

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»
(Н И У « Б е л Г У »)**

ИНСТИТУТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ И ЕСТЕСТВЕННЫХ НАУК
КАФЕДРА ИНФОРМАЦИОННЫХ И РОБОТОТЕХНИЧЕСКИХ СИСТЕМ

**РАЗРАБОТКА ПОДСИСТЕМЫ ПОДБОРА ЛЕКАРСТВЕННЫХ
СРЕДСТВ**

Выпускная квалификационная работа студента
обучающегося по направления подготовки 09.03.02 Информационные
системы и технологии
очной формы обучения, группы 07001407
Калиниченко Алины Сергеевны

Научный руководитель
доцент, к.т.н.
Гахов Р.П.

БЕЛГОРОД 2018

РЕФЕРАТ

Разработка подсистемы подбора лекарственных средств – Калиниченко Алина Сергеевна, выпускная квалификационная работа бакалавра, Белгород, Белгородский государственный национальный исследовательский университет (НИУ «БелГУ»), количество страниц 52, включая приложения 61, количество рисунков 31, количество таблиц 1, количество использованных источников 24.

КЛЮЧЕВЫЕ СЛОВА: мобильное приложение, медицинская информационная система, фармакологический справочник, календарь приема лекарств.

ОБЪЕКТ ИССЛЕДОВАНИЯ: технологии разработки мобильного приложения.

ПРЕДМЕТОМ ИССЛЕДОВАНИЯ: фармакологический справочник и календарь приема лекарств.

ЦЕЛЬ РАБОТЫ: повышение эффективности применения лекарственных препаратов путем разработки и внедрения интерактивных программных средств в виде мобильного приложения.

ЗАДАЧИ ИССЛЕДОВАНИЯ: рассмотреть уже существующие средства подбора лекарственных средств, провести обзор целевой аудитории, спроектировать и создать базу данных, разработать мобильное приложение с фармакологическим справочником и позволяющее назначить график приема лекарств, протестировать разработанное приложение.

ПОЛУЧЕННЫЕ РЕЗУЛЬТАТЫ: произведен анализ предметной области; определена целевая аудитория, создана база данных; разработано и протестировано мобильное приложение с фармакологическим справочником и позволяющее назначить график приема лекарств.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
1 Анализ предметной области	6
1.1 Концепция проекта	6
1.2 Обзор и анализ существующих систем подбора лекарственных средств	10
1.3 Обзор целевой аудитории.....	14
2 Разработка концептуальной модели предметной области.....	17
2.1 Разработка функциональной модели	17
2.2 Разработка инфологической модели	26
2.3 Разработка даталогической модели.....	29
3 Программная реализация.....	31
3.1 Разработка базы данных	32
3.3 Разработка и тестирование информационной системы	37
ЗАКЛЮЧЕНИЕ	50
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	51
ПРИЛОЖЕНИЕ А	53
ПРИЛОЖЕНИЕ Б.....	57

ВВЕДЕНИЕ

По данным Всемирной организации здравоохранения средняя продолжительность жизни человека в мире за последние 70 лет возросла на 23 года и согласно отчету о мировой статистике здоровья за 2017 год (World Health Statistics 2017: Monitoring health for the SDGs) составляет 71,4 год на 2015 год [1]. Это связано с большим количеством различных факторов, примерами могут быть: улучшение качества питания, улучшение жилищных условий, уменьшение количества вооруженных конфликтов, повышение уровня гигиены да и в целом культуры населения и самое важное - улучшение уровня медицины.

Люди стали больше следить за своим здоровьем, делать прививки, лечить болезни на ранних стадиях развития, более тщательно подходить к подбору лекарств. С улучшением качества образования, развитием технологий и появлением интернета общая грамотность населения касательно медицины возросла в сотни раз. Человек основываясь на ранних симптомах уже может узнать примерную область заболевания и к какому врачу ему стоит обратиться. Люди стали более ответственно относиться к своему здоровью и лечению болезней, стали появляться достойные медицинские учреждения и образованные, компетентные врачи. Это все выводит личностный контроль за своим здоровьем на абсолютно новый уровень. Мы стали внимательней относиться даже к самым малым признакам заболевания того или иного характера и подобная тенденция не может не развиваться, что и послужило идеей для проекта о котором пойдет речь далее.

Человек имеет доступ к различного рода лекарствам и в случае болезни может сам проходить курс лечения, назначенный врачом. А так как препаратов на данный момент бесчисленное множество в каждой из отраслей медицины, например от одной болезни может быть много разных лекарств, было бы не плохо их структурировать и поместить в одно приложение. Дабы достичь наилучшего из возможных результатов лечения очень важно соблюдать

точный график приема подходящих лекарств. Разработка данного мобильного приложения поможет человеку структурировать курс своего лечения и не забывать принимать лекарства в нужный момент с помощью уведомлений. Так же в данном приложении человек сможет найти всю интересующую его информацию касательно лекарственных средств систематизированных по терапевтическому применению согласно анатомо-терапевтическо-химической классификации. В первую очередь это носит информационный характер, однако сейчас остро стоит вопрос самолечения. Люди не всегда обращаются за помощью к врачу, а стараются определить диагноз и подобрать лекарства самостоятельно, в случаях с явными признаками обычной простуды или же головной болью это не приведет к необратим последствиям, но очень часто встречаются случаи когда неправильное использование медикаментов приводит к еще большим проблемам со здоровьем и последствия могут быть самыми разными. Например постоянное самолечение болеутоляющими может скрывать симптомы, которые указывают на проблемы более тяжелые, и требующие немедленного и серьезного лечения. Для того что бы обезопасить человека от приема не подходящих лекарственных средств в базу была включена классификация лекарств по фармакологическому действию, описаны противопоказания, взаимодействие с другими лекарствами, а также предупреждения о возможных последствиях при длительном приеме лекарств без консультации у врача.

1 Анализ предметной области

1.1 Концепция проекта

В данной работе рассматриваются вопросы проектирования и разработки мобильного приложения [2], предназначенного для подбора лекарственных средств и назначения графика приема различных медикаментов. Система должна хранить подробные сведения о лекарствах, о фармакологических действиях и группах медикаментов, АТХ классификации, а также данные пользователя, включающие личную информацию и график приема лекарств (какие медикаменты, количество повторений в день, периодичность и др.) [3].

Цель работы – повышение эффективности применения лекарственных препаратов путем разработки и внедрения интерактивных программных средств в виде мобильного приложения.

Система должна иметь понятный и многофункциональный интерфейс, который будет приятный и интересный пользователю с точки зрения эстетики, интерактивности и полезности.

В связи с этим были определены следующие задачи необходимые для достижения цели проекта:

- изучить современный рынок подобных мобильных приложений;
- рассмотреть уже существующие средства подбора лекарственных средств;
- провести обзор целевой аудитории;
- спроектировать и создать базу данных;
- разработать мобильное приложение с фармакологическим справочником и позволяющее назначить график приема лекарств;
- протестировать разработанное приложение.

В ходе проектирования концепции проекта было выделено 3 главных этапа реализации:

- обзор рынка;
- создание базы данных;
- разработка программного продукта.

Каждый этап разбит на более конкретные работы, полный перечень работ с целевыми показателями представлен в таблице 1.

Таблица 1 – Этапы реализации проекта

Этапы	Работы	Целевые показатели
Обзор рынка		Оценка данного проекта относительно других вариантов и информация о целевой аудитории
	Изучение существующих вариантов решения данного вопроса	Плюсы и минусы аналогов, сравнительная характеристика с данным проектом
	Обзор целевой аудитории	Возраст аудитории и основные потребности
Создание базы данных		База данных лекарств, на основе которой будет создаваться программа
	Ознакомление с материалами	Изученная классификация медикаментов и основные характеристики
	Проектирование модели базы данных	Схематическое отображение базы данных
	Наполнение базы данных	Ввод данных о лекарственных средствах

Продолжение таблицы 1

Этапы	Работы	Целевые показатели
Разработка программно-продукта		ПО работающее на основе БД, реализующее основные функции ведения МЧК
	Описание информационной системы	Сформированный список функций, которые будут выполняться приложением
	Формирование системы поиска	Поиск по ключевым параметрам (торговое название, действующее вещество, терапевтическое применение, код АТХ, фармакологическое действие, фармакологическая группа),
	Реализация отображения информации о медикаменте	Отображение полной информации о медикаменте (инструкция по применению препарата)
	Создание формы регистрации и авторизации пользователя	Ввод личных данных и проверка пользователя при входе в приложение
	Написание алгоритма добавления лекарства и назначение определенного графика приема	Возможность добавлять лекарства в список принимаемых и редактировать график приема (какие медикаменты, количество повторений в день, периодичность и др.)
	Создание алгоритма оповещения	Оповещение пользователя о необходимости принять лекарства

Продолжение таблицы 1

Этапы	Работы	Целевые показатели
Разработка программного продукта	Создание формы оценки качества курса лечения	Возможность оценить эффективность принимаемого медикамента
	Работа над дизайном	Стилистически оформленное приложение
	Тестирование приложения	Корректная работа программы

Объектом данной работы является технологии разработки мобильного приложения, а предметом – фармакологический справочник и календарь приема лекарств.

Информационная система позволяет структурировать и автоматизировать процесс контролирующий график приема лекарств, это способствует повышению качества управленческой информации, что приводит к повышению качества принятия решений по подбору лекарств [4]. Также поддерживает аналитические функции, т.е. операции по анализу данных, полученных из разных источников, в данном случае это оценка эффективности лекарственного средства от пользователя.

Основными функциями данной информационной системы являются:

- предоставление пользователю полной информации о медикаменте;
- классификация медикаментов по анатомо-терапевтическо-химическому признаку;
- классификация медикаментов по фармакологическому действию;
- просмотр аналогов медикаментов;
- поиск лекарственного средства по торговому названию и действующему веществу;

- регистрация пользователя в системе;
- ведения графика приема лекарств;
- уведомление пользователя о необходимости принять препарат;
- настройка уведомлений и периодичности приема медикаментов;
- настройка уведомлений и системы под конкретного пользователя в зависимости от личных данных пользователя.

1.2 Обзор и анализ существующих систем подбора лекарственных средств

Анализ конкурентов рынка – это обязательный первый шаг перед реализацией собственного проекта [5].

Основная цель анализа рынка и обзора аналоговых мобильных приложений – получить достаточную информацию о сильных и слабых сторонах конкурента, о его намерениях и возможностях для повышения эффективности собственных разработок [6].

В открытом доступе было найдено не так много приложений обеспечивающих подбор лекарственных средств и ведение графика приема медикаментов. Больше всего существует приложений типа справочник лекарственных средств и болезней, намного меньше создано приложений, выполняющих функцию напоминания о приеме медикаментов, приложений включающих в себе и справочник и отслеживание графика приема медикаментов еще меньше. Еще одна группа подобных приложений – это медицинские помощники для управления запасами лекарств в аптечке. Обзор самых популярных аналоговых приложений представлен ниже.

Первое найденное приложение было создано разработчиком Tyrant Ventures LLC и называется – «Мои таблетки – напоминания Lite» [7]. Это приложение представляет собой программу напоминающую о приеме лекарств для того, чтобы пользователи не забывали выпить таблетки, прописанные врачом. В программе реализованы настройка различных схем и

курсов, повторные напоминания и архив курсов.

Существует две версии данного приложения: платная и бесплатная. При скачивании бесплатной версии, в течении первых 15 дней пользователю доступна полнофункциональная версия. Далее будет доступен только один активный курс, то есть напоминания будут работать только для одного лекарства. Функционирующую в полном объеме программу можно приобрести за 119 рублей. Это исключает из возможных пользователей определённое количество людей.

Одним из минусов данного приложения является то, что при добавлении лекарств в список принимаемых, поиск медикамента возможен только по торговому названию, то есть пользователь не может просмотреть аналоговые лекарственные средства используя действующее вещество. Также в базе данных хранятся только названия таблеток и нет возможности получить более подробную информацию о лекарственном средстве.

После изучения отзывов пользователей, можно сделать вывод, что основной проблемой, с которой люди сталкиваются в процессе эксплуатации данного мобильного приложения – это несрабатывание сигнала, если экран телефона не активен. Уведомление о приеме медикамента – основная функция данного приложения и ее некорректная работа неприемлема, так это может повлечь за собой негативные последствия для пользователей, которым очень важно соблюдать график и принимать определённые медикаменты в определенное время.

Следующим рассмотренным приложением является «Напоминания о таблетках – MyTherapy» [8], созданием которого занимается компания по разработке программного обеспечения в Мюнхене, Германия – Smartpatient gmbh. Оно включает в себя ряд таких достоинств, как напоминания о приёме лекарств, отслеживание перепатов настроения, журнал состояния здоровья, график изменения веса и многое другое.

Приложение MyTherapy полностью бесплатно и доступно любому пользователю. Однако, по данным с официального сайта компании, в

настоящее время пользователи из России могут испытывать трудности с доступом к онлайн-сервисам в следствии блокировки Роскомнадзором ip-адресов.

Еще один минус для пользователей из Российской Федерации это отсутствие базы данных лекарств нашего региона, в приложении добавлена информация о медикаментах, но только для стран европейского региона.

Поэтому данное приложение, хоть и популярно на территории Российской Федерации, является не актуальным для жителей данного региона так как не несет в себе никакой информативности.

Еще одно популярное приложение представлено продавцом IT Primus и имеющее название – «Медицинский помощник» [9]. Это хорошее приложение для управления запасами лекарств в домашней аптечке. В приложение интегрирован справочник лекарств. Функция справочника интегрирована с аптечкой, приложение подсказывает правильное название препарата при его добавлении. Также реализована возможность узнать предположительную причину недуга по симптомам, препараты для лечения и врачи к которым следует обратиться за консультацией той или иной проблемы.

Данное приложение может служить хорошим примером реализации справочника лекарственных средств и симптомов. «Медицинский помощник» достаточно информативное приложение.

После изучения рынка и рассмотрения аналоговых мобильных приложений, можно сделать вывод, что подобных программ существует не так много, чего нельзя сказать о спросе на них, люди проявляют интерес к таким продуктам. Однако в основном качество исполнения не удовлетворят пользователей, на это влияет и некорректная работа приложения, и его внешний вид, сложности в ходе пользования, связанные с не интуитивным интерфейсом, недостаточность функций и т.п. Одно из отличий данного проекта от уже существующих вариантов это совмещение в себе фармакологического справочника и календаря графика приема медикаментов, что является очень удобным для пользователей.

Большим плюсом разрабатываемого мобильного приложения является распространение на бесплатной основе, которое планируется провести с помощью социальных программ, которые сейчас являются востребованными. Ни одно из просмотренных приложение не включает настройку уведомлений и системы в целом под каждого пользователя в соответствии с введенными личными данными. Например если девушка укажет что ожидает ребенка, медикаменты которые противопоказаны при беременности, будут помечены отличительным знаком или же если пользователь укажет, что водит машину, при приеме медикамента запрещающего вождение, появиться уведомительное сообщение. Также пользователь может указать имеющиеся аллергии на определенные действующие вещества в медикаментах. И при наполнении календаря приема лекарственных средств, медикаменты содержащие данное действующее вещество будут исключены из списка, а при поиске в фармакологическом справочнике данные препараты будут помечены отличительным знаком.

