Изображение выглядит как текст, цепь

Автоматически созданное описание

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«Санкт-Петербургский государственный технологический институт

(технический университет)»

|  |  |
| --- | --- |
| **ФАКУЛЬТЕТ** | Информационных технологий и управления |
| **КАФЕДРА** | Систем автоматизированного проектирования и управления |
| **НАПРАВЛЕНИЕ** | 09.03.03 Прикладная информатика |
| **НАПРАВЛЕННОСТЬ** | Прикладная информатика в химии |
| **ДИСЦИПЛИНА** | Базы данных |

**Курсовой проект**

|  |  |
| --- | --- |
| **Тема:** | Разработка базы данных «Классификатор аптечных препаратов» |

|  |  |  |  |
| --- | --- | --- | --- |
| Студент: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (подпись) | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (дата) | М. А. Шахов  Группа №405 |
| Руководитель: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (подпись) | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (дата) | Г.В. Кузнецова |

Санкт–Петербург

2022

Введение 3

1 Аналитический обзор 4

1.1 Характеристика предметной области 4

1.2 Обзор программных продуктов, представленных на рынке специализированного программного обеспечения. Обзор технологий и интерфейсов доступа к данным (ODBC, OLE DB, ADO.NET) 8

2 Техническое задание на разработку 11

2.1 Функциональные требования к программному продукту 11

2.2 Пользователи системы и их функциональные обязанности 11

2.3 Правила организация системы доступа пользователей к элементам информационной системы 12

2.4 Требования к системе поиска и фильтрации 12

2.5 Требования эргономики 12

2.6 Требование к программному и аппаратному обеспечению 13

3 Технологическая часть – Технология разработки программного комплекса 16

3.1 Проектирование базы данных 16

3.2 Система поиска и фильтрации 20

3.3 Интерфейсы пользователей 22

Выводы по курсовому проекту 32

Список использованных источников 33

Введение

Вне зависимости от возраста или пола люди сталкиваются с различными заболеваниями и недомоганиями. Каким бы крепким ни было здоровье отдельного человека, время от времени так или иначе лекарственные препараты нужны всем — как больным, так и здоровым. В таких случаях возникает необходимость обратиться в аптеку и приобрести необходимые лекарства.

В современном мире существует огромное разнообразие лекарственных препаратов. Сложность их подбора обусловлена большим количеством доступных средств, множеством фармакологических групп, а также наличием дженериков и других лекарств с одинаковыми активными веществами. Эта сложность постоянно возрастает.

Именно для того, чтобы облегчить поиск информации о лекарственных средствах и их подбор, было принято решение создать базу данных «Классификатор аптечных препаратов» и соответствующее программное обеспечение. Оно позволит взаимодействовать с базой данных, выполняя добавление, редактирование и удаление лекарственных препаратов и их компонентов. Интуитивно понятный интерфейс программы поможет пользователю легко подобрать нужное лекарственное средство.

1 Аналитический обзор

1.1 Характеристика предметной области

**Лекарственное средство**, **лекарственный препарат**, **медикамент**, **лекарство**, **фармпрепарат** — вещество или смесь веществ синтетического или природного происхождения в виде лекарственной формы (таблетки, капсулы, раствора, мази и т. п.), применяемые для профилактики, диагностики и лечения заболеваний.

Следует иметь в виду, что термины **лекарственное средство** и **лекарственный препарат** (фармпрепарат, медикамент) не являются полными синонимами:

* лекарственное средство — по своей сути это действующее вещество (или смесь действующих веществ) [1]. В России в юридическом отношении к лекарственным средствам относятся лекарственные препараты и фармацевтические субстанции;
* лекарственный препарат — это лекарственное средство в определённой дозе в виде определённой лекарственной формы, готовое к применению [1]. Лекарственные препараты нередко содержат кроме действующего вещества, ещё и вспомогательные вещества (формообразователи, растворители, стабилизаторы, консерванты, красители, ароматизаторы и т. д.) [2]. Аналоги лекарственных препаратов с одинаковыми лекарственными средствами, но отличающиеся в зависимости от производителя составами вспомогательных веществ и(или) формой выпуска, внешним видом (к примеру таблеток), или только запатентованным названием, называются дженериками. Лекарственные препараты в зависимости от патентной защиты, могут быть *оригинальными* и *воспроизведёнными* [2], торговые названия могут быть одинаковыми с названием лекарственного средства (МНН — Международное Непатентованное Название) или иметь отличное от него *коммерческое название,* защищённое авторскими правами [2]. Препараты, относящиеся к одной *фармакологической группе*, но имеющие разный состав лекарственного средства именуются *аналогами* [2].

Термин же **лекарство** может употребляться к любому из них и является более просторечным.

Перед употреблением в медицинской практике лекарственные средства должны проходить клинические исследования и получать разрешение к применению [1].

**Классификация**

Существует несколько классификаций, основанных на различных признаках лекарственных средств:

* по химическому строению (например, соединения-производные фурфурола, имидазола, пиримидина и др.);
* по происхождению — природные, синтетические, минеральные;
* по фармакологической группе — наиболее распространённая в России классификация основана на воздействии препарата на организм человека;
* нозологическая классификация — классификация по заболеваниям, для лечения которых используется лекарственный препарат;
* анатомо-терапевтическо-химическая классификация (ATХ) — международная классификация, в которой учитывается фармакологическая группа препарата, его химическая природа и нозология заболевания, для лечения которого предназначен препарат.

Сырьём для получения лекарственных средств служат [2]:

* растения (листья, трава, цветки, семена, плоды, кора, корни) и продукты их обработки (жирные и эфирные масла, соки, камеди, смолы);
* животные — железы и органы животных, сало, воск, тресковая печень, жир овечьей шерсти и другое;
* ископаемое органическое сырьё — нефть и продукты её перегонки, продукты перегонки каменного угля;
* неорганические ископаемые — минеральные породы и продукты их обработки химической промышленностью и металлургией (металлы);
* всевозможные органические соединения — продукты крупной химической промышленности.

**Действие**

Действие лекарственных средств осуществляется главным образом путём изменения физико-химических свойств среды, в которой находятся клеточные элементы организма; при этом действие может иметь характер химического соединения лекарства с элементами организма и в некоторых случаях при непосредственном действии на протоплазму клеток, сопровождаться полным разрушением их. Физиологическим эффектом действия лекарства является либо возбуждение, либо угнетение жизнедеятельности клеточных элементов; при этом большую роль играет доза лекарственного вещества, так как одно и то же лекарство в различных дозах может вызывать разное действие — возбуждать в малых дозах и угнетать (вплоть до паралича) в больших дозах [1].

