СОДЕРЖАНИЕ

[Список сокращений 7](#_Toc73976510)

[Введение 8](#_Toc73976511)

[1 Теоретическая часть 11](#_Toc73976512)

[1.1 Автоматизация бизнеса 11](#_Toc73976513)

[1.2 Описание предметной области 12](#_Toc73976514)

[1.3 Требования к программному продукту «Автоматизация аптеки» 15](#_Toc73976515)

[1.4 Анализ аналогов и прототипов 15](#_Toc73976516)

[2 Постановочная часть 19](#_Toc73976517)

[2.1 Постановка задачи проектирования 19](#_Toc73976518)

[2.2 Требования к интерфейсу 19](#_Toc73976519)

[2.3 Описание алгоритма работы аптеки 21](#_Toc73976520)

[2.4 Описание входных и выходных данных 22](#_Toc73976521)

[3 Проектная часть 24](#_Toc73976522)

[3.1 Краткая характеристика среды разработки Visual Studio 24](#_Toc73976523)

[3.2 Краткая характеристика языка SQL 26](#_Toc73976524)

[3.3 Краткая характеристика языка программирования C# 28](#_Toc73976525)

[3.4 Краткая характеристика БД 31](#_Toc73976526)

[3.5 Windows Forms 36](#_Toc73976527)

[4 Экономическая часть 39](#_Toc73976528)

[4.1 Расчет расходов на программу 39](#_Toc73976529)

[5 Эксплуатация 43](#_Toc73976530)

[5.1 Требования к аппаратному обеспечению 43](#_Toc73976531)

[**5.2 Роли пользователей** 43](#_Toc73976532)

[5.3 Инструкция пользователю 45](#_Toc73976533)

[6 Мероприятия по технике безопасности и охране окружающей среды 53](#_Toc73976534)

[6.1 Общие принципы и законодательство по охране труда 53](#_Toc73976535)

[6.2 Техника безопасности при работе с персональным компьютером 55](#_Toc73976536)

[6.3 Противопожарная безопасность при работе с персональным компьютером 56](#_Toc73976537)

[6.4 Мероприятия, проводимые для охраны окружающей среды 57](#_Toc73976538)

[6.5 Требования к освещению помещений и рабочих мест 59](#_Toc73976539)

[6.6 Режим труда и отдыха при работе за компьютером 61](#_Toc73976540)

[Заключение 64](#_Toc73976541)

[Список использованной литературы 66](#_Toc73976542)

[ПРИЛОЖЕНИЕ 68](#_Toc73976543)

# Список сокращений

Список сокращений и их значения, которые используются в пояснительной записке дипломной работы, представлен в таблице 1.

Таблица 1 «Список сокращений»

|  |  |
| --- | --- |
| Сокращение | Значение |
| 1 | 2 |
| CD | Compact disc |
| COM | Component Object Model |
| MSSCCI | Microsoft Visual Studio Team Foundation Server |
| SQL | Structured Query Language |
| SDK | Software Development Kit |
| SSMS | SQL Server Management Studio |
| XML | Extensible Markup Language |
| БД | База данных |
| КЕО | Коэффициент естественной освещенности |
| ПК | Персональный компьютер |
| ПЭВМ | Персональная электронно-вычислительная машина |
| Тг. | Тенге |

# Введение

В послании народу Казахстана Назарбаев подчеркнул, что развитие цифровой индустрии обеспечит импульс всем другим отраслям. В этой связи Первый президент поставил задачу по развитию новых индустрий, которые создаются с применением цифровых технологий.

Уровень технического развития персональных компьютеров позволяет использовать их как в быту, так и в различных организациях. Современное программное обеспечение по вводу, обработке, хранению и модификации данных позволяет сэкономить ощутимое количество труда и времени. Программные продукты позволяют легко оперировать с огромными массивами информации, выполнять точный и полный анализ данных и выводить общие или скрупулезные отчеты по итогам работы, проводить быстрый поиск необходимой информации и многое другое. Они позволяют автоматизировать многие операции в организации и упростить систему документооборота за счет замены бумажных документов их электронными копиями. На сегодняшний день рынок фармацевтических товаров в Казахстане является одним из самых активных и быстро развивающихся. С помощью различных инструментов этих программных продуктов можно разработать автоматизированную информационную систему, поддерживающую и упрощающую работу аптеки.

Автоматизация бизнеса - это совокупность работающих средств (программ и устройств), обеспечивающих минимизацию рутины, оптимизацию трудовых и производственных ресурсов с целью наращивания продуктивности и эффективности всех бизнес-процессов.