Еще одним плюсом данного приложения является отслеживание эффективности каждого препарата. В ходе пользования программой человек может дать оценку эффективности принимаемого медикамента, все данные будут собираться в общую статистику и выводиться для каждого лекарственного средства. Исходя из этого можно сделать вывод, что реализация данного проекта имеет место быть.

Еще одним решением такого вопроса является так называемые таймеры для таблеток [10]. Они представляют собой емкость в которую помещаются таблетки, которая сама напомнит звуковым сигналом о необходимости приёма лекарства. Однако это не всегда удобно, так как это еще они предмет который постоянно должен находиться рядом, а телефон и так всегда под рукой. Пример такого таймера представлен на рисунке 1.1.



Рисунок 1.1 – Пример таймера для таблеток

В современном мире существует много различных вариаций таких таймеров – это и наручные устройства, напоминающие часы, и контейнеры различных форм с большим количеством отделов для разных таблеток и емкость в виде брелока.

1.3 Обзор целевой аудитории

Целевая аудитория [11] – это те люди, которым адресуется информация, на которых направлены действия программы, проекта, кампании в целом. С точки зрения маркетинга – это совокупность реальных и потенциальных потребителей товара или услуги, которые готовы изменить свои предпочтения в пользу данного товара или услуги под воздействием определенных мер.

Изучение и анализ целевой аудитории помогает определить организации и людей, которые играют важную роль в успешной реализации всего задуманного, сэкономить ресурсы и сделать максимально эффективной

программу, проект, компанию [12].

По данным на 2017 год 55% людей по всему миру пользуются смартфонами, а это больше 4х миллиардов человек. Возраст целевой аудитории, которая пользуется мобильными приложениями, варьируется от 12 до 53 лет, это говорит о том, что люди абсолютно разных возрастных категорий используют в своей повседневной жизни различные мобильные приложения. Проанализировав эти данные можно предположить какой должна быть целевая аудитория описанной предметной области в данной дипломной работе. Так как, к сожалению, болезни поражают людей разных возрастных категорий, данное приложение может быть полезно как детям так и людям пожилого возраста.

Рассмотрим детальнее основные функции данного приложения тем самым выведем его целевую аудиторию. Возьмем один из полезных аспектов приложения – напоминание о приеме лекарственных средств. Смартфоны являются нашими постоянными спутниками, кому как не им поручить задание по напоминанию о приеме лекарств, ведь в суматохе обыденных дел очень просто забыть принять нужный препарат тем самым нарушив режим, что в разной степени негативно повлияет на результат лечения. Следовательно, неважно, кто будет использовать данное мобильное приложение: ученик, который находится в школе на занятиях, пожилой учитель или среднестатистический взрослый человек, звуковой сигнал оповестит пользователя о необходимости принять препарат в данный момент, со всеми необходимыми рекомендациями по применению.

Следующим плюсом является полная информация о лекарстве, включая информацию о побочных эффектах и информацию об аналогичных лекарствах, которая находится в приложении. Она будет актуальна уже для людей более старшего возраста. Например, у родителей возникли вопросы о курсе приема лекарства их ребенка. Они смогут найти и почитать полную информацию дома, без всяких бумажных инструкций и выписок, которые имеют свойство теряться. Хорошо то, что фармакологический справочник

доступен не только зарегистрированным в системе пользователям, но и всем желающим, потому, что не всем необходимо вести график приема лекарств, кто-то не захочет создавать аккаунт, или же человеку будет полезен только справочник. В результате можно сделать вывод, что целевая аудитория данного приложения люди от 12 до 50 лет (возраст людей пользующихся смартфонами) как мужского так и женского пола.

Основные потребности пользователей [12] это простой и многофункциональный интерфейс, что позволит человеку на пользовательском уровне разобраться в такой сложной отрасли как фармацевтика, и достаточное количество достоверной информации, а описания к лекарствам позволят без медицинских терминов понять как действует лекарство, при каких симптомах его обычно прописывают и как лучше его принимать. Очень важно сделать программу интуитивной, что бы использование не вызывало дискомфорт и непонимание.

Удобная поисковая система позволит быстро находить требуемые медикаменты, а также просмотреть их аналоги. Также очень полезной является возможность просмотреть список лекарств в соответствии с терапевтическим применением.

Выводы по первому разделу

В ходе выполнения работы были поставлены задачи необходимые для достижения цели проекта и выделены этапы и конкретные работы реализации

В результате проведенной работы был выполнен первый этап реализации выпускной квалификационной работы – обзор рынка. Произведена оценка данного проекта относительно других вариантов и целевой аудитории

После обзора аналогов можно сделать вывод что разрабатываемое приложение достаточно конкурентоспособное. Были учтены все недостатки и недоработки уже имеющихся приложений, для создания максимально удобного и полезного мобильного приложения.

2 Разработка концептуальной модели предметной области

2.1 Разработка функциональной модели

Изучение любой системы предполагает создание модели системы, позволяющей произвести анализ и предсказать ее поведение в определенном диапазоне условий [13]. Функциональная модель представляет собой модель системы, которая отражает последовательность во времени отдельных операций процесса, совокупность которых обеспечивает получение заданного результата. Функциональные модели, используемые на стадии проектирования ПО, предназначены для описания функциональной структуры проектируемой системы [4]. В данном случае функциональная модель отображает принцип работы приложения, раскрывая его основные процессы (формирование базы данных, авторизация пользователя, поиск лекарственных средств и формирование графика приема лекарств). В данном разделе рассматривается графический способ описания функциональной модели.

Одним из графических способов функционального описания системы является стандарт IDEF0. Описание IDEF0 модели построено в виде иерархической пирамиды [14], в вершине которой представляется самое общее описание системы, а основание представляет собой множество более детальных описаний. При создании модели сначала необходимо изобразить самый высокий уровень – действие контекста. Наименование действия описывает систему непосредственно.

Каждый блок может иметь различные типы связанных с ним стрелок. Стрелки обозначают людей, место, вещи, понятия или события. Стрелки связывают границы диаграммы с блоками, а также действия (блоки) на диаграмме между собой. В диаграммах IDEF0 имеется четыре основных типа стрелок.

Вход блока представляет материал или информацию, которая должна быть использована или преобразована блоком, чтобы произвести продукцию

(выпуск). Стрелки входа всегда направляются в левую сторону блока. Стрелки входа необязательны, так как не все действия могут преобразовать или изменять (заменять) что-либо.

Каждый блок должен иметь по крайней мере одну стрелку контроля (управления). Управление всегда входит в вершину блока. Управление, как правило, представляется в виде правил, инструкций, политики компании, процедур или стандартов. Оно влияет на деятельность без фактического преобразования чего-либо. Управление может также использоваться для описания процедуры начала или окончания выполнения действия.

Стрелки выхода (выпуска) – это материал или информация, произведенная блоком. Каждый блок должен иметь по крайней мере одну стрелку выхода (выпуска). Процессы, которые не производят продукции (выпуска), лучше не моделировать вообще.

Механизмы исполнения – это те ресурсы, которые обеспечивают выполнение действия. В качестве механизма исполнения могут быть рассмотрены персонал компании, машины или оборудование, которые обеспечивают выполнение деятельности. Стрелка механизма может отсутствовать, если определено, что это не важно для работы блока.

Для построения контекстной диаграммы информационной системы использовалось AllFusion Process Modeler r7 (BPwin) – CASE-средство для описания бизнес-процессов. Для декомпозиции используется, выше описанный, стандарт IDEF0. Декомпозиционное разложение модели используется в моделировании бизнес-процессов, для того чтобы дать более подробное описание блоков [15].

Главная функция системы – подбор лекарственных средств. На рисунке 2.1 представлена контекстная диаграмма всей системы в целом.

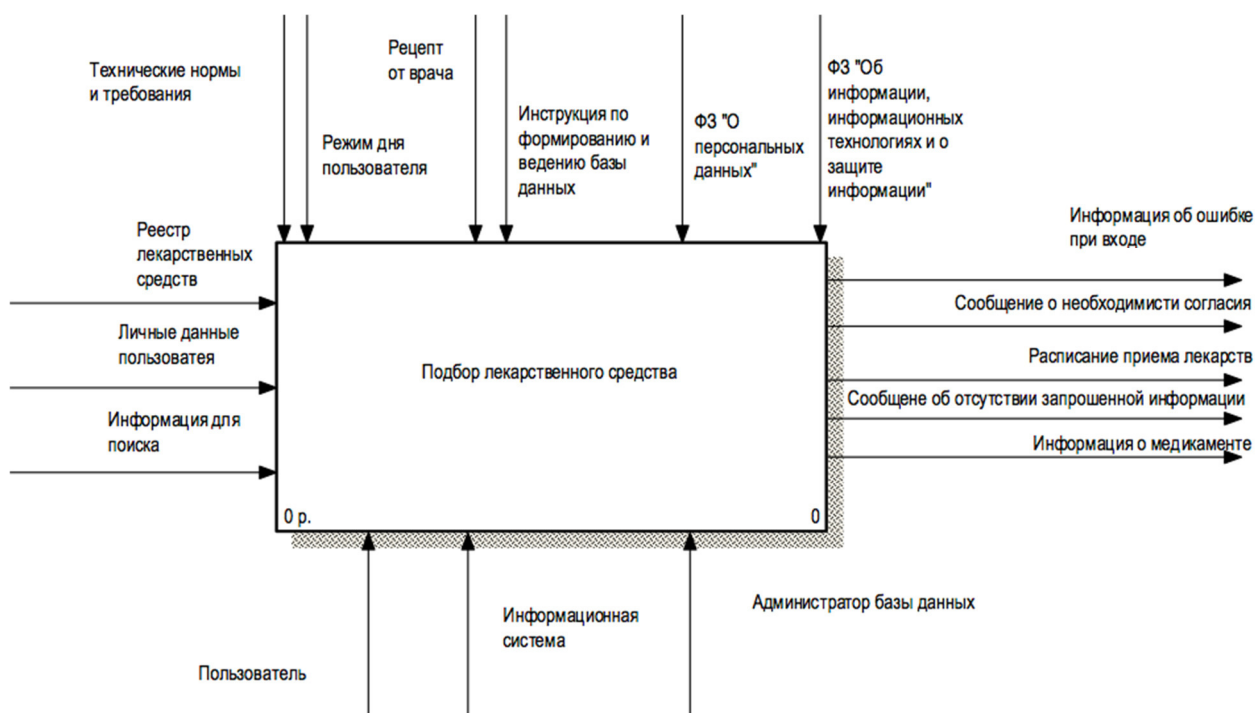


Рисунок 2.1 – Контекстная диаграмма «Подбор лекарственного средства»

На данной диаграмме к работам были подключены стрелки всех типов. Стрелки описывают взаимодействие работ и представляют собой некую информацию.:

Вход (Input) – материал или информация, которые используются или преобразуются работой для получения результата (выхода).

- реестр лекарственных средств;
- личные данные пользователя;
- информация для поиска.

Управление (Control) – правила, стратегии, процедуры или стандарты, которыми руководствуется работа. В данной диаграмме это законы.

- технические нормы и требования;
- режим дня пользователя;
- рецепт от врача;
- инструкция по формированию и ведению базы данных;
- ФЗ «О персональных данных»;
- ФЗ «Об информации, информационных технологиях и о защите информации».

Выход (Output) – материалы или информация, которые производятся работой.

- информация об ошибке при входе;
- сообщение о необходимости согласия;
- сообщение об отсутствии запрошенной информации;
- информация о медикаменте;
- расписание приема лекарств.

Механизм (Mechanism) – ресурсы, которые выполняют работу.

- администратор базы данных;
- информационная система;
- пользователь.

Произведенная декомпозиция главной задачи системы представлена на рисунке 2.2. Основные процессы, которые протекают в системе:

- формирование базы данных;
- авторизация пользователя;
- поиск лекарственного средства;
- формирование графика приема лекарств.

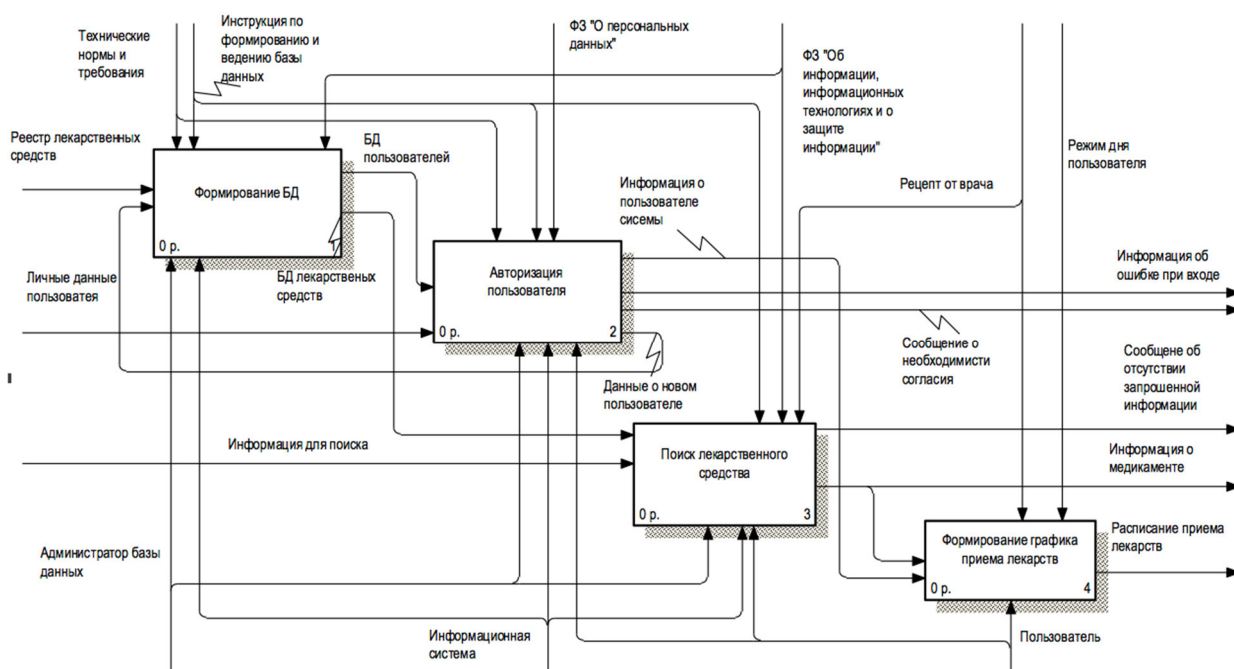


Рисунок 2.2 – Диаграмма декомпозиции «Подбор лекарственного средства»

Между работами существуют информационные потоки: между работами «Формирование базы данных» и «Авторизация пользователя» – база данных пользователей и данные о новом пользователе, между работами «Авторизация пользователя» и «Формирование графика приема лекарств» – информация о пользователе системы. База данных лекарственных средств является информационным потоком между процессами «Формирование базы данных» и «Поиск лекарственного средства». И еще один информационный поток соединяющий две работы данной диаграммы – информация о медикаменте связывает процесс «Поиск лекарственного средства» и «Формирование графика приема лекарств», он же является одним из результатов работы системы.