Существенным моментом является фаза действия лекарственных средств: одни лекарства могут проявлять своё действие в момент проникновения в организм (фаза вхождения по Кравкову), другие — большинство — в период максимальной концентрации в организме (фаза насыщения), третьи — в момент падения концентрации (фаза выхождения); при этом чрезвычайно важным является способность некоторых лекарств к кумуляции, проявляющейся в резком усилении, а иногда и извращении их действия при повторном введении, что объясняется накоплением лекарства в организме и накоплением эффекта действия его[2].

Действие лекарства зависит от возраста, пола, состояния здоровья и индивидуальных особенностей организма лица, принимающего его. Ряд лекарственных средств в соответственно уменьшенной дозе оказывает на детей гораздо более сильное действие, чем на взрослых (часто отравляющее); женщины в период менструации, беременности, лактации реагируют на лекарства иначе, чем обычно; на некоторых лиц лекарство действует ненормально сильно, что объясняется повышенной чувствительностью организма к определённым веществам (Идиосинкразия) [2].

**Способы применения**

Пути введения лекарства в организм весьма разнообразны. Чаще всего используют приём лекарственных средств внутрь перорально. Во избежание быстрого разложения лекарственного препарата, раздражения желудочно-кишечного канала или для достижения наибольшей быстроты действия лекарство вводят с помощью шприца подкожно, внутримышечно или внутривенно. Некоторые лекарства вводят через прямую кишку или путём вдыхания [2].

Наружным применением лекарственных средств считается нанесение их на кожу и на слизистые оболочки глаза, носа, ушей, полости рта, мочеполовых путей (до места входа в мочевой пузырь и до цервикального канала матки), на слизистую оболочку прямой кишки (до места расположения внутреннего сфинктера) [2].

В организме лекарственные средства разрушаются, изменяются и, входя в химические соединения с его солями и жидкостями, теряют свои ядовитые свойства (а иногда, наоборот, приобретают их) и в том или другом виде выводятся из организма через кишечник, почки, дыхательные пути, потовые железы и т. д. [2]

**Оригинальные лекарственные средства, синонимы, «дженерики» и аналоги**

**Оригинальным** лекарством называется препарат, ранее неизвестный и впервые выпущенный на рынок фирмой-разработчиком или патентодержателем. Как правило, разработка и продвижение на рынки нового препарата — очень дорогостоящий и длительный процесс. Из множества известных соединений, а также вновь синтезируемых, методом перебора, на основании баз данных по их свойствам и компьютерного моделирования предполагаемой биологической активности, выявляются и синтезируются вещества, имеющие максимальную целевую активность. После экспериментов на животных, в случае положительного результата, проводятся ограниченные клинические испытания на группах добровольцев. Если эффективность подтверждается, а побочные явления незначительны — лекарство идёт в производство, и на основании результатов дополнительных испытаний уточняются возможные особенности действия, выявляются нежелательные эффекты. Часто самые вредные побочные действия выясняются именно при клиническом применении [1].

В настоящее время практически все новые лекарственные средства патентуются. Патентное законодательство большинства стран предусматривает патентную защиту не только способа получения нового лекарственного средства, но и патентную защиту самого лекарственного средства. В РФ срок действия патента на изобретение, относящееся к лекарственному средству, для применения которого требуется получение в установленном законом порядке разрешения, продлевается федеральным органом исполнительной власти по интеллектуальной собственности по ходатайству патентообладателя на срок, исчисляемый с даты подачи заявки на изобретение до даты получения первого такого разрешения на применение, за вычетом пяти лет. При этом срок, на который продлевается действие патента на изобретение, не может превышать пять лет. После окончания срока действия патента другие производители могут воспроизвести и выпустить на рынок похожее лекарственные средство (так называемый **дженерик**), если докажут биоэквивалентность воспроизведённого и оригинального препаратов. При этом технология производства дженерика может быть любой, но не попадающей под действие существующей в стране патентной защиты. Производитель дженерика не может использовать название бренда для этого лекарства, а только международное непатентованное наименование (МНН) или какое-либо новое, запатентованное им (**синоним**) [1].

С точки зрения химии действующее вещество оригинального лекарства и дженерика одно и то же, но различна технология производства, возможна разная степень очистки. Есть и другие факторы, влияющие на эффективность препарата. Например, долгое время разные фирмы не могли добиться той же эффективности ацетилсалициловой кислоты для дженериков, как у фирмы Bayer — производителя оригинального препарата «аспирин». Оказалось, что дело не только в чистоте сырья, но и в особом способе кристаллизации, дающим в результате особые, более мелкие, кристаллы. Возможен и противоположный результат, когда дженерик получается удачнее, чем оригинальное лекарство [1].

**Аналог** — лекарственное средство на основе вещества, отличного от используемого в сопоставляемом оригинале и, следовательно, с другим МНН. Поэтому, если подбор синонима или дженерика — прерогатива провизора, то для назначения аналога требуется квалификация врача [1].

1.2 Обзор программных продуктов, представленных на рынке специализированного программного обеспечения. Обзор технологий и интерфейсов доступа к данным (ODBC, OLE DB, ADO.NET)

В настоящее время существует множество различных программных интерфейсов, позволяющих с лёгкостью обращаться с информацией. С помощью различных API компании стараются предоставлять пользователям универсальные и удобные средства общения с различными видами БД. Для более подробного описания мы возьмём такие интерфейсы предоставления доступа как: ODBC, OLE DB, ADO.NET.

ODBC – это программный интерфейс доступа к БД, разработанный компанией Microsoft в сотрудничестве с Simba Technologies на основе спецификаций CLI, появившийся благодаря организациям SQL Access Group, X/Open и Microsoft. Впоследствии CLI был стандартизован ISO. Стандарт CLI призван унифицировать программное взаимодействие с СУБД, сделать его независимым от поставщика СУБД и программно-аппаратной платформы [6].

В начале 1990 г. существовало несколько поставщиков БД, каждый из которых имел собственный интерфейс. Если приложению было необходимо общаться с несколькими источниками данных, для взаимодействия с каждой из БД было необходимо написать свой код. Для решения возникшей проблемы Microsoft и ряд других компаний создали стандартный интерфейс для получения и отправки источникам данных различных типов. Он был назван открытым механизмом взаимодействия с БД [7].

