AMD Radeon R7

На рисунке 21 изображено Аппаратное обеспечение информационной системы составления расписания, на ней можно увидеть персональный компьютер, который отправляет запросы на выборку из базы данных на сервер, сервер принимает запрос и передает его к системе управления базами данных, которая в свою очередь делает выборку из базы данных по заданному запросу и отправляет ответ обратно на клиентскую часть приложения.

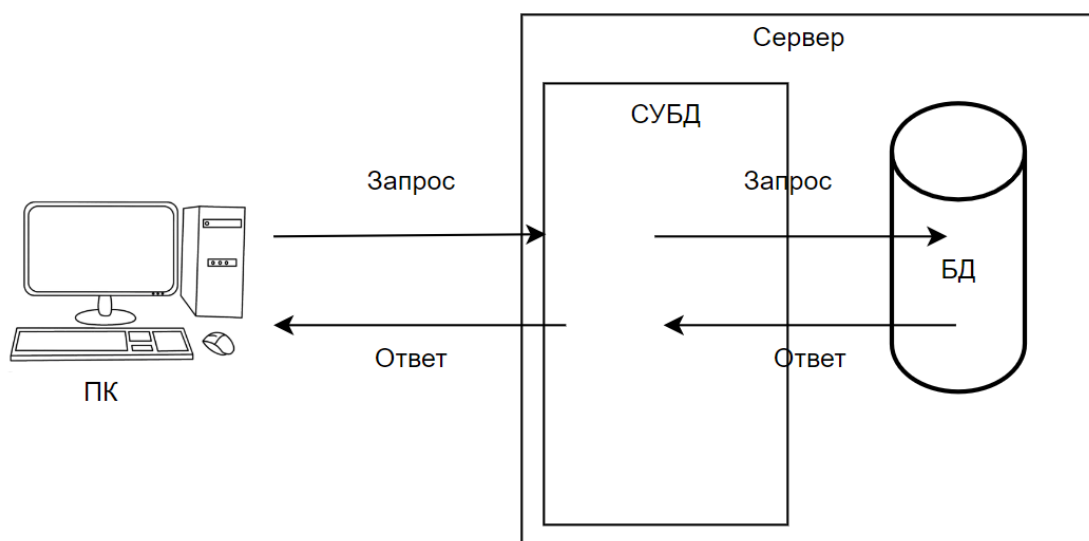


Рисунок 21 – схема взаимодействия аппаратных компонентов

### 2.2.2 Анализ и доработка

В течении всего процесса разработки информационной системы было изменено и доработано множество аспектов.

База данных была изменена множество раз, чтобы сделать ее более актуальной для предметной области. Эти изменения включали изменение сущностей и атрибутов, которые больше не соответствовали требованиям. Также был изменен интерфейс, в частности, форма редактирования пар. Изначально это было отдельное всплывающее окно, которое перекрывало остальные пары. Чтобы улучшить пользовательский опыт, было решено добавить его справа от карточек дней, тем самым редактирование дней не перекрывало его содержимое.

## Заключение

Исходя из проделанной работы, данный проект можно считать актуальным решением существующих на данный момент проблем с составлением расписания в образовательном учреждении.

Проект «Разработка информационной системы для составления учебного расписания СибГИУ» был успешно разработан командой начинающих разработчиков в указанные сроки, а также подготовлена вся документация для конечного продукта, такая как устав проекта, техническое задание, оценка рисков, прототипирование интерфейсов, описание информационного обеспечения, обоснование выбора программного продукта и техническое обеспечение проекта [10].

Разработка информационной системы и сайта позволяет использовать современные технологии для создания учебного расписания по средствам удобного и функционального интерфейса, а также обеспечивает высокую надежность и безопасность системы. А также решает важный вопрос

Внедрение данного комплекса программного обеспечения в СибГИУ позволит значительно упростить процесс составления и использования учебного расписания, а также повысить эффективность работы преподавателей и студентов.

Таким образом, проект «Разработка информационной системы для составления учебного расписания СибГИУ» является важным и перспективным проектом, который может значительно улучшить качество образования в СибГИУ, так как от того, насколько удачно составлено расписание, зависит качество обучения, экономическая эффективность обучения, комфортность учебы студентов и работы преподавательского состава. Разработка данной информационной системы и сайта, является важным шагом в развитии систем автоматизации в образовательных учреждениях, что позволит значительно улучшить качество образования и повысить эффективность работы учебных заведений.

Кроме того, данный проект может быть применен не только в СибГИУ, но и в других образовательных учреждениях, у которых возникают проблемы с составлением расписания. Команда готова к дальнейшей работе над развитием и

усовершенствованием данного проекта, чтобы обеспечить максимальную удобность и эффективность его использования. Мы уверены, что информационная система для составления учебного расписания и сайт для его удобного просмотра будет полезным и необходимым инструментом для любой образовательной организации.

## Библиографический список

- 1) Ивлев В. А., Попова Т. В. ABIS. Информационные системы на основе действий; 1С-Публишинг - М., 2017. - 248 с.
- 2) Сенкевич Г. Е. Информационная система малого предприятия «с нуля». Самое необходимое; БХВ-Петербург - М., 2020. - 400 с. 26. Хубаев Георгий Николаевич Информатика. Информационные системы. Информационные технологии. Тестирование. Подготовка к интернет-экзамену; Феникс - М., 2015. - 117 с.
- 3) Криницкий, Н.А.; Миронов, Г.А.; Фролов, Г.Д. Автоматизированные информационные системы; Наука - М., 2017. - 382 с.
- 4) Троелсен Э. Язык программирования C# 5.0 и платформа .NET 4.5 /
- 5) Стоунз PostgreSQL. Основы / Стоунз, Мэттью Ричард; , Нейл. - М.: СПб: Символ-Плюс, 2002. - 640 с.
- 6) Информационные системы и технологии управления: Учебник / Под ред. Г.А. Титоренко. - М.: Юнити, 2013. - 591 с.
- 7) Персианов Вячеслав Венедиктович; Логвинова Екатерина Ивановна Информационные Системы (Специальность Доу). Учебно-Методическое Пособие; РГГУ - Москва, 2021. - 197 с.
- 8) Беленькая М. Н., Малиновский С. Т., Яковенко Н. В. Администрирование в информационных системах; Горячая линия - Телеком - М., 2021. - 400 с.
- 9) Информационные технологии и вычислительные системы: Обработка информации и анализ данных. Программная инженерия. Математическое моделирование. Прикладные аспекты информатики / Под ред. С.В. Емельянова. - М.: Ленанд, 2015. - 104 с.
- 10) Информационные технологии и вычислительные системы / Под ред. С.В. Емельянова. - М.: Ленанд, 2006. - 128 с.