На рисунке 2.3 изображена декомпозиция диаграммы «Формирование базы данных».

Этот процесс включает в себя следующие задачи:

- проверка на наличие обновлений;
- формирование запроса на обновление базы данных;
- обработка данных.

Между работами существует информационные потоки – измененная информация о лекарственных средствах и запрос на обновление. Администратор базы данных следит за наличием обновлений информации в реестре лекарственных средств и при наличии таковых передает их для формирования запроса на обновление базы данных. Сформированные данные добавляются в базу и после обновления новая информация доступна пользователю.

Также этот процесс включает обработку данных пользователя. Для обновления базы данных передается информация о новых пользователях, сформированная при выполнении процесса «Добавление нового пользователя», работа которого описана ниже.

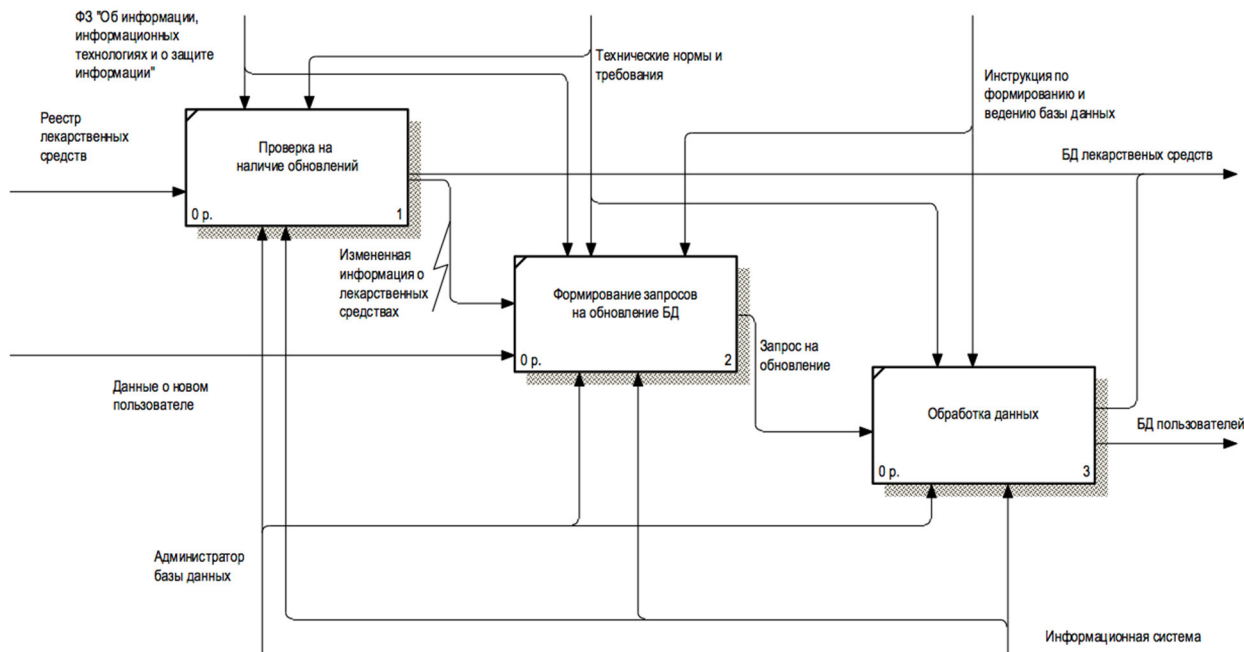


Рисунок 2.3 – Декомпозиция диаграммы «Формирование базы данных»

Декомпозиция диаграммы «Авторизация пользователя» представлена на рисунке 2.4. В этом процессе описаны 4 задачи:

- ввод пароля;
- проверка введенных данных;
- вход в систему;
- добавление нового пользователя.

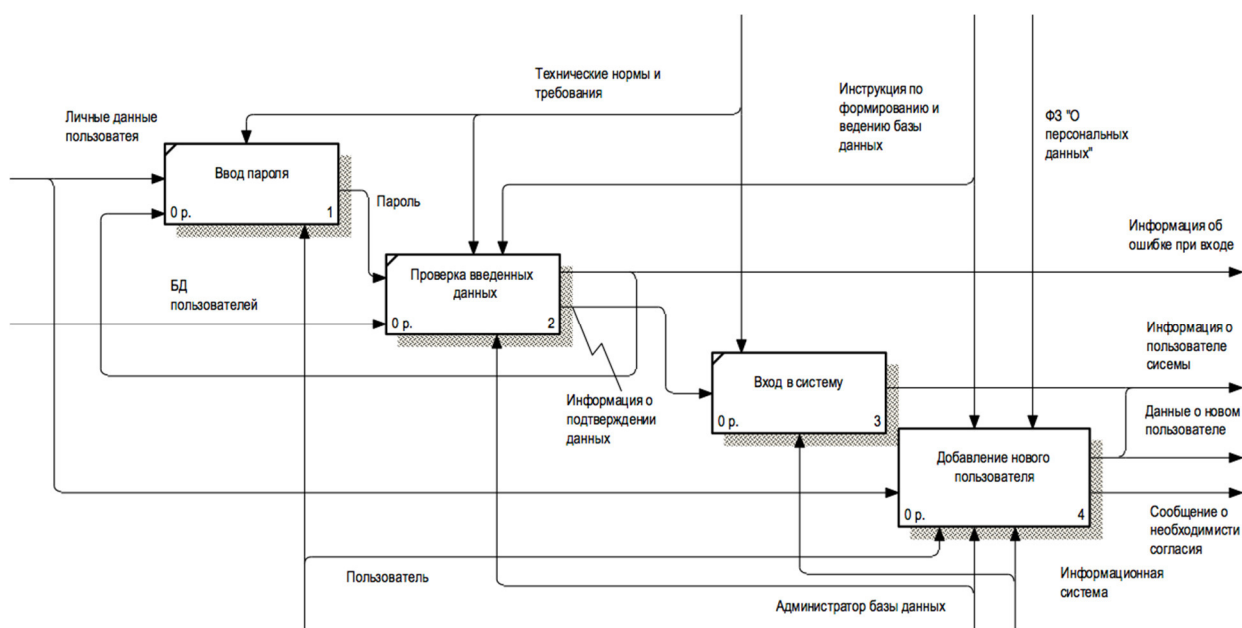


Рисунок 2.4 – Декомпозиция диаграммы «Авторизация пользователя»

Чтобы войти в систему, и получить доступ к личным данным, пользователю необходимо ввести пароль авторизации. После сравнения данных с данными базы, пользователь либо входит в систему, либо получает сообщение о том, что введенные данные не верны.

Не зарегистрированным пользователям предлагается создать аккаунт. Процесс «Добавление нового пользователя» представлен на рисунке 2.5.

Этот процесс включает в себя следующие задачи:

- ввод личных данных;
- подтверждение на обработку персональных данных;
- создание нового пользователя.

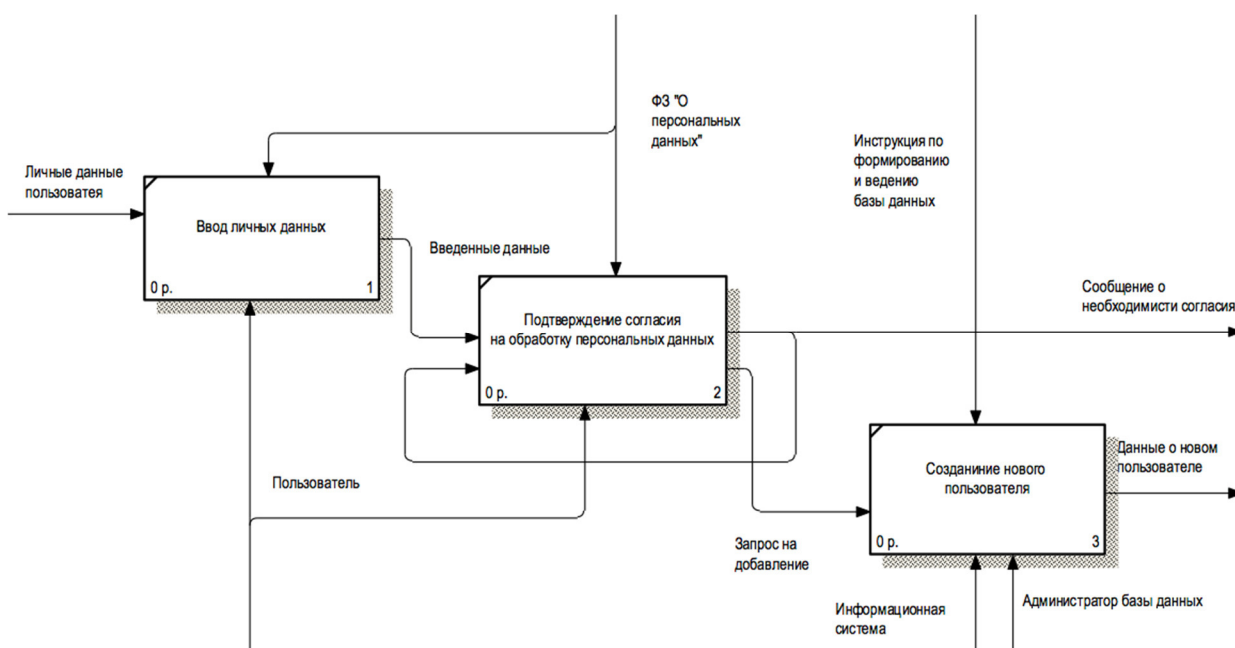


Рисунок 2.5 – Декомпозиция диаграммы «Добавление нового пользователя»

Для того что бы создать новый аккаунт в системе пользователь должен ввести необходимую личную информацию. Обязательным условием является подтверждение согласия на обработку персональных данных [16]. Одним из результатов работы системы является сообщение о необходимости согласия в случае не выполнения такового.

При выполнении всех условий формируется запрос на добавление информации о новом пользователе в базу данных, правильность выполнения и обработка данных обеспечивается администратором базы данных и программистом.

На рисунке 2.6 показана декомпозиция диаграммы «Поиск лекарственного средства». Процесс включает в себя следующие задачи:

- ввод искомым данных;
- формирование запроса поиска;
- обработка запроса;
- результат поиска.

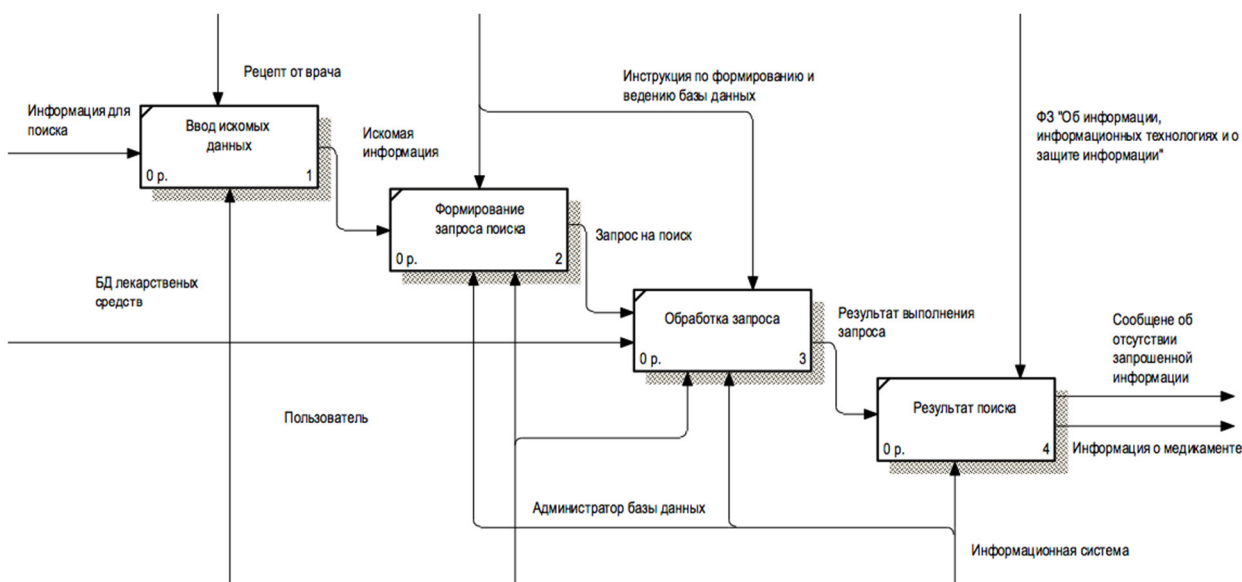


Рисунок 2.6 – Декомпозиция диаграммы «Поиск лекарственного средства»

Воспользоваться фармакологическим справочником могут не только зарегистрированные в системе пользователи, но и любой желающий, который по каким-либо причинам не ведет график приема лекарственных средств.

Пользователь вводит название лекарственного средства (также это может быть действующее вещество), далее формируется запрос поиска в базе данных. После обработки запроса, пользователь получает результат: это либо информация, удовлетворяющая условию поиска, либо сообщение о том, что поиск не дал результатов.

Декомпозиция диаграммы «Формирование графика приема лекарственных средств» представлена на рисунке 2.7. В этом процессе описаны 4 задачи:

- добавление лекарственного средства в список;
- ввод информации о периодичности приема;
- настройка уведомлений;
- подтверждение данных.

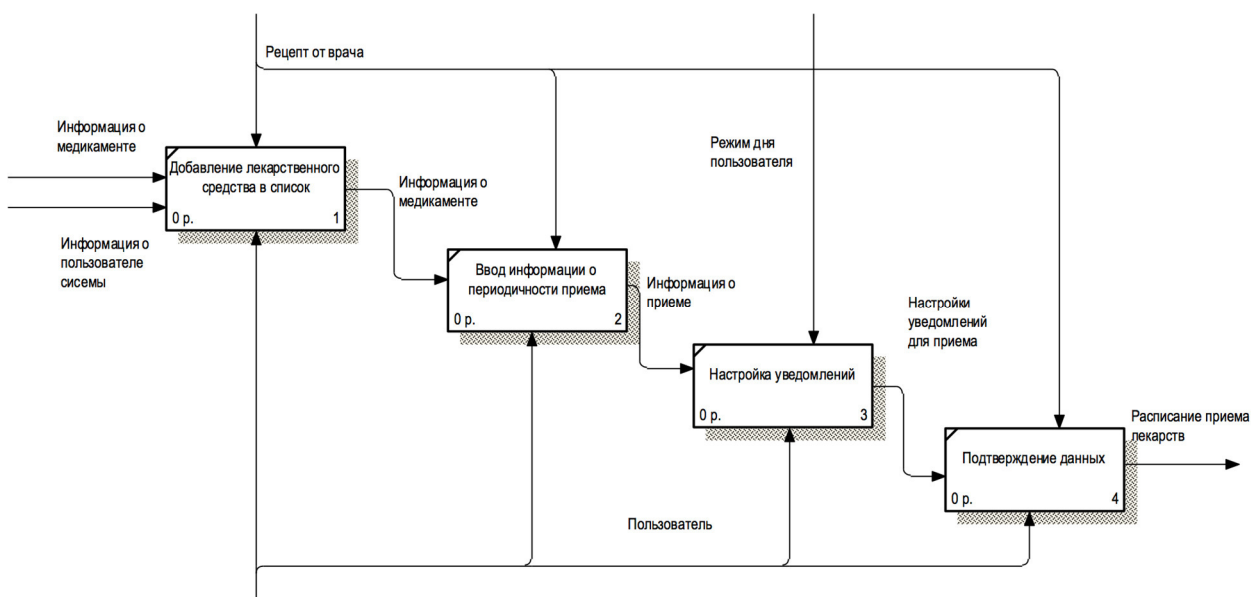


Рисунок 2.7 – Декомпозиция диаграммы «Формирование графика приема лекарств»

Формирование графика приема лекарственных средств происходит следующим образом: вначале пользователь добавляет необходимый ему медикамент в список с помощью поиска, с учетом дозировки препарата, далее необходимо ввести информацию о периодичности приема (доза одного приема, количество повторений в сутки, период, длительность приема). Следующий шаг – это настройка уведомлений, это процесс подразумевает под собой установку типа уведомлений, количество повторений, время повторения и т.п. После подтверждения данных формируется график приема лекарств.