C помощью ODBC прикладные программисты могли разрабатывать приложения для использования одного интерфейса доступа к данным, не беспокоясь о тонкостях взаимодействия с несколькими источниками. Это достигается благодаря тому, что поставщики различных БД создают драйверы, реализующие конкретное наполнение стандартных функций из ODBC API с учётом особенностей их продукта. MFC усовершенствовала ODBC для разработчиков приложений. Истинный интерфейс ODBC является обычным процедурным API. Вместо создания простой оболочки процедурного API разработчики MFC создали набор абстрактных классов, представляющих логические сущности в БД [8].

OLE DB – набор COM-интерфейсов, позволяющие приложениям работать с данными разных источников и хранилищ информации. Она был разработана Microsoft в качестве дальнейшего развития технологии доступа к данным и должен был прийти на замену и в качестве преемника ODBC, расширяя набор функций для поддержки более широкого круга нереляционных источников данных, таких как объектно-ориентированные БД или электронные таблицы, и для которых необязательно использовать SQL [6].

OLE DB отделяет хранилище данных от приложения, которое должно иметь доступ к нему через набор абстракций, состоящий из источника данных (DataSource), сессии (Session), команды (Command) и набора строк (Rowset). Это было сделано для предоставления унифицированного доступа к различным видам и источникам данных и изоляцию специфики взаимодействия с конкретным хранилищем. OLE DB концептуально разделена на потребителей (клиентов) и поставщиков (провайдеров). Потребителем является приложение, которому необходим доступ к данным, а поставщик реализует интерфейс доступа к данным и, следовательно, обеспечивает информацией потребителя [9].

ADO.NET является такой технологией работы с данными, которая основана на платформе .NET Framework. Эта технология представляет нам набор классов, через которые мы можем отправлять запросы к БД, устанавливать подключения, получать ответ от БД и производить ряд других операций. Можно сказать, что ADO.NET – это некая управляемая библиотека, составляющие которой используют протокол управления памятью CLR.

Эту библиотеку можно использовать двумя способами, в подключённом режиме и в автономном. При использовании подключённого уровня кодовая база явно подключается к соответствующему хранилищу данных и отключается от него. При таком способе использования ADO.NET происходит взаимодействие с хранилищем данных с помощью объектов подключения, объектов команд и объектов чтения данных. При автономном уровне ADO.NET позволяет отображать реляционные данные с помощью модели объектов в память. Подключение не занимает времени СУБД, подключаясь и отключаюсь автоматически [10-13].

Реализация поставленной задачи требует разработки программного продукта с графическим пользовательским интерфейсом для обеспечения простоты и удобства взаимодействия с базой данных. Для этого был выбран язык программирования C# в силу того, что он имеет сравнительно невысокий порог вхождения, и достаточно часто используется для реализации задач, похожих на задачи данного курсового проекта, а значит располагает необходимыми инструментами и библиотеками. В качестве среды разработки выбрана Visual Studio 2022, так как она имеет максимальную совместимость с выбранным языком программирования и располагает множеством инструментов для быстрой и качественной разработки.

В качестве программной среды для моделирования БД был выбрана свободная реляционная система управления базами данных SQLite, которая представляет собой компактную встраиваемую СУБД. Слово «встраиваемый» означает, что SQLite не использует парадигмы клиент-сервер, то есть движок SQLite не является отдельно работающим процессом, с которым взаимодействует программа, а представляет собой библиотеку, с которой программа компонуется, и движок становится составной частью программы. Таким образом, в качестве протокола обмена используются вызовы функций (API) библиотеки SQLite. Такой подход уменьшает накладные расходы, время отклика и упрощает программу. SQLite хранит всю базу данных (включая определения, таблицы, индексы и данные) в единственном стандартном файле на том компьютере, на котором исполняется программа. Простота реализации достигается за счёт того, что перед началом исполнения транзакции записи весь файл, хранящий базу данных, блокируется; ACID-функции достигаются в том числе за счёт создания файла журнала.

Несколько процессов или потоков могут одновременно без каких-либо проблем читать данные из одной базы. Запись в базу можно осуществить только в том случае, если никаких других запросов в данный момент не обслуживается; в противном случае попытка записи оканчивается неудачей, и в программу возвращается код ошибки. Другим вариантом развития событий является автоматическое повторение попыток записи в течение заданного интервала времени.

2 Техническое задание на разработку

2.1 Функциональные требования к программному продукту

Необходимо разработать базу данных «Классификатор аптечных препаратов», а также создать прикладное программное обеспечение, позволяющее незатруднительно взаимодействовать с БД как с пользовательской части, так и со стороны администрирования.

Программный продукт должен реализовывать следующие задачи:

* Авторизация и регистрация;
* Разграничение прав пользователей, деление их на администраторов, провизоров и пользователей соответственно;
* Свободный доступ к изменению прав пользователей и удаления их со стороны администратора;
* Интерфейс добавления в БД новых активных веществ, производителей, стран, и показаний;
* Добавление новых препаратов и инструкций к ним, удаление, а также возможность редактирования старых;
* Реализовать поиск препаратов по названию, группе, производителю, показанию и активному веществу;
* Предоставить возможность добавления выбранного пользователем препарата в свой список. Реализовать увеличение и уменьшение количества препарата в своем списке, удаление из списка, а также чтение инструкции к нему

2.2 Пользователи системы и их функциональные обязанности

Среди пользователей разработанного программного обеспечения можно выделить 3 группы: администратор, провизор и пользователь.

Пользователь имеет доступ к списку препаратов, может осуществлять поиск нужного лекарственного препарата в БД, с использованием предоставленного ему инструменту фильтрации базы данных, по ключевым словам, различными способами. Он имеет возможность добавления копий определенных препаратов в свой список (заметки) и взаимодействовать с этим списком.

Провизор обладает теми же возможностями, что и пользователь, а также специально наделенными ему правами: добавление в БД новых активных веществ, производителей, стран, и показаний, новых препаратов и инструкций к ним, редактирования и удаление препаратов.

Администратор совмещает в себе права пользователя и провизора. В дополнении к этому он может администрировать пользователей информационной системы, путем удаления пользователей и изменения их прав.

2.3 Правила организация системы доступа пользователей к элементам информационной системы

В ходе реализации программного продукта было предусмотрено разграничение прав доступа к функциям и способам взаимодействия с ИС. Ниже представлена таблица, отображающая матрицу доступа.

Таблица 1 – Матрица доступа

|  |  |  |  |
| --- | --- | --- | --- |
| Данные | Пользователи | | |
| Администратор | Провизор | Пользователь |
| Пользователи | Чтение, запись, обновление, удаление | – | – |
| Активные вещества | Чтение, запись, | Чтение, запись, | Чтение |
| Производитель | Чтение, запись, | Чтение, запись, | Чтение |
| Показания | Чтение, запись, | Чтение, запись, | Чтение |
| Инструкция | Чтение, запись, обновление, удаление | Чтение, запись, обновление, удаление | Чтение |
| Препараты | Чтение, запись, обновление, удаление | Чтение, запись, обновление, удаление | Чтение |

Пользователи добавляются путем регистрации. Все новые зарегистрированные пользователи по умолчанию получают права пользователя. Лишь администратор может установить новые права для пользователя

2.4 Требования к системе поиска и фильтрации

Информационная система обязана поддерживать фильтрацию на форме поиска препаратов. Поиск должен осуществляться по следующим данным:

* По названию препарата (Аквадетрим, Терафлю, Спазмалгон);
* По группе (антибиотики, противовирусные, гомеопатические препараты);
* По действующему веществу (Алверин, Парацетамол, Цетилпиридиния хлорид);
* По производителю (Берлин-Хеми, ГлаксоСмитКлайн, Сентис Фарма);
* По показанию (Тонзиллит, бурсит, заболевания органов малого таза);

2.5 Требования эргономики

Взаимодействие пользователей с прикладным программным обеспечением, входящим в состав автоматизированной информационной системы, будет осуществляться посредством визуального графического интерфейса, который должен быть выполнен по следующим основным правилам:

* Элементы каждой из форм обязаны быть симметричными для фиксации внимания пользователя в нужном направлении;
* Информация должна быть сгруппирована и упорядочена для лёгкого усвоения и понимания пользователем, а общая информация – минимизирована и представляла только то, что необходимо пользователю в конкретный момент;
* Все надписи экранных форм, а также сообщения, выдаваемые пользователю, должны быть на русском языке, так как программа рассчитана для русскоязычных пользователей.

2.6 Требование к программному и аппаратному обеспечению

Программный комплекс должен работать на машине при следующем минимальном аппаратном и программном обеспечении:

Минимальное аппаратное обеспечение: Intel inside CORE i3 5005U, оперативная память: 4096 Мб, DDR3L, 1600 МГц.

Минимальное программное обеспечение: операционная система Microsoft Windows 7 и выше, Microsoft Office, Visual Studio 2022, SQL Server.

Программный комплекс должен быть разработан на языке программирования C#. С# - современный объектно-ориентированный и безопасный язык программирования, позволяющий разработчикам создавать множество типов безопасных и надежных приложений, работающих в экосистеме .NET. Программы C# можно хранить в нескольких исходных файлах. При компиляции программы C# все исходные файлы обрабатываются вместе, при этом они могут свободно ссылаться друг на друга. В C# никогда не используются опережающие объявления, так как порядок объявления, за редким исключением, не играет никакой роли. В C# нет требований объявлять только один открытый тип в одном исходном файле, а также имя исходного файла не обязано совпадать с типом, объявляемом в этом файле.

Программный комплекс должен быть реализован в Microsoft Visual Studio. Visual Studio - полноценная среда разработки с богатым функционалом. Подходит для крупных проектов (web, enterprise). Полная поддержка технологий .NET. Удобное сопровождение и масштабирование кода.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1 – Структура программного обеспечения

Таблица 2 – Минимальные системные требования

|  |  |
| --- | --- |
| Показатель | Значение |
| Тип ЭВМ | Персональный компьютер |
| Тактовая частота процессора, ГГц | 2 |
| Объем оперативной памяти, ГБ | 4 |
| Объем внешней памяти, ГБ | 128 |
| Состав и характеристика периферийных устройств ЭВМ | Мышь |
| Операционная система | Windows 10 |
| Прикладное программное обеспечение, необходимое для функционирования программного комплекса | .NET Framework 4.7.2 или выше |

Таблица 3 – Характеристика программного обеспечения

|  |  |
| --- | --- |
| Показатель | Значение |
| Среда разработки | Microsoft Visual Studio 2022 |
| Технология программирования | ООП |
| Язык программирования | С# |
| Модель описания данных | Реляционная |
| СУБД | SQLite |
| Количество таблиц в БД | 11 |
| Типы данных | INT, TEXT, BLOB |
| Текущий объем БД, КиБ | 444 |

3 Технологическая часть – Технология разработки программного комплекса

3.1 Проектирование базы данных

Были созданы следующие таблицы:

* ActiveIngredient;
* Class;
* Manufacturer
* Country;
* Indication
* Drug;
* DrugActiveIngredient;
* DrugIndication;
* User;
* UserMode
* UserDrug

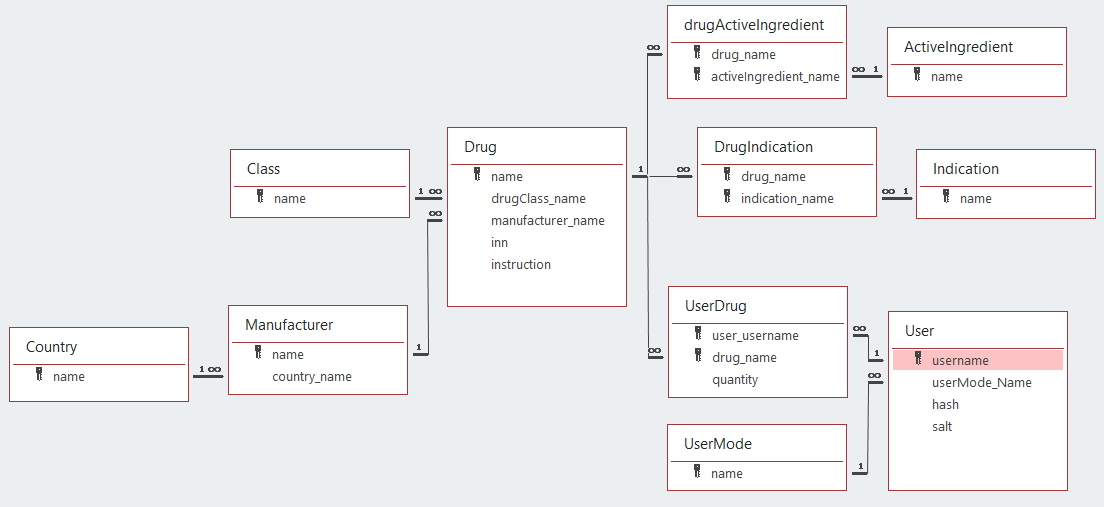


Рисунок 2 – Даталогическая модель БД

Ниже приведены структуры различных таблиц. Жирным шрифтом выделены ключи PK, шрифт с курсивом – ключи FK