2.2 Разработка инфологической модели

Цель инфологического этапа проектирования состоит в получении семантических (концептуальных) моделей, отражающих предметную область и информационные потребности пользователей [17]. Семантическое моделирование представляет собой моделирование структуры данных, опираясь на смысл этих данных. В качестве инструмента для построения семантических моделей данных на этапе инфологического проектирования является неформальная модель «Сущность-Связь» [18] (ER-модель – Entity-Relationship). Моделирование предметной области базируется на использовании графических диаграмм, включающих небольшое число разнородных компонентов.

Модели верхнего уровня разрабатываются на начальных стадиях проектирования информационных систем. Модели нижнего уровня разрабатываются на этапе создания программного кода и тестирования.

Основными понятиями ER-модели являются сущность, связь и атрибут.

В работе было определено 19 сущностей: Фармакологические свойства, Фармакологическое действие, Фармакологическая группа, Условия отпуска Действующее вещество, АТС[1], АТС[2], АТС[3], АТС[4], АТС[5], Препарат АТС, Препарат - Фармакологическое действие, Препарат - Фармакологическая группа, Препарат - Условия отпуска, Препарат - Действующее вещество, Инструкция, Пользователь, Препараты пользователя, Прием.

Рассмотрим подробно каждую из них.

В сущности Фармакологические свойства описаны свойства каждого препарата. В сущности Фармакологическое действие храниться список возможных оказываемых лекарственным средством действий на организм. Сущность Фармакологическая группа включает в себя классификацию фармакологических групп. Для хранения списка веществ используемых в медикаментах была создана сущность Действующее вещество.

Перечисленные выше сущности созданы для дальнейшей классификации лекарств по различным признакам.

В сущности Условия отпуска хранятся варианты продажи лекарств (без рецепта либо же по рецепту врача).

Для классификации лекарств по терапевтическому применению используется международная система классификации лекарственных средств АТХ (Анатомо-терапевтическо-химическая классификация) или АТС (Anatomical Therapeutic Chemical). АТХ-классификация имеет, понятную, иерархическую структуру, что облегчает поиск нужных лекарственных препаратов. Код АТХ состоит из семи символов, условно его можно разделить на 5 частей указывающих на определённый уровень иерархии в классификации. Для этого и были созданы следующие сущности: АТС[1], АТС[2], АТС[3], АТС[4], АТС[5], Препарат АТС.

Сущности: Препарат - Фармакологическое действие, Препарат - Фармакологическая группа, Препарат - Условия отпуска, Препарат - Действующее вещество были созданы для сопоставления лекарственных средств с определенными свойствами, указанными в каждой сущности.

Сущность Инструкция содержит в себе всю информацию о медикаменте включая состав, описание, показания к применению, противопоказания, способ применения и др.

Сущность Пользователь создана для хранения личной информации пользователя (имя, дату рождения, почту и пароль от аккаунта).

В таблице Препараты будут добавляться все медикаменты которые будет принимать пользователь, с датой начала и окончания приема лекарства и указанием периода повторений.

Прием – сущность, созданная для хранения необходимого количества приемов для каждого препарата. Если в день определенный медикамент необходимо принимать три раза, то будет создана три приема для данного препарата с разным временем.

Для каждой сущности были определены атрибуты, характеризующие

объект. Какие именно можно увидеть на рисунке 2.8 где изображена логическая модель данных.

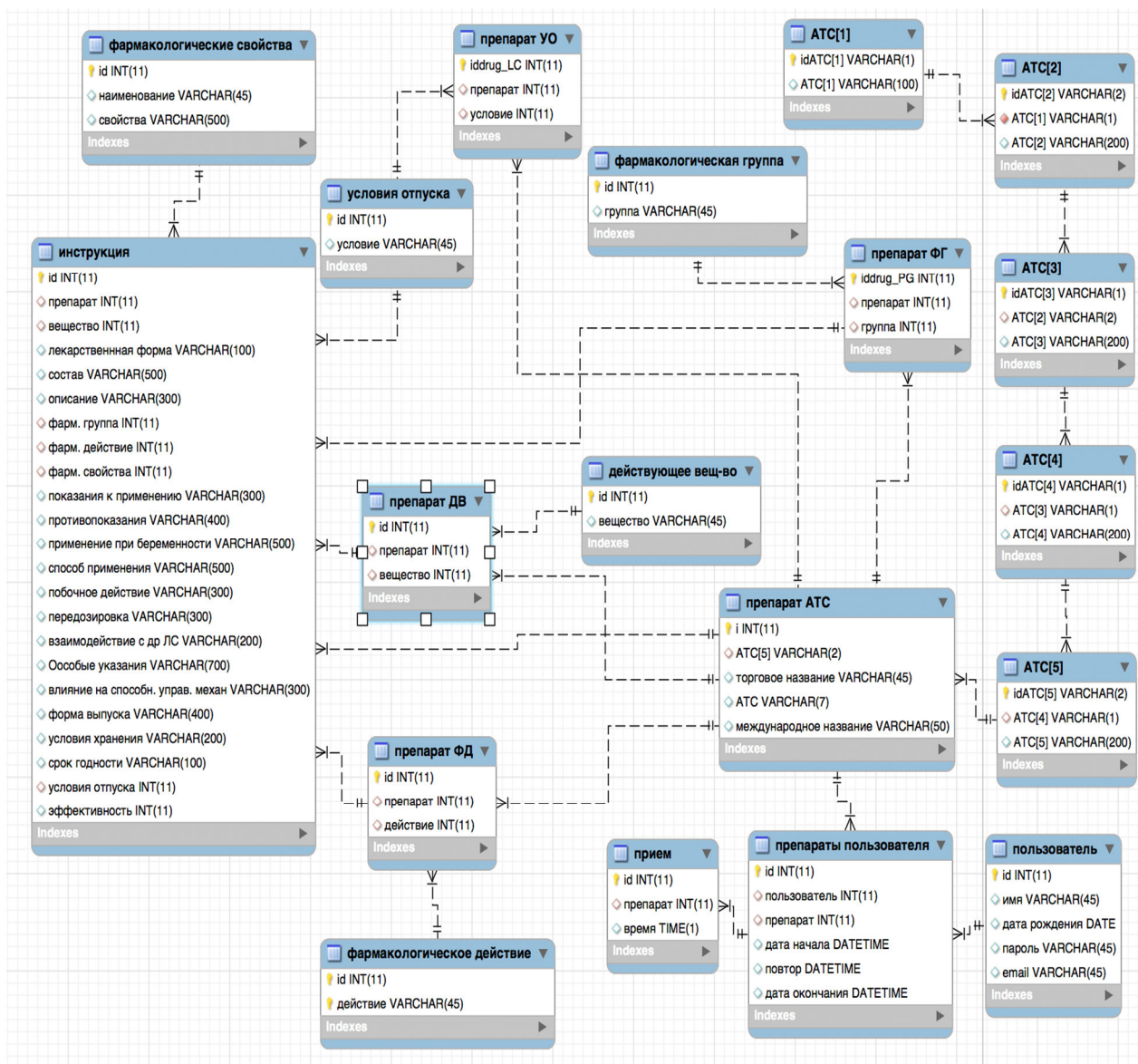


Рисунок 2.8 – Логическая модель данных

Следующим очень важным шагом является правильно спроектировать связи между таблицами с целью обеспечить связь с другим таблицами [19].

Связь (Relationship) – это поименованная графически изображаемая ассоциация, устанавливаемая между сущностями и представляющая собой абстракцию набора отношений, которые систематически возникают между различными видами предметов в реальном мире.

Всего было определено 22 внешних связи между таблицами, 17 из которых имеют тип один-ко-многим, а 5 – многие-ко-многим. Это связи определяющие отношения между таблицами содержащими информацию о фармакологическом действии, фармакологической группе, действующих веществах, условия отпуска медикаментов, препаратов пользователя и таблицей содержащей данные лекарственного средства.

2.3 Разработка даталогической модели

Следующим шагом проектирования является создание физической (даталогической) модели базы данных [19]. Исходными данными для даталогического проектирования является инфологическая модель предметной области, которая представлена на рисунке 2.8.

При этом основной задачей даталогического моделирования является описание свойств понятий предметной области, типы данных полей, их взаимосвязь и ограничения, накладываемые на данные. Даталогическая модель является начальным прототипом создаваемой базы данных.

Конечным результатом даталогического проектирования является описание логической структуры БД на языке описания данных либо схематическое графическое изображение структуры БД. Спроектировать структуру данных – значит определить все информационные единицы и связи между ними, задать их имена, определить их тип [20].

Физическая модель построена на основе логической модели данных и изображена на рисунке 2.9 – физическая модель учета лекарственных средств.

Эти сущности и атрибуты являются основой для создания таблиц в базе данных для дальнейшего хранения информации. Изменение данных в одной таблице повлечет за собой изменения в другой, таким образом обеспечивается целостность данных.

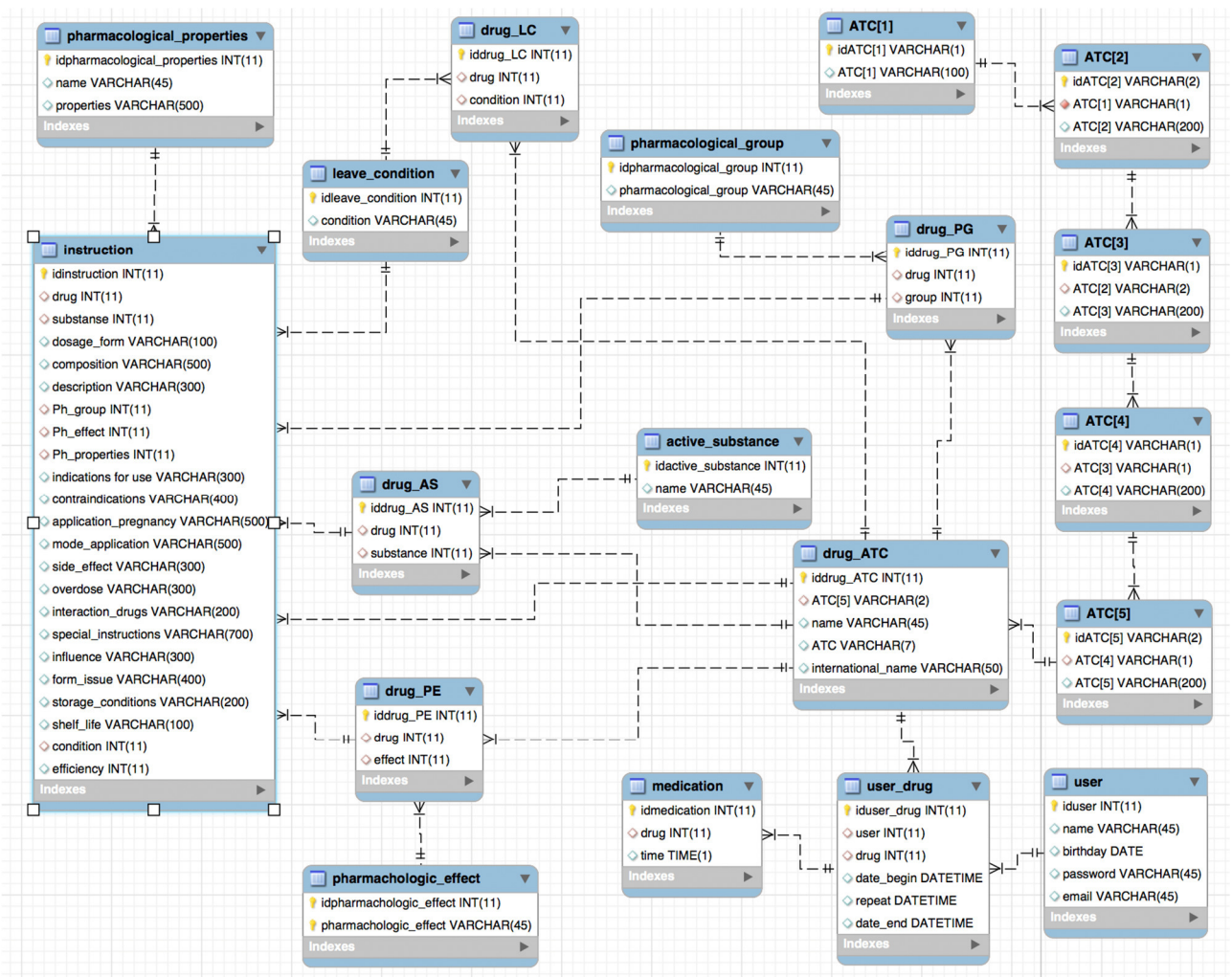


Рисунок 2.9 – Физическая модель данных

Выводы по второму разделу.

В результате проведенных работ была получена физическая модель базы данных, которая будет использоваться в разрабатываемом мобильном приложении. Были определены и описаны 19 сущностей: Фармакологические свойства, Фармакологическое действие, Фармакологическая группа, Условия отпуска Действующее вещество, АТС[1], АТС[2], АТС[3], АТС[4], АТС[5], Препарат АТС, Препарат - Фармакологическое действие, Препарат - Фармакологическая группа, Препарат - Условия отпуска, Препарат - Действующее вещество, Инструкция, Пользователь, Препараты пользователя, Прием. Атрибуты каждой таблицы с указанным типом данных можно увидеть на рисунках выше или в Приложении А.

3 Программная реализация

Для создания базы данных была выбрана система управления базами данных SQLite [21]. Это легко встраиваемая в приложения база данных. Так как эта система базируется на файлах, то она предоставляет довольно широкий набор инструментов для работы с ней, по сравнению с сетевыми СУБД. При работе с этой СУБД обращения происходят напрямую к файлам (в эти файлах хранятся данные), вместо портов и сокетов в сетевых СУБД. Именно поэтому SQLite очень быстрая, а также мощная благодаря технологиям обслуживающих библиотек.