Таблица 4 – Структура таблицы «ActiveIngredient»

|  |  |
| --- | --- |
| Имя поля | Тип данных |
| **name** | TEXT |

Таблица 5 – Структура таблицы «Class»

|  |  |
| --- | --- |
| Имя поля | Тип данных |
| **name** | TEXT |

Таблица 6 – Структура таблицы «Country»

|  |  |
| --- | --- |
| Имя поля | Тип данных |
| **name** | TEXT |

Таблица 7 – Структура таблицы «Drug»

|  |  |
| --- | --- |
| Имя поля | Тип данных |
| **name** | TEXT |
| *drugClass\_name* | TEXT |
| *manufacturer\_name* | TEXT |
| inn | TEXT |
| instruction | TEXT |

Таблица 8 – Структура таблицы «DrugActiveIngredient»

|  |  |
| --- | --- |
| Имя поля | Тип данных |
| ***drug\_name*** | TEXT |
| ***activeIngredient\_name*** | TEXT |

Таблица 9 – Структура таблицы «DrugIndication»

|  |  |
| --- | --- |
| Имя поля | Тип данных |
| ***drug\_name*** | TEXT |
| ***indication\_name*** | TEXT |

Таблица 10 – Структура таблицы «Indication»

|  |  |
| --- | --- |
| Имя поля | Тип данных |
| **name** | TEXT |

Таблица 11 – Структура таблицы «Manufacturer»

|  |  |
| --- | --- |
| Имя поля | Тип данных |
| **name** | TEXT |
| *country\_name* | TEXT |

Таблица 12 – Структура таблицы «User»

|  |  |
| --- | --- |
| Имя поля | Тип данных |
| **username** | TEXT |
| *userMode\_name* | TEXT |
| hash | BLOB |
| salt | BLOB |

Таблица 13 – Структура таблицы «UserDrug»

|  |  |
| --- | --- |
| Имя поля | Тип данных |
| ***user\_name*** | TEXT |
| ***drug\_name*** | TEXT |
| quantity | INTEGER |

Таблица 14 – Структура таблицы «UserMode»

|  |  |
| --- | --- |
| Имя поля | Тип данных |
| **name** | TEXT |

Таблица 15 – Пример заполнения базы данных

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Значение | Описание |
| username | TEXT | Admin | Имя пользователя |
| userMode\_name | TEXT | admin | Права пользователя |
| hash | BLOB | 9dde416a84b404667b7eb5d16fa6d12f  b0e93ca22ff5d3a6bbfecc2f4f2bbabf | Хэш |
| salt | BLOB | cb6d6e8f473cb5ac1086c8a6808b679e  f48e884f3ef2dea3135fa25d2c9fe65e | Соль |
| user\_username | TEXT | Admin | Имя пользователя |
| quantity | INTEGER | 1 | Количество препарата в своем списке |
| drug\_name | TEXT | Афобазол® | Имя препарата в своем списке |
| drugClass\_name | TEXT | Нервная система | Название группы препарата |
| manufacturer\_name | TEXT | Фармстандарт-Лексредства | Имя производителя |
| inn | TEXT | Фабомотизол | Международное название |
| instruction | TEXT | \*ТЕКСТ ИНСТРУКЦИИ\* | Инструкция препарата |

Ниже, на рисунке 3, представлена UML-диаграмма, отражающая возможности пользователей.

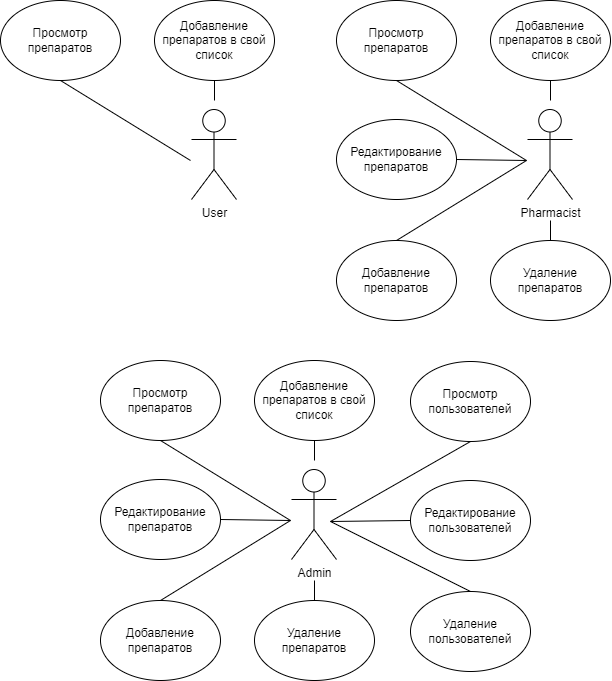


Рисунок 3 - UML-диаграмма

3.2 Система поиска и фильтрации

Для работы с базой данных в информационной системе были реализованы следующие типы запросов: добавление, удаление и обновление записи сущности. Также использовались запросы на выборку и вставку.

**Запросы для выборки:**

SELECT d.name, d.manufacturer\_name, m.Country\_name as Manufacturer\_country, d.inn

FROM Drug d, Manufacturer m

WHERE d.manufacturer\_name = m.name

GROUP BY d.name

SELECT d.name, d.manufacturer\_name, m.Country\_name as Manufacturer\_country, d.inn

FROM Drug d, Manufacturer m INNER JOIN DrugActiveIngredient da on d.name = da.drug\_name

WHERE d.manufacturer\_name = m.name AND da.activeIngredient\_name = @activeIngredient

GROUP BY d.name

SELECT d.name, d.manufacturer\_name, m.Country\_name as Manufacturer\_country, d.inn

FROM Drug d, Manufacturer m INNER JOIN DrugIndication di on d.name = di.drug\_name

WHERE d.manufacturer\_name = m.name AND di.indication\_name = @indication

GROUP BY d.name

SELECT d.name, d.manufacturer\_name, m.Country\_name as Manufacturer\_country, d.inn

FROM Drug d, Manufacturer m

WHERE d.manufacturer\_name = m.name AND d.drugClass\_name = @drugClass

GROUP BY d.name

SELECT d.name, d.manufacturer\_name, m.Country\_name as Manufacturer\_country, d.inn

FROM Drug d INNER JOIN Manufacturer m ON d.manufacturer\_name = m.name

WHERE d.manufacturer\_name LIKE @manufacturer

GROUP BY d.name

SELECT dai.activeIngredient\_name as 'name'

FROM DrugActiveIngredient dai

WHERE dai.drug\_name = @drugName

GROUP BY dai.activeIngredient\_name

SELECT d.drugClass\_name

FROM Drug d

WHERE d.name = @drugName

GROUP BY d.drugClass\_name

SELECT di.indication\_name as 'name'

FROM DrugIndication di

WHERE di.drug\_name = @drugName

GROUP BY di.indication\_name

SELECT \*

FROM UserDrug ud

WHERE ud.user\_username = @username AND ud.drug\_name = @drugName

SELECT d.name, d.manufacturer\_name, m.Country\_name as Manufacturer\_country, d.inn

FROM Drug d INNER JOIN Manufacturer m ON d.manufacturer\_name = m.name

WHERE d.name LIKE @name

GROUP BY d.name

SELECT d.name, d.manufacturer\_name, m.Country\_name as Manufacturer\_country, d.inn, ud.quantity

FROM Drug d INNER JOIN Manufacturer m ON d.manufacturer\_name = m.name INNER JOIN UserDrug ud ON d.name = ud.drug\_name

WHERE ud.user\_username = @username

GROUP BY d.name

SELECT \* FROM Country

GROUP BY Country.name

SELECT \* FROM Manufacturer

GROUP BY Manufacturer.name

SELECT \* FROM ActiveIngredient

GROUP BY ActiveIngredient.name

SELECT \* FROM Indication

GROUP BY Indication.name

SELECT \* FROM Class

GROUP BY Class.name

SELECT d.instruction FROM Drug d WHERE d.name = @name

**Запросы на обновление:**

UPDATE User

SET userMode\_name = @UserMode\_name

WHERE username = @Username

UPDATE UserDrug

SET quantity = @Quantity

WHERE user\_username = @User\_username AND drug\_name = @Drug\_name

**Запросы на вставку:**

INSERT INTO User (username, UserMode\_name, hash, salt)

values (@Username, @UserMode\_name, @Hash, @Salt)

INSERT INTO Country (Name)

values (@Name)

INSERT INTO Class (Name)

values (@Name)

INSERT INTO Manufacturer (Name, country\_name)

values (@Name, @Country\_name)

INSERT INTO ActiveIngredient (Name)

values (@Name)

INSERT INTO Indication (Name)

values (@Name)

INSERT INTO DrugActiveIngredient(drug\_name, activeIngredient\_name)

VALUES(@drug\_name, @activeIngredient\_name)

INSERT INTO DrugIndication(drug\_name, indication\_name)

VALUES(@drug\_name, @indication\_name)

INSERT INTO UserDrug (user\_username, drug\_name, quantity)

values (@User\_username, @Drug\_name, @Quantity)

INSERT INTO Drug (name, drugClass\_name, manufacturer\_name, inn, instruction)

values (@Name, @DrugClass\_name, @Manufacturer\_name, @Inn, @Instruction)

**Запросы на удаление:**

DELETE FROM UserDrug WHERE drug\_name = @Name;

DELETE FROM DrugActiveIngredient WHERE drug\_name = @Name;

DELETE FROM DrugIndication WHERE drug\_name = @Name;

DELETE FROM Drug WHERE name = @Name;

DELETE FROM UserDrug WHERE user\_username = @UserName;

DELETE FROM User WHERE username = @UserName"

DELETE FROM UserDrug WHERE drug\_name = @Drug\_name

3.3 Интерфейсы пользователей

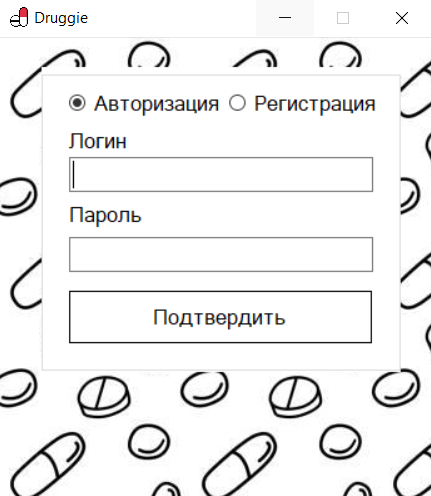


Рисунок 4 – Окно авторизации

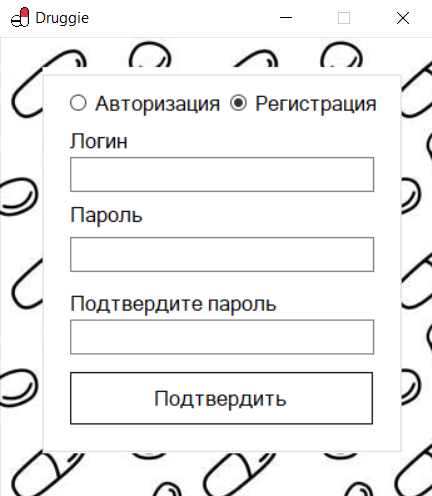


Рисунок 5 – Окно Регистрации

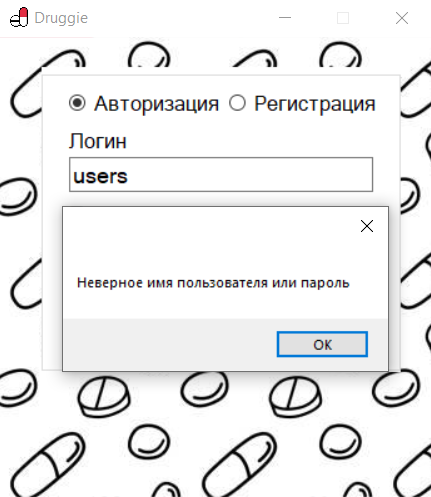


Рисунок 6 – Пример неверных данных при авторизации

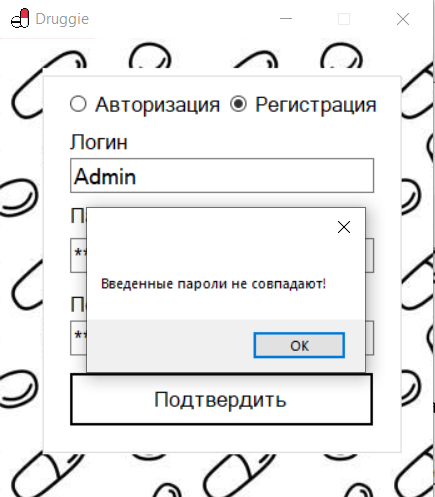


Рисунок 7 – Пример несовпадающих паролей при регистрации

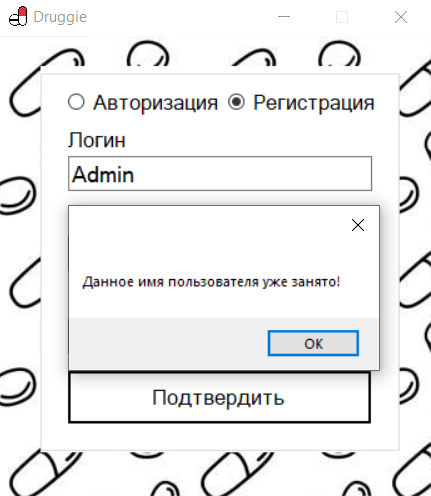


Рисунок 8 – Пример использования занятого имени пользователя при регистрации

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 9 – Панель администратора



Рисунок 10 – Панель провизора

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 11 – Панель пользователя

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 23 – Пример поиска препарата по действующему веществу