Основным преимуществом данной СУБД является файловая структура. Вся база данных состоит из одного файла, поэтому её очень легко переносить на разные машины. Также эта СУБД является отличным вариантом при разработке и тестировании. Очень часто в процессе разработки приложений появляется необходимость масштабирования. SQLite предлагает всё что необходимо для этих целей, так как состоит всего из одного файла и библиотеки написанной на языке C.

Главный недостаток СУБД SQLite это проблемы с записью больших объемов данных – одно из ограничений SQLite это операции записи. Разрешен только один процесс записи в промежуток времени, что сильно ограничивает производительность.

В связи с тем, что большинство людей являются пользователями смартфонов под управлением системы android, подсистема подбора лекарственных средств была реализована именно для такого вида смартфонов. В связи с этим в качестве средства разработки программы используется язык программирования Java и среда разработки Android Studio. К обязательным инструментам относится Android SDK – набор средств программирования, который содержит инструменты, необходимые для создания, компиляции и сборки мобильного приложения. Android Studio [22] — интегрированная среда разработки производства Google, с помощью которой разработчикам

становятся доступны инструменты для создания приложений на платформе Android OS. Android Studio является универсальным приложением потому, что. IDE подходит как для крупных проектов с большим количеством задействованных специалистов, так и для одиночных разработчиков.

3.1 Разработка базы данных

Используя описанную ранее инфологическую модель (рисунок 2.9) была создана база данных [20].

Для создания базы данных используется высококачественный, визуальный инструмент – DB Browser для SQLite. Этот инструмент имеет открытый исходный код для создания, проектирования и редактирования файлов баз данных, совместимых с SQLite [21]. Он использует привычный табличный интерфейс, поэтому не нужно изучать сложные SQL-команды.

Средства управления DB Browser для SQLite позволяют: создавать файлы базы данных; создавать, определять, изменять и удалять таблицы; создавать, определять и удалять индексы; просматривать, редактировать, добавлять и удалять записи; искать записи; импортировать и экспортировать записи как текст; импортировать и экспортировать таблицы из/в файлы CSV; импортировать и экспортировать базы данных как SQL дампы; выполнять SQL-запросы и проверять результаты; изучать журнал всех SQL команд.

Для создания базы и таблиц использовался конструктор. Процесс создания базы данных представлен на рисунке 3.1.

При создании таблицы с помощью конструктора необходимо ввести основные данные – это название, при необходимости комментарии. Там же добавляются все необходимые поля таблицы и их свойства (название, тип данных, первичный ключ, при необходимости определить свойство генерации уникального идентификатора для новых строк, различные ограничения и др.).

В этом же конструкторе сразу можно добавить вторичные ключи, тем самым обеспечить связь с другими таблицами, указав ссылку на необходимую

таблицу и поле при редактировании данных. Пример представлен на рисунке 3.2. Характеристики полей каждой таблицы представлены в Приложении А.

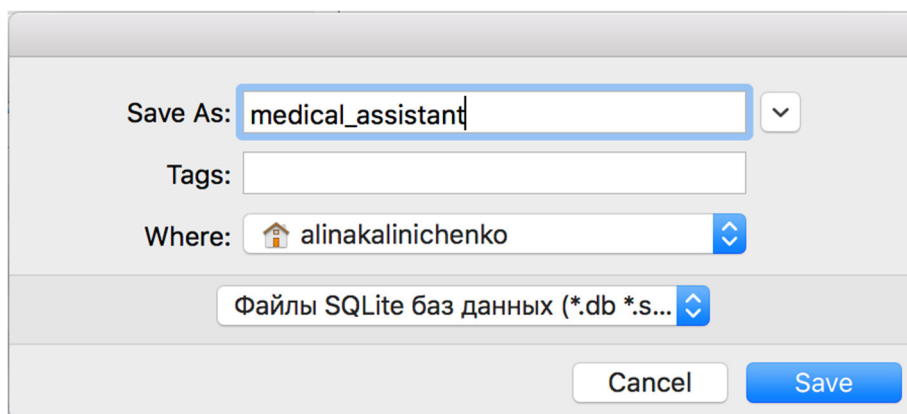


Рисунок 3.1 – Создание базы данных

На рисунке 3.3 показана структура всей базы данных. На вкладке отображаются все созданные таблицы, индексы, представления и триггеры с отражением SQL запроса создания. Также доступно редактирование всех объектов

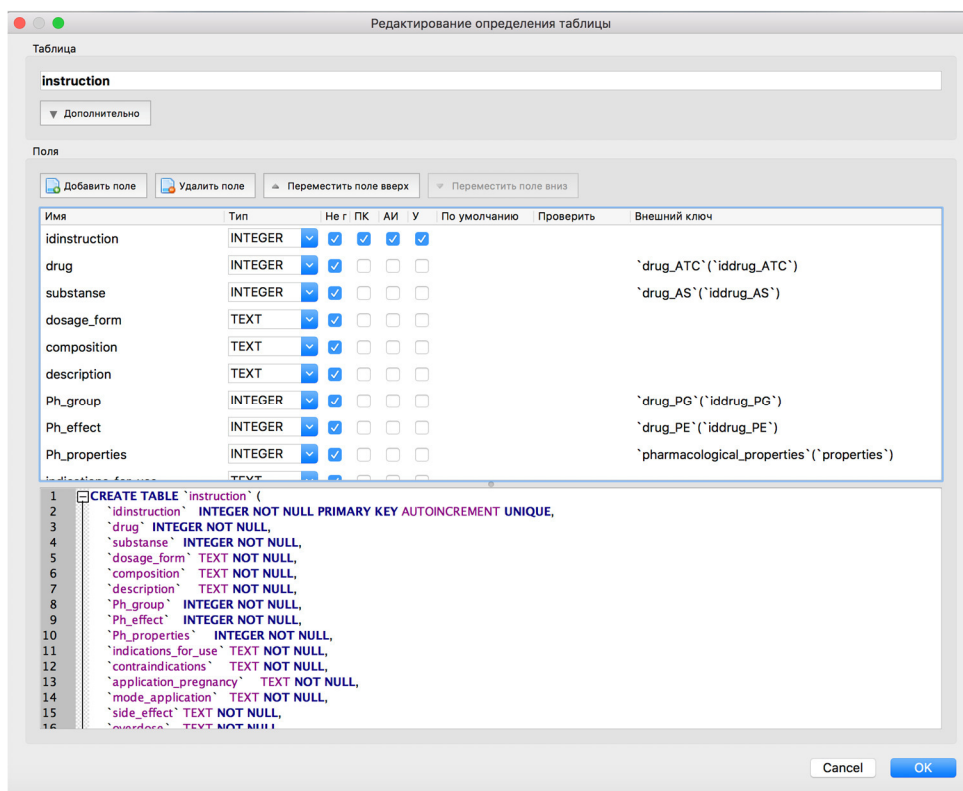


Рисунок 3.2 – Ввод основных данных при создании таблицы

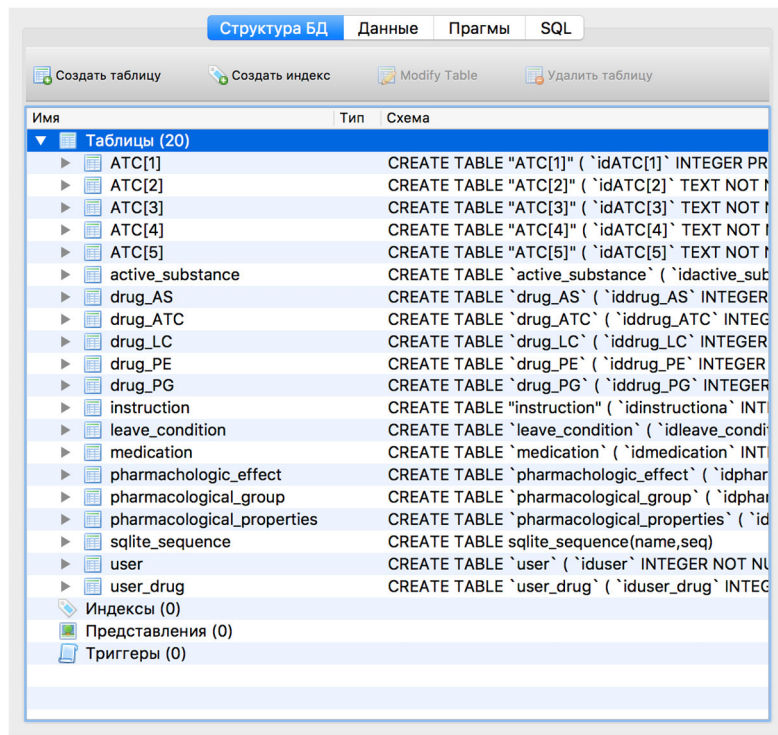


Рисунок 3.3 – Структура базы данных

На рисунке 3.4 представлена форма добавления индексов, создаваемых с целью повышения производительности поиска данных. Обычно индексы формируются для первичных и вторичных ключей.

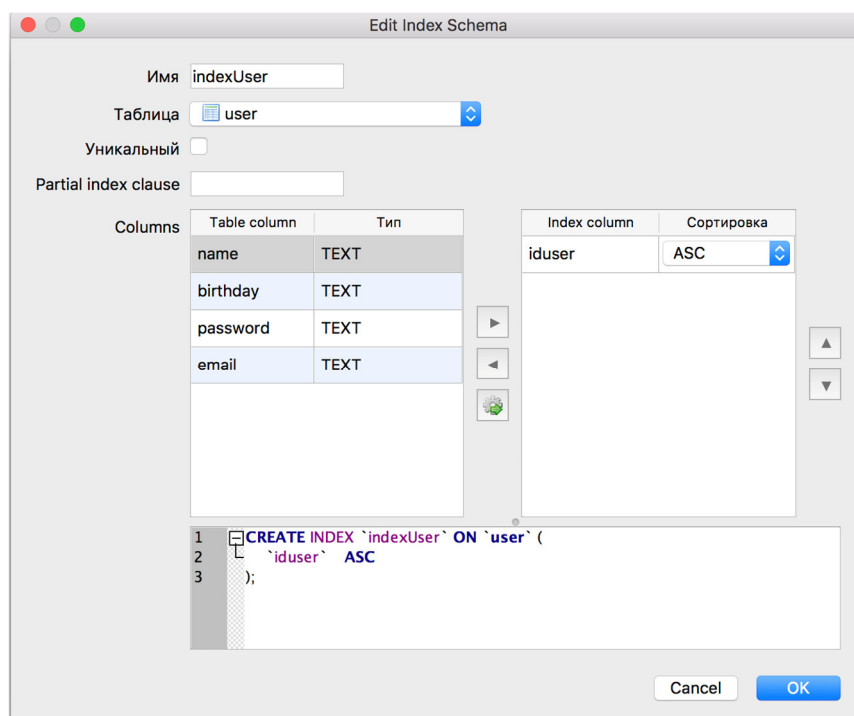


Рисунок 3.4 – Добавление индексов

Так же можно создать хранимые процедуры, которые не вызываются непосредственно, а исполняются при наступлении определенного события (вставка, удаление, обновление строки) – триггеры. Для предоставления интерфейса обратной совместимости, моделирующего таблицу, схема которой изменилась используют представления. Еще представления создаются в качестве механизма безопасности, позволяющего пользователям обращаться к данным через представления, но не предоставляя им разрешений на непосредственный доступ к базовым таблицам.

Добавлять данные в таблицу можно как через SQL запрос так и через конструктор. Где можно упорядочивать и сортировать данные прописывая различные фильтры. Пример представлен на рисунке 3.5

Структура БД **Данные** Прагмы SQL

Table:

	idATC[1] ▲	ATC[1]	name
	Фильтр	Фильтр	Фильтр
1	1	A	Пищеварительный тракт и обмен веществ
2	2	B	Кровь и система кроветворения
3	3	C	Сердечно-сосудистая система
4	4	D	Дерматологические препараты
5	5	G	Мочеполовая система и половые гормоны
6	6	H	Гормональные препараты системного действия, кроме п...
7	7	J	Противомикробные препараты системного действия
8	8	L	Противоопухолевые препараты и иммуномодуляторы
9	9	M	Костно-мышечная система
10	10	N	Нервная система
11	11	P	Противопаразитарные препараты, инсектициды и репелл...
12	12	R	Дыхательная система
13	13	S	Органы чувств
14	14	V	Прочие препараты

1 - 14 из 14

Рисунок 3.5 – Отображение данных таблицы

Редактирование значений ячейки доступно в отдельной форме, что является очень удобным для ввода в ячейку текста большой длины, так как данные просматриваются в полном объеме. Пример такого редактирования представлен на рисунке 3.6. Также на форме есть возможность импортировать и экспортировать данные отдельно конкретной ячейки.

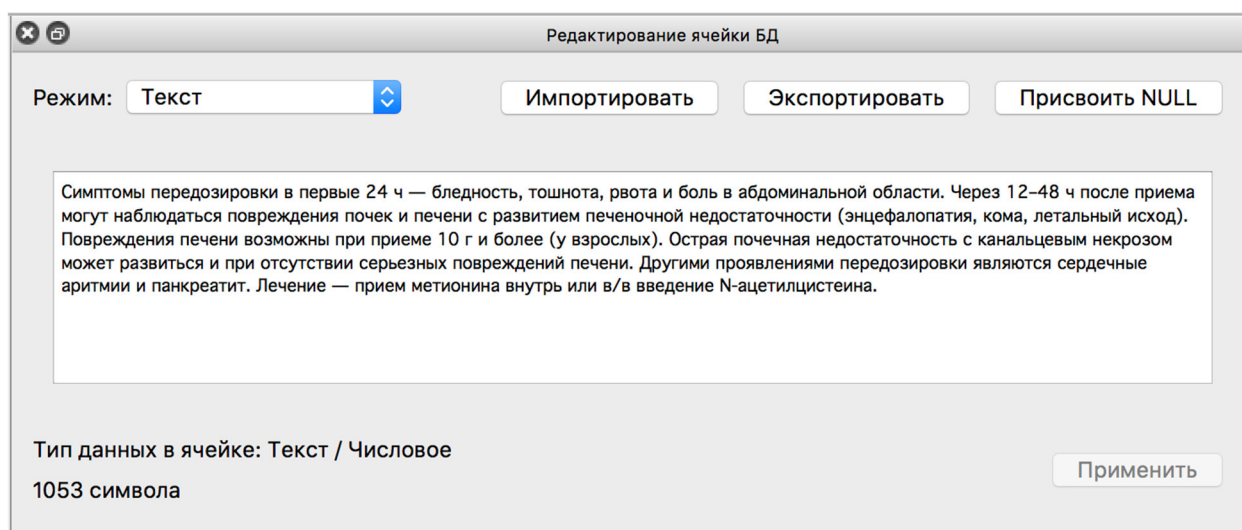


Рисунок 3.6 – Редактирование значения ячейки

Далее созданная и заполненная база данных была добавлена в проект приложения [23], сам файл был скопирован в папку проекта, а также были созданы специальные классы для работы с базой данных. Для подключения базы данных нужно указать основные переменные: `DB_NAME` – имя файла базы данных, `DB_PATH` – путь к базе данных и `DB_VERSION`. Последняя переменная это номер версии базы данных. В момент создания установки приложения программа должна скопировать созданную базу данных на устройство. Потом через какое-то время необходимо обновить приложение, так как база данных обновилась: структура базы поменялась, были добавлены новые данные. Необходимо заменить старую БД на новую. Для этого в данной переменной необходимо прописать новую версию БД. И при открытии приложения будет произведена проверки версии БД и файл БД обновится. Вначале версия БД равна 1.

3.3 Разработка и тестирование информационной системы

При запуске приложения на экране появляется изображение символа медицины и сообщение призывающее людей не заниматься самолечением. Так как данное приложение не пропагандирует самолечение, а наоборот создано с целью привить более серьезное отношение к своему здоровью и обезопасить людей от процесса самолечения. Скриншот экрана представлен на рисунке 3.7.

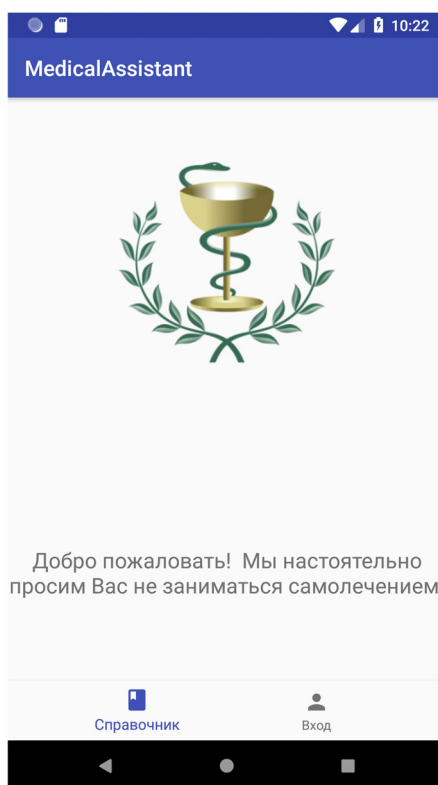


Рисунок 3.7 – Экран смартфона при запуске приложения

Изначально на нижней панели навигации доступны лишь два действия – это просмотр справочника и возможность входа в систему. Фармакологический справочник доступен не только зарегистрированным в системе пользователям. Это было сделано для увеличения целевой аудитории, что бы привлечь людей, которые по каким-либо причинам не ведут график приема лекарственных средств.

В справочнике данные классифицированы по анатомо-терапевтическо-химическому признаку. АТХ-классификация имеет, понятную, иерархическую структуру, что облегчает поиск нужных лекарственных препаратов. Пример показан на рисунке 3.8. Все данные для справочника включая и саму АТХ-классификацию черпаются из реестра лекарственных средств.

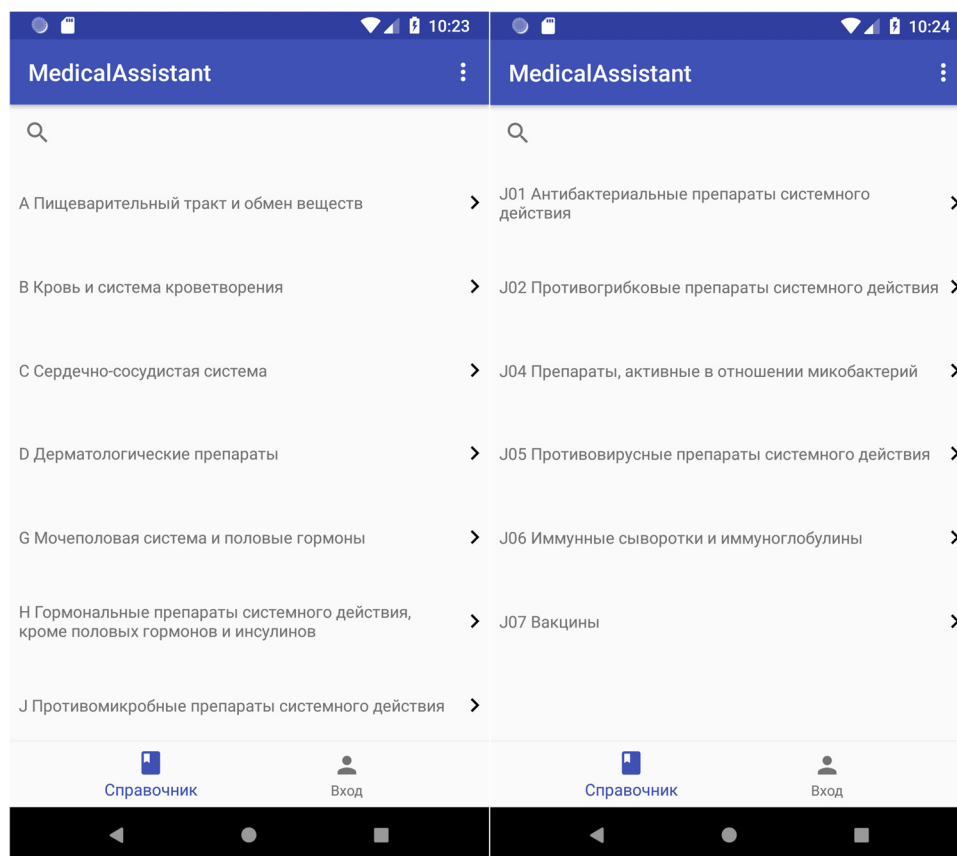


Рисунок 3.8 – Классификация медикаментов в справочнике

Государственный реестр лекарственных средств (Госреестр лекарственных средств) – перечень отечественных и зарубежных лекарств, медико-профилактических и диагностических средств, зарегистрированных Минздравом России [3].

Госреестр лекарственных средств ведется и издается в соответствии с Законом о лекарственных средствах и является официальным изданием Министерства здравоохранения Российской Федерации. При составлении

Реестра используются следующие официальные документы Минздрава России: приказы о разрешении медицинского применения, регистрационные удостоверения, фармакопейные и временные фармакопейные статьи, нормативные документы на зарубежные лекарственные средства, удостоверение регистрации уникального номера.

Информация о торговых и международных непатентованных названиях, лекарственных формах, упаковках, дозировках, фармакологических группах, приведенная в Реестре, должна служить основой для формирования различных перечней и списков

На рисунке 3.9 представлен пример отображения списка медикаментов в справочнике

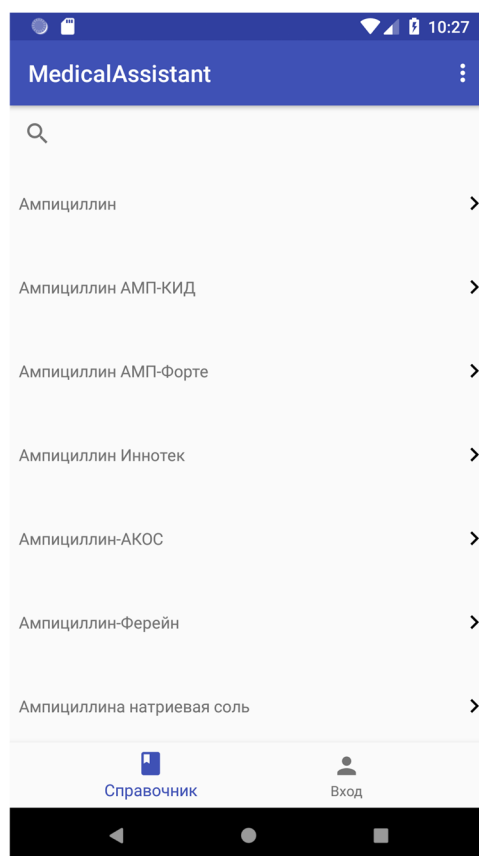


Рисунок 3.9 – Пример отображения списка медикаментов в справочнике

Для удобства, при просмотре инструкции лекарственного средства вся информации была разбита на группы, которые можно просмотреть отдельно.

Также присутствует возможность просмотреть аналоги данного медикамента. Пример представлен на рисунке 3.10.

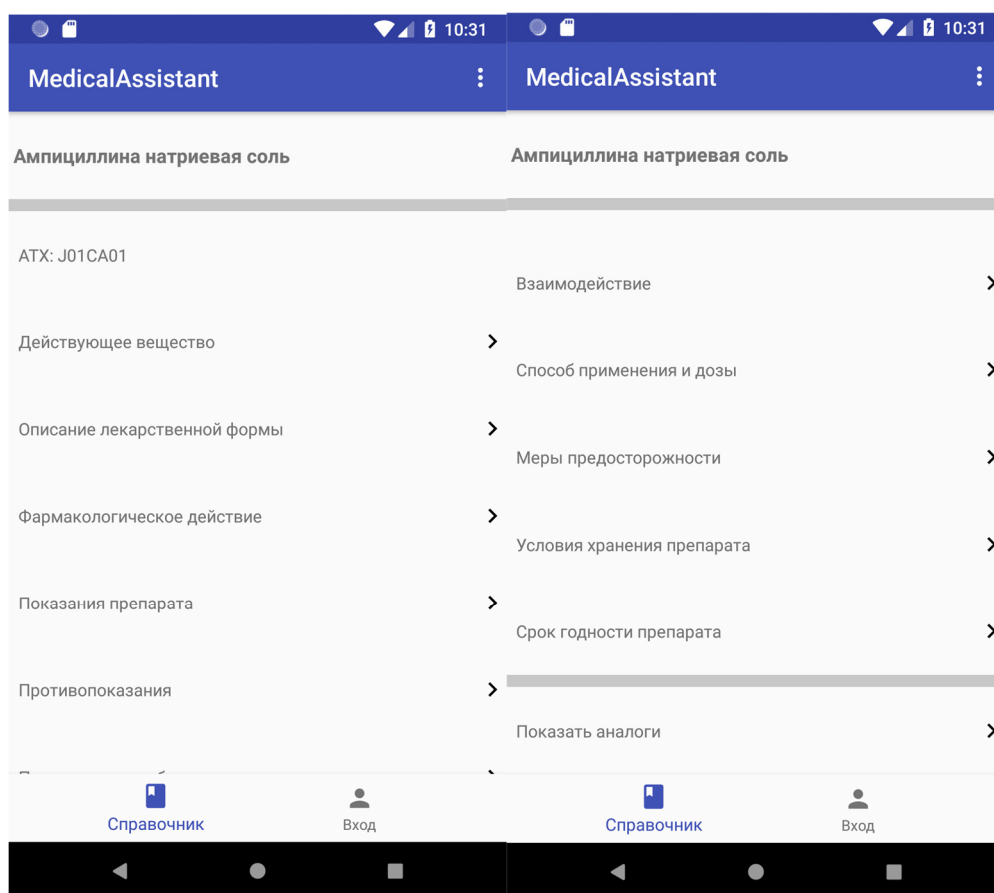


Рисунок 3.10 – Группирование основной информации лекарственного средства

При переходе в какую либо категорию откроется новая страница с подробным описанием, пример показан на рисунках 3.11 и 3.12.

Отдельно выделена возможность просмотра аналоговых препаратов, эта функция работает по принципу поиска лекарственных средств с таким же действующим веществом или сочетанием веществ в базе данных. При переходе в эту категорию появится список подходящих лекарственных средств, полную информацию о которых также можно просмотреть перейдя на страницу отдельного препарата.

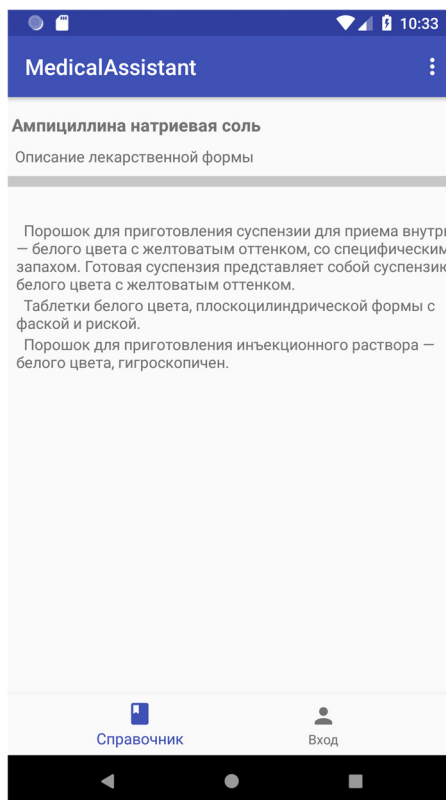


Рисунок 3.11 – Просмотр подробной информации о лекарственной форме медикамента

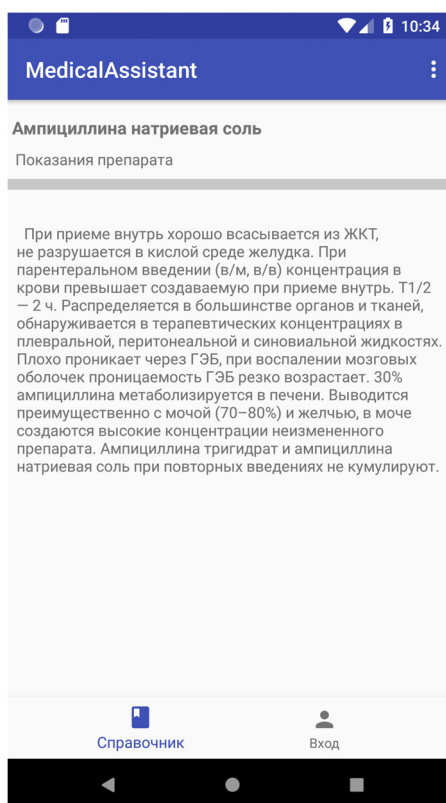


Рисунок 3.12 – Просмотр подробной информации касательно показаний препарата

Также в системе доступен поиск по ключевым параметрам, таким как: торговое название, действующее вещество, терапевтическое применение, код АТХ, фармакологическое действие, или же фармакологическая группа. Можно сортировать и упорядочивать списки лекарств без поиска конкретного необходимого медикамента. Есть возможность просмотреть классификацию фармакологических групп и список лекарств по фармакологическому действию.