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 12 – Пример поиска препарата по показанию

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 13 – Пример поиска препарата по группе

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 14 – Пример поиска препарата по производителю

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 15 – Пример поиска препарата по названию

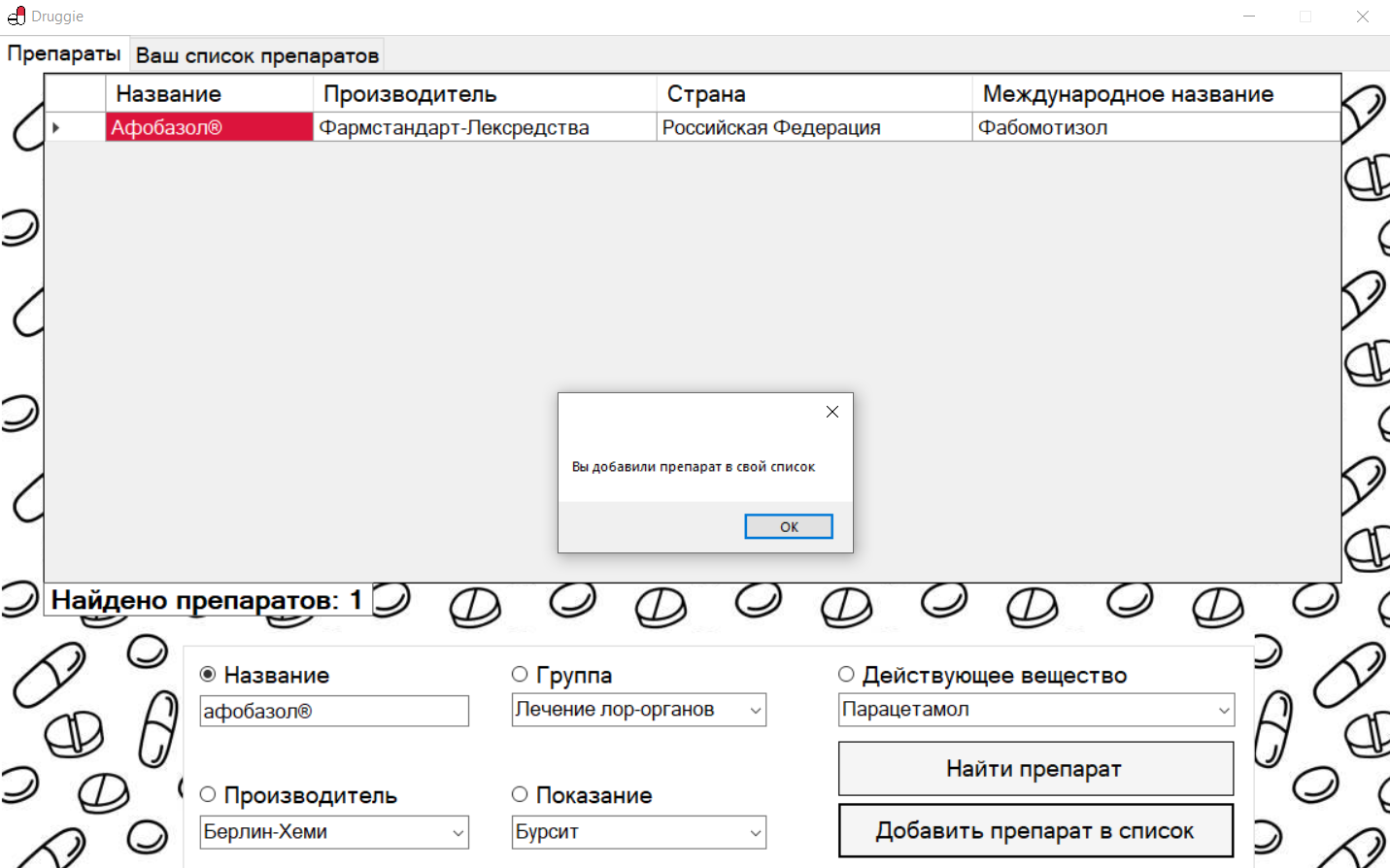


Рисунок 16 – Пример добавления препарата в свой список

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 17 – Окно препаратов, которые пользователь добавил себе в список

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 18 – Пример увлечения количества препарата в пользовательском списке