На рисунке 3.13 представлен пример поиска медикаментов по действующему веществу

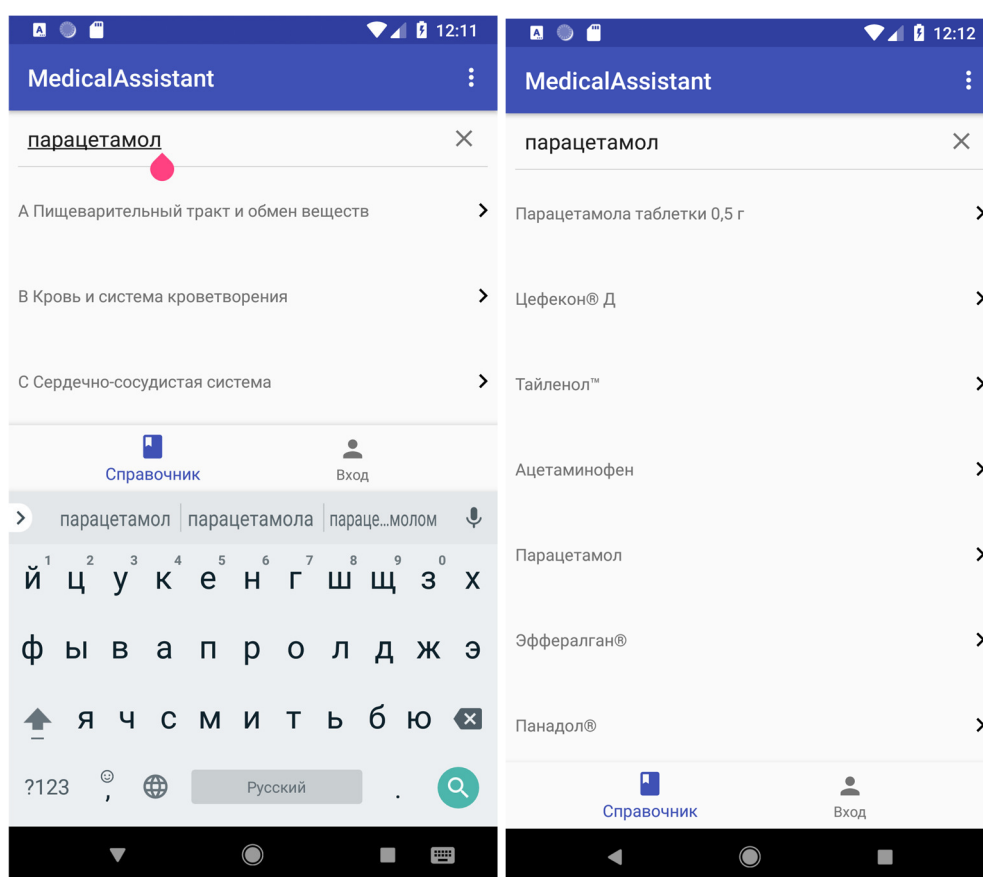


Рисунок 3.13 – Поиск медикаментов по действующему веществу

При поиске медикаментов есть возможность упорядочить данные по двум признакам: название и индекс Вышковского [20] (рисунок 3.14). По умолчанию сортировка происходит по значению индекса Вышковского. Данный индекс используется в реестре лекарственных средств и представляет

из себя опережающий индикатор рынка, позволяющий в режиме реального времени не только оценивать эффективность рекламных акций и прогнозировать изменения объемов продаж лекарственных средств, но и влиять на эти показатели.

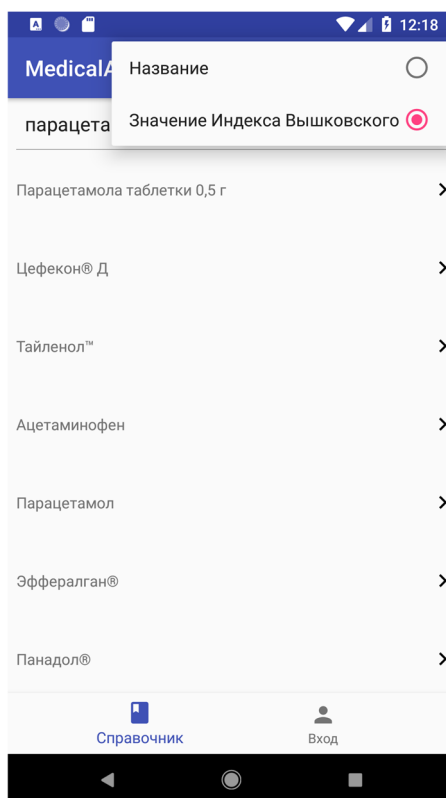


Рисунок 3.14 – Упорядочивание данных при поиске

При переходе на вкладку «Вход» пользователю доступны два действия – это вход в систему, если пользователь уже зарегистрирован, и создание нового аккаунта. Скриншот экрана представлен на рисунке 3.15.

На рисунке 3.16 изображена форма входа в систему, где пользователю необходимо заполнить указанные поля. Данные, введенные пользователем, сравниваются с данными в базе. Если существует пользователь с введенными логином и паролем, то приложение переходит в режим работы пользователя. В ином случае появляется уведомление о не верном вводе логина или пароля.

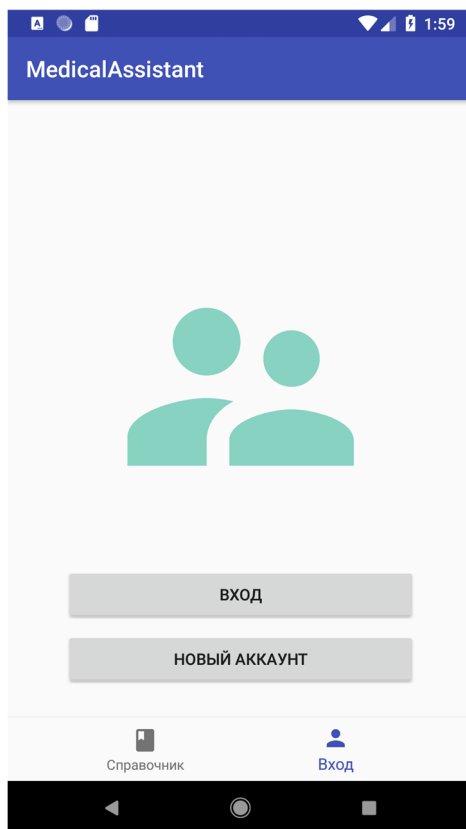


Рисунок 3.15 – Скриншот экрана при переходе на вкладку входа

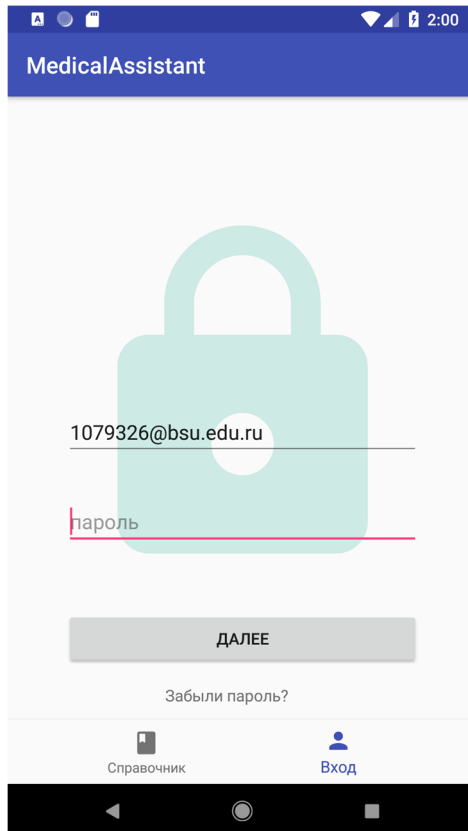


Рисунок 3.16 – Форма входа в систему

В случае если пользователь забыл пароль его можно восстановить, указав привязанную почту и отправив письмо с указаниями на почту.

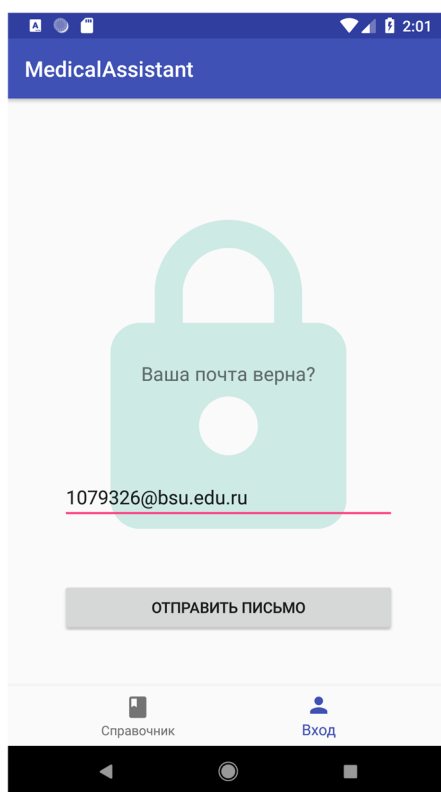


Рисунок 3.17– Скриншот экрана при восстановлении пароля

Если пользователь решил зарегистрироваться в системе, ему нужно ввести всю необходимую личную информацию: имя, почту, пароль, дату рождения, пол. При дальнейшем редактировании пользователь по желанию может указать больше информации о себе включая рост, вес, существующие аллергии и др., для того, что бы во время использования приложения можно было сопоставить личную информацию и характеристику медикамента для создания уведомлений под каждого пользователя.

Например если девушка укажет что ожидает ребенка, медикаменты которые противопоказаны при беременности, будут помечены отличительным знаком или же если пользователь укажет, что водит машину, при приеме медикамента запрещающего вождение, появится уведомительное сообщение.

Обязательным условием является подтверждение согласия на обработку персональных данных, согласно части 1 статьи 6 закона «О персональных данных» [16]. Одним из результатов работы системы является сообщение о необходимости согласия в случае не выполнения такового. Форма регистрации представлена на рисунке 3.18.

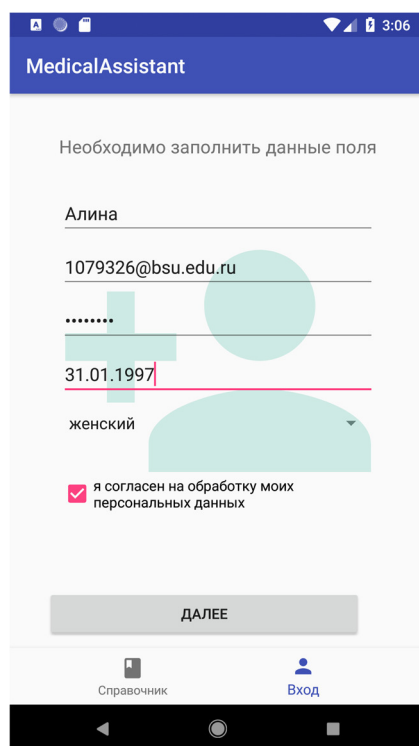


Рисунок 3.18 – Ввод данных при регистрации в системе

После входа в систему в панели навигации доступны еще две вкладки – это вкладка «График», где осуществляется настройка графика приема лекарственных средств и «История», где храниться список всех принимаешь ранее медикаментов.

На рисунке 3.19 показана вкладка «График». На экране есть две кнопки, обеспечивающие ведение графика. Кнопка «+» служит для добавления в список нового препарата. При нажатии этой кнопки открывается список лекарств по АТХ-классификации, где можно найти необходимый препарат и добавить его в список принимаемых.

После нажатия кнопки «Изменить» становится доступна функция удаления лекарств из списка принимаемых, однако информация останется в истории принимаемых лекарственных средств.

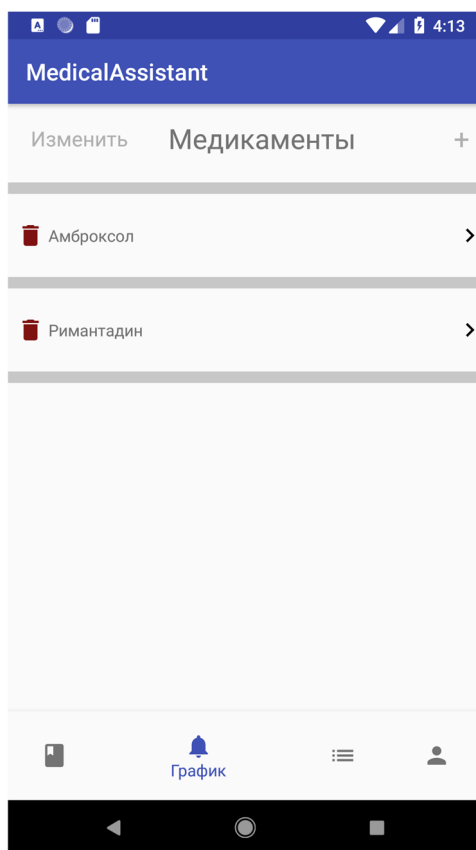


Рисунок 3.19 – Список лекарств добавленных в график приема

После выбора одного медикамента из имеющихся в списке, становится доступным редактирование информации о периодичности приема и настройка уведомлений. На экране появляется список приемов медикамента, пример показан на рисунке 3.20.

Пользователь может добавить нужное ему количество приемов для каждого лекарственного средства. Например, если больному необходимо принимать лекарство три раза в день, ежедневно в течении 7 дней, нужно добавить три приема, где важно указать время приема для каждого раза.

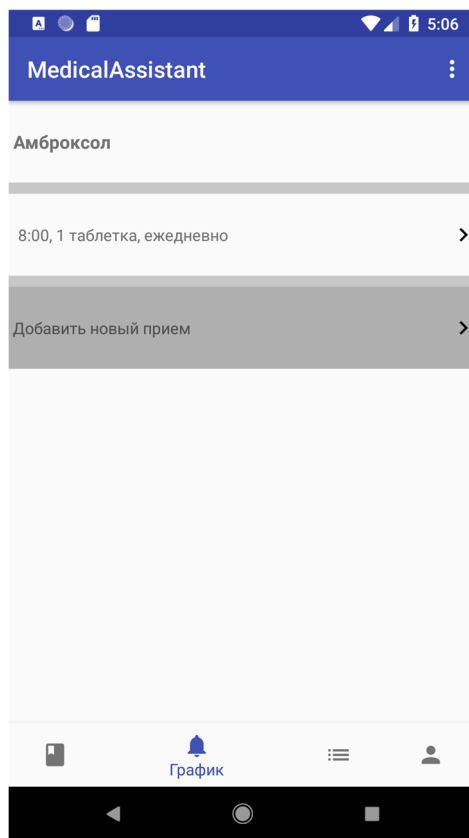


Рисунок 3.20 – Редактирование графика приема конкретного медикамента

На рисунке 3.21 показан пример добавления нового приема. Необходимо указать время приема, когда нужно показать уведомление, какая необходимая доза, с какой периодичность необходимо принимать препарат, с какой даты начинать и сколько приемов включает в себя курс.

Если препарат необходимо принимать с определенным временным промежутком, например 8 часов, достаточно создать один прием, где в поле периодичность указать требуемый промежуток.

После нажатия кнопки «Готово» данные будут добавлены в список приемов и при наступлении времени, указываемого при настройке появится уведомление о необходимости принять препарат. В уведомлении отображаются все важные рекомендации по применению: доза, информация о том, когда лучше принять лекарство: после приема пищи, во время и т.п., информация о том, какой по счету это прием и сколько еще осталось. Если после принятия препарата не рекомендуется вождение, будет соответствующее сообщение.

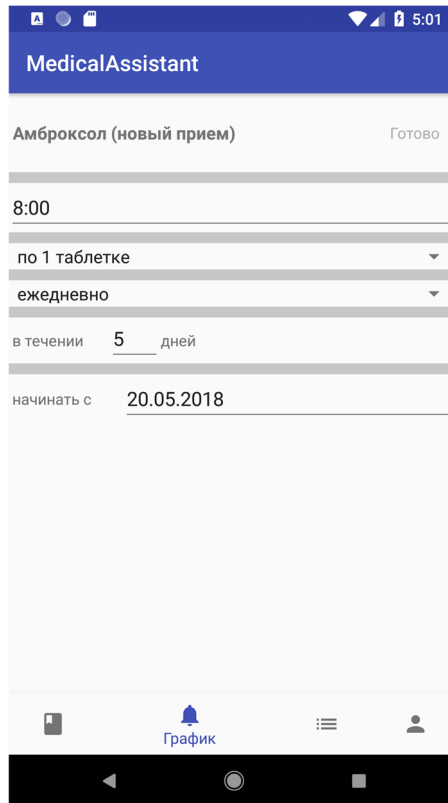


Рисунок 3.21 – Добавление нового приема

Выводы по третьему разделу.