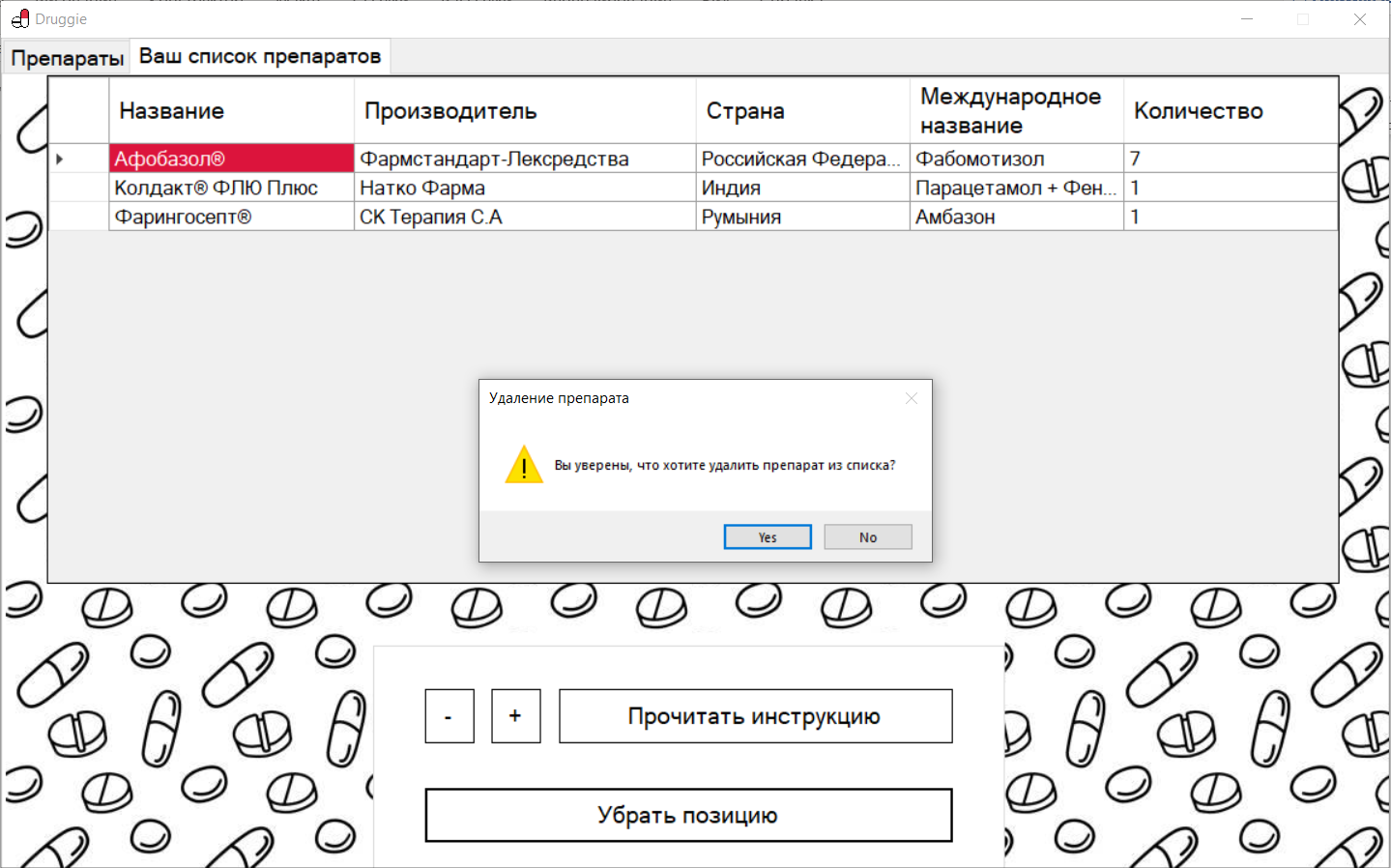


Рисунок 19 – Окно удаления препарата из пользовательского списка

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 20 – Окно инструкции на примере препарата «Афобазол®»

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 21 – Форма добавления препарата

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 22 – Пример редактирования препарата «Аквадетрим»

Выводы по курсовому проекту

В процессе выполнения данной курсовой работы была создана модель и структура базы данных «Классификатор аптечных препаратов», разработан программный комплекс взаимодействия с ней. В ходе разработки была поставлена цель и определены задачи, которые было необходимо решить в процессе создания программного комплекса. Итоги работы над задачами таковы:

1. Исследована предметная область
2. Разработана схема информационных потоков информационной системы;
3. Построена функциональная структура информационной системы;
4. Составлена UML-диаграмма прецедентов использования;
5. Построена даталогическая модель информационной системы;
6. Была произведена нормализация БД;
7. Создана структура интерфейса;
8. Реализовано приложения на языке C#;
9. Произведено тестирование приложения;
10. Проанализированы результаты работы программного комплекса;
11. Оформлена необходимая документация по проекту, следуя установленным требованиям к оформлению работы.

Также для базы данных были добавлены:

1. Таблицы, соответствующие разработанной даталогической модели данных;
2. данные, демонстрирующие возможности работы информационной системы.

Данный проект может быть расширен до аптечного приложения, путем добавления цен на конкретные препараты, корзины, личного кабинета и платежной системы.

Все поставленные цели были достигнуты, курсовой проект реализован в полном соответствии с заданием.

Список использованных источников

1 Малая медицинская энциклопедия [Текст]: [в 6 т.] / гл. ред. В. И. Покровский. – Москва : Советская энциклопедия, 1991–1996. - 27 см.; ISBN 5-85270-013-4

2. Стефанкина Е. В. Новейший справочник. Лекарственные средства. — М.: ОЛМА-ПРЕСС, 2006. — 960 с. ISBN 5-224-05129-0.

3 Скит, Дж. C# для профессионалов. Тонкости программирования / Дж. Скит. – Москва : Вильямс, 2017. – 608 с. – ISBN 978-5-8459-1909-0.

4 Троелсен, Э. У. Язык программирования C# 7 и платформы .NET и .NET Core / Э. У. Троелсен, Ф. Джепикс. – 8-е изд. – Москва : Диалектика, 2019. – 1328 с. – ISBN 978-5-6040723-1.

5 Албахари, Дж. C# 7.0. Справочник / Дж. Албахари, Б. Албахари. – Москва : Вильямс, 2018. – 1024 с. – ISBN 978-5-6040043-7-1.

6 Норенков, И. П. Автоматизированные информационные системы: учебное пособие / И. П. Норенков. – Москва : Изд-во МГТУ им. Н.Э. Баумана, 2011. – 342 с. – ISBN 978-5-7038-3446-6.

7 Советов, Б. Я. Базы данных. Учебник / Б. Я. Советов, В. В. Цехановский, В. Д. Чертовской. – 2-е изд. – Москва : Издательсво Юрайт, 2017. – 463 с. – ISBN 978-5-534-07217-4.

8 Дейт, К. Дж. Введение в системы баз данных / К. Дж. Дейт. – 8-е изд. – Москва : Вильямс, 2016. – 1328 с. – ISBN 5-8459-0788-8.

9 Гарсиа-Молина, Г. Системы баз данных. Полный курс: пер. с англ. / Г. Гарсиа-Молина, Дж. Ульман, Дж. Уидом. – Москва : Вильямс, 2003. – 1088 с. – ISBN 5-8459-0384-Х.

10 Крёнке, Д. Теория и практика построения баз данных. – 8-е изд.: учебное пособие. – Санкт-Петербург : Питер, 2003. – 800 с. – ISBN 5-94723-275-8.

11 Грофф, Дж. Р. SQL. Полное руководство / Джеймс. Р. Грофф, Пол. Н. Вайнберг, ЭндрюДж. Оппель. – Москва : Вильямс, 2014. – 960 с. – ISBN 978-5-8459-1654-9.

12 Буч, Г. Введение в UML от создателей языка / Г. Буч, Дж. Рамбо, А. Якобсон. – Москва : ДМК Пресс, 2015. – 496 с. – ISBN 978-5-94074-644-7.

13 Плонский, В. Ю. Проектирование баз данных в СУБД MySQL: Практикум. / В. Ю. Плонский, Г. В. Кузнецова. – Санкт-Петербург : СПбГТИ(ТУ), 2019. – 54 с.