Таким образом, была осуществлена программная реализация подсистемы подбора лекарственных средств, а также был проверен функционал, который система должна выполнять. Функции соответствуют описанным.

Проанализировав выше представленные результаты, можно убедиться, что данная программа работает правильно и готова к дальнейшему использованию. В подобных системах главное – обновление и улучшение функционала, оптимизация и пополнение базы данных новыми сведениями. Данную программу можно развивать и совершенствовать, что и планируется делать в дальнейшем.

ЗАКЛЮЧЕНИЕ

В ходе выполнения выпускной квалификационной работы было разработано мобильное приложение, позволяющее получить полную информацию о медикаментах в различных областях применения и назначить график принимаемых лекарств, что поможет соблюдать график приема лекарств для улучшения эффективности лечения. Уровень изменения эффективности лечения будет высчитываться спустя 3 месяца пользования приложением, посредством опроса клиентов.

Были выполнены все поставленные задачи: была описана концепция проекта разработки мобильного приложения по подбору лекарственных средств, изучены существующие варианты решения данного вопроса, проведен обзор целевой аудитории. Также была описана функциональная схема предметной области, спроектированные логическая и физическая схемы модели базы данных.

Из результата проведенных работ можно сделать вывод, что разработка данного мобильного приложения является актуальной, так как спрос на продукт достаточно велик. Главное отличие данного проекта от уже существующих вариантов – это совмещение в себе фармакологического справочника и календаря графика приема медикаментов, что является очень удобным для пользователей. Еще очень важным преимуществом данной программы является настройка уведомлений и системы в целом под каждого пользователя в соответствии с введенными личными данными.

Информационная система позволяет структурировать и автоматизировать процесс контролирующей график приема лекарств, это способствует повышению качества управленческой информации, что приводит к повышению качества принятия решений по подбору лекарств.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. World Health Organization. World health statistics 2016: monitoring health for the SDGs sustainable development goals. – World Health Organization, 2016.
2. Новожилова Н. В., Алякина Л. А. Вопросы проектирования информационных систем //Вестник Чувашского университета. – 2011. – №. 1.
3. Реестр лекарственных средств России [Электронный ресурс] / Электрон. справочник. – 2000. – Режим доступа: <http://docplayer.ru/38569900-Elektronnaya-registratura.html>. – Дата доступа: 26.05.2018.
4. Репин В., Елиферов В. Процессный подход к управлению. Моделирование бизнес-процессов. – Litres, 2017.
5. Фатхутдинов Р. А. Стратегический маркетинг: Учебник для вузов. 5-е издание. – «Издательский дом «Питер»», 2008.
6. Литвинова О. А., Речинский А. В. IT: обзор рынка и примеры классификации специальностей //Прикладная информатика. – 2007. – №. 4.
7. AbleBits mobile, Мои таблетки [Электронный ресурс] / Электрон. текст. данные. – 2015. – Режим доступа: <https://mobile.ablebits.com/ru/pills-reminder.php>. – Дата доступа: 27.04.2018.
8. MyTherapy: Your personal pill reminder and medication tracker app [Электронный ресурс] / Электрон. текст. данные. – 2018. – Режим доступа: <https://www.mytherapyapp.com>. – Дата доступа: 27.04.2018.
9. Медицинский помощник [Электронный ресурс] / Электрон. текст. данные. – 2016. – Режим доступа: <http://medicalhelper.com.ua>. – Дата доступа: 30.04.2018.
10. Surkova E. V., Melnikova O. G. Adherence to medicamentous therapy of type 2 diabetes mellitus: results of patient questionnaires //Diabetes mellitus. – 2009. – Т. 12. – №. 1. – С. 48-50.
11. Котлер Ф., Келлер К. Л. Маркетинг менеджмент. – Издательский дом " Питер", 2007.

12. Кметь Е. Б. Методика исследования медиа предпочтений целевой аудитории: полное погружение // Экономика и предпринимательство. – 2014. – №. 12-3. – С. 919.
13. Волкова В. Н., Ден исов А. А. Теория систем и системный анализ // М.: Юрайт. – 2010.
14. Kim S. H., Jang K. J. Designing performance analysis and IDEF0 for enterprise modelling in BPR // International Journal of production economics. – 2002. – Т. 76. – №. 2. – С. 121-133.
15. Федоров Н. В. Проектирование информационных систем на основе современных CASE-технологий: учебное пособие. – МГИУ, 2008.
16. Копылов В. А. Информационное право. – М. : Юристъ, 1997. – С. 22.
17. Пирогов В. Ю. Информационные системы и базы данных: организация и проектирование. – БХВ-Петербург, 2009.
18. Костенко А. Б., Погребняк Б. И. Принцип декомпозиции в проектировании инфологических моделей информационных систем // Коммунальное хозяйство городов. – 2006. – №. 69. – С. 105-109.
19. Головчинер М. Н. и др. Базы данных: Основные понятия, модели данных, процесс проектирования: учебное пособие. – 2009.
20. Михелёв В.М. Базы данных и СУБД / В.М. Михелёв. – Белгород.: БелГУ, – 2007. – 200 с.
21. Owens M., Allen G. SQLite. – Apress LP, 2010.
22. Smyth N. Android Studio 3.0 Development Essentials-Android 8 Edition. – Payload Media, Inc., 2017.
23. Collins C., Galpin M., Kärpler M. Android in practice. – Manning Publications Co., 2011.
24. Вышковский Г. Л. Особенности российского рынка лекарств: взгляд со стороны системы РЛС // Ремедиум. Журнал о российском рынке лекарств и медицинской технике. – 2006. – №. 4.

ПРИЛОЖЕНИЕ А

Характеристики полей каждой таблицы базы данных

Таблица 1– атрибуты таблицы АТС[1]

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
idАТС[1]	integer	+	+	+	+	
АТС[1]	text	+				
Name	text	+				

Таблица 2– атрибуты таблицы АТС[2]

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
idАТС[2]	integer	+	+	+	+	
АТС[2]	text	+				
АТС[1]	integer	+			+	АТС[1].idАТС[1]
Name	text	+				

Таблица 3– атрибуты таблицы АТС[3]

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
idАТС[3]	integer	+	+	+	+	
АТС[3]	text	+				
АТС[2]	integer	+			+	АТС[2].idАТС[2]
Name	text	+				

Таблица 4– атрибуты таблицы АТС[4]

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
idАТС[4]	integer	+	+	+	+	
АТС[4]	text	+				
АТС[3]	integer	+			+	АТС[3].idАТС[3]
Name	text	+				

Таблица 5– атрибуты таблицы АТС[5]

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
idАТС[5]	integer	+	+	+	+	
АТС[5]	text	+				
АТС[4]	integer	+			+	АТС[4].idАТС[4]
Name	text	+				

Таблица 6– атрибуты таблицы active substance

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
idactive_substance	integer	+	+	+	+	
name	text	+				

Таблица 7– атрибуты таблицы drug_ATC

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
iddrug_ATC	integer	+	+	+	+	
ATC[5]	text	+				ATC[5].idATC[5]
name	text	+				
ATC	text	+				
international_name	text	+				

Таблица 8– атрибуты таблицы leave_condition

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
idleave_condition	integer	+	+	+	+	
condition	text	+				

Таблица 9– атрибуты таблицы pharmacologic_effect

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
idpharmacologic_effect	integer	+	+	+	+	
pharmacologic_effect	text	+				

Таблица 10– атрибуты таблицы pharmacological_group

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
idpharmacological_group	integer	+	+	+	+	
pharmacological_group	text	+				

Таблица 11– атрибуты таблицы pharmacological_properties

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
idpharmacological_properties	integer	+	+	+	+	
name	text	+				
properties	text	+				

Таблица 12– атрибуты таблицы user

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
iduser	integer	+	+	+	+	
name	text	+				
birthday	text	+				
password	text	+				
email	text	+				

Таблица 13– атрибуты таблицы user_drug

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
Iduser_drug	integer	+	+	+	+	
user	integer	+			+	user.iduser
drug	integer	+			+	drug.iddrug
Date_begin	text	+				
repeat	text	+				
Date_end	text	+				

Таблица 14– атрибуты таблицы medication

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
idmedication	integer	+	+	+	+	
drug	integer	+			+	user_drug.iduser_drug
time	text	+				

Таблица 15– атрибуты таблицы drug_AS

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
iddrug_AS	integer	+	+	+	+	
drug	integer	+			+	drug_ATC.iddrug_ATC
substance	integer	+			+	active_substance.idactive_substance

Таблица 16– атрибуты таблицы drug_LC

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
iddrug_LC	integer	+	+	+	+	
drug	integer	+			+	drug_ATC.iddrug_ATC
condition	integer	+			+	leave_condition.idleave_condition

Таблица 17– атрибуты таблицы drug_PE

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
iddrug_PE	integer	+	+	+	+	
drug	integer	+			+	drug_ATC.iddrug_ATC
effect	integer	+			+	pharmacologic_effect.idpharmacologic_effect

Таблица 18– атрибуты таблицы drug_PG

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
iddrug_PG	integer	+	+	+	+	
drug	integer	+			+	drug_ATC.iddrug_ATC
group	integer	+			+	pharmacologic_group.idpharmacologic_group

Таблица 19– атрибуты таблицы instruction

Имя	Тип	Не пустое	ПК	АИ	У	Вторичный ключ
idnstruction	integer	+	+	+	+	
drug	integer	+			+	drug_ATC.iddrug_ATC
substanse	integer	+			+	active_substance.idactive_substance
dosage_form	text	+				
composition	text	+				
description	text	+				
Ph_group	integer	+			+	drug_PG.iddrug_PG
Ph_effect	integer	+			+	drug_PE.iddrug_PE
Ph_properties`	integer	+			+	pharmacological_properties.properties
indications for use	text	+				
contraindications	text	+				
application_pregnancy	text	+				
mode_application	text	+				
side_effect	text	+				
overdose	text	+				
interaction_drugs	text	+				
special_instructions	text	+				
influence	text	+				
form_issue	text	+				
storage_conditions	text	+				
shelf_life	text	+				
condition	integer	+			+	leave_condition.idleave_condition
efficiency	integer	+				

ПРИЛОЖЕНИЕ Б

Листинг основного класса

```
package com.example.alinakalinichenko.medicalassistant;
import android.os.Bundle;
import android.support.annotation.NonNull;
import android.support.design.widget.BottomNavigationView;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.RecyclerView;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;
import java.util.List;
public class MainActivity extends AppCompatActivity {
    private TextView mTextMessage;
    private BottomNavigationView.OnNavigationItemSelectedListener
mOnNavigationItemSelectedListener
        = new BottomNavigationView.OnNavigationItemSelectedListener()
{
    @Override
    public boolean onNavigationItemSelectedListener(@NonNull MenuItem item) {
        switch (item.getItemId()) {
            case R.id.navigation_home:
                return true;
            case R.id.navigation_dashboard:
```

```

        return true;
    case R.id.navigation_history:
        return true;
    case R.id.navigation_schedule:
        return true;
    }
    return false;
}
};

```

@Override

```

public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu_main, menu);
    return true;
}

```

@Override

```

public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        case R.id.action_settings:
        case R.id.action_index_v:
            if (item.isChecked()) item.setChecked(false);
            else item.setChecked(true);
            return true;
        default:
            return super.onOptionsItemSelected(item);
    }
}

```

```

String[] data = {"пол", "женский", "мужской"};

```

```

String[] data_sum_tablet = {"по 1/4 таблетке", "по 1/2 таблетке", "по 1
таблетке", "по 5 мг", "по 10 мг", "другое"};

```

```

String[] data_period = {"ежедневно", "через день", "через неделю",
"другое"};

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    mTextMessage = (TextView) findViewById(R.id.message);
    BottomNavigationView navigation = (BottomNavigationView)
findViewById(R.id.navigation);
    navigation.setOnNavigationItemSelectedListener(mOnNavigationItemSelectedListener);

    // адаптер
    ArrayAdapter<String> adapter_sum_tablet = new
ArrayAdapter<String>(this, android.R.layout.simple_spinner_item,
data_sum_tablet);
    adapter_sum_tablet.setDropDownViewResource(android.R.layout.simple_s
pinner_dropdown_item);
    Spinner spinner_sum_tablet = (Spinner)
findViewById(R.id.sum_tablet);
    spinner_sum_tablet.setAdapter(adapter_sum_tablet);
    // выделяем элемент
    spinner_sum_tablet.setSelection(2);
    // spinner.getChildAt(1).setClickable(false);
    // устанавливаем обработчик нажатия
    spinner_sum_tablet.setOnItemSelectedListener(new
AdapterView.OnItemSelectedListener() {
        @Override
        public void onItemSelected(AdapterView<?> parent, View view,
int position, long id) {
            // показываем позиция нажатого элемента

```

```

        Toast.makeText(getBaseContext(), "Position = " + position,
Toast.LENGTH_SHORT).show();
    }
    @Override
    public void onNothingSelected(AdapterView<?> arg0) {
    }
});
    ArrayAdapter<String> adapter_period = new
ArrayAdapter<String>(this, android.R.layout.simple_spinner_item, data_period);

adapter_period.setDropDownViewResource(android.R.layout.simple_spinner_dro
pdown_item);

    Spinner spinner_period = (Spinner) findViewById(R.id.period);
    spinner_period.setAdapter(adapter_period);
    // заголовок
    // spinner.setPrompt("");
    // выделяем элемент
    spinner_period.setSelection(2);
    // устанавливаем обработчик нажатия
    spinner_period.setOnItemClickListener(new
AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> parent, View view,
            int position, long id) {
            // показываем позиция нажатого элемента
            Toast.makeText(getBaseContext(), "Position = " + position,
Toast.LENGTH_SHORT).show();
        }
    });
    @Override
    public void onNothingSelected(AdapterView<?> arg0) {

```

```

        }
    });
    //пол
    ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
android.R.layout.simple_spinner_item, data);
    adapter.setDropDownViewResource(android.R.layout.simple_spinner_drop
down_item);
    Spinner spinner = (Spinner) findViewById(R.id.spinner);
    spinner.setAdapter(adapter);
    // заголовок
    // spinner.setPrompt("Пол");
    // выделяем элемент
    spinner.setSelection(0);
    // устанавливаем обработчик нажатия
    spinner.setOnItemClickListener(new
AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> parent, View view,
            int position, long id) {
            // показываем позиция нажатого элемента
            Toast.makeText(getBaseContext(), "Position = " + position,
Toast.LENGTH_SHORT).show();
        }
        @Override
        public void onNothingSelected(AdapterView<?> arg0) {
        }
    });
}
}

```


Выпускная квалификационная работа выполнена мной совершенно самостоятельно. Все использованные в работе материалы и концепции из опубликованной научной литературы и других источников имеют ссылки на них.

« » _____ Г.

Калиниченко А.С.