Целью дипломного проекта является: разработать автоматизированную информационную систему для сокращения временных затрат сотрудников аптеки на учет, формирование документации и отчетности аптеки.

Для осуществления поставленной цели необходимо реализовать следующие задачи:

1. выявить актуальные проблемы существующих продуктов;
2. разработать автоматизированную информационную систему для учета лекарств в аптеке на C#;
3. провести теоретический анализ научно-методической литературы по аптечному бизнесу;
4. автоматическое формирование потребности аптеки;
5. решение проблем мошенничества среди персонала;

Назначение: Программа для автоматизации работы аптек и медицинских учреждений, которая позволяет оптимизировать все ручные процессы, а также отслеживает все этапы движения лекарственных средств от поставщика до конечного потребителя [17].

В предлагаемом дипломном проекте содержатся следующие разделы:

Раздел «Теоретическая часть» содержит:

* информацию об автоматизации бизнеса;
* описание предметной области;

Раздел «Постановочная часть» содержит:

* формулировку задачи, которую предстоит выполнить на дипломном проектировании;
* требования к интерфейсу разрабатываемого программного продукта: графический интерфейс пользователя, принципы разработки, ориентированные на пользователя;
* описание алгоритмов работы аптеки: графическое представление работы аптеки;

Раздел « Проектная часть» содержит:

* технологию создания аптеки;
* краткую характеристику языков C#, SQL;

Раздел «Экономическая часть»

* расчет расходов на программу;
* раздел «Эксплуатация» содержит;
* требования к аппаратному обеспечению: минимальная конфигурация аппаратных ресурсов, перечень необходимых устройств, перечень необходимых материалов;
* инструкцию пользователю: развернутая инструкция по эксплуатации программного продукта;
* инструкцию программисту: условия и правила установки программы, сведения по настройке и модификации.

Раздел «Мероприятия по технике безопасности и охране труда» содержит:

* обеспечение техники безопасности и охрана труда оператора ЭВМ.

Раздел «Заключение» содержит:

* краткие выводы по результатам выполненной работы;
* предложения по её использованию, включая внедрение в учебные заведения, возможности изменения и доработки.

Раздел «Список литературы» содержит список литературы, с помощью которой был разработан дипломный проект.

Приложения состоят из следующих документов:

* листинг основного модуля программы
* СD диск с откомпилированной программой

# 1 Теоретическая часть

## 1.1 Автоматизация бизнеса

Автоматизация бизнеса позволяет улучшить работу предприятия в разы, рутинные процессы упраздняются, прибыль растет. Но для каждого процесса есть свои тонкости и нюансы. Важно грамотно подобрать автоматизированные инструменты. О них мы и поговорим в сегодняшнем материале.

Когда рутинные задачи бизнеса и стандартные операции полностью либо частично возлагаются на специализированное программное или программно-аппаратное обеспечение, говорят об автоматизации данного бизнеса. В результате высвобождается значительное количество человеческих и финансовых ресурсов, повышается производительность труда, формируется более эффективная стратегия управления [13].

Существует две основных области внедрения автоматизации:

* базовые бизнес-процессы (управление продажами, клиентское обслуживание). Задача в данном случае с помощью автоматизации повысить объем выпускаемой продукции, увеличить продажи и повысить доходность бизнеса.
* сервисные и оперативные процессы (бухучет, ведение отчетности, делопроизводство). Здесь смысл состоит в сокращении затрат ресурсов на рутинные операции. На повышение доходности компании автоматизация данных процессов влияет лишь опосредованно.

Автоматизация управления бизнесом решает следующие общие задачи:

* более эффективно ведется оперативная деятельность компании, включая учет и контроль;
* значительно упрощается документооборот - автоматически формируются документы любого типа при работе с партнерами (деловые предложения, различные накладные и счета-фактуры, акты сверки);
* также возможно создавать отчеты в любом разрезе деятельности и за любой временной период;
* сокращаются затраты на штат сотрудников, увеличивается их продуктивность благодаря освобождению от рутинных обязанностей, более эффективно используется рабочее время;
* влияние человеческого фактора на основные процессы бизнеса сводится к минимуму;
* гарантируется безопасность хранения информации;
* повышается общее качество работы с клиентами.
* При условии грамотно выполненной автоматизации полностью реализуются поставленные задачи. Перечислим их еще раз:
* эффективная поддержка оперативной работы с внедрением учета и контроля;
* быстрая подготовка всех документов как для партнеров, так и для клиентов компании;
* автоматическое формирование отчетов, демонстрирующих положение дел во всех областях деятельности организации;
* оптимизация затрат на персонал, разгрузка сотрудников от рутинных задач и повышение эффективности работы штата;
* оптимизация основных бизнес-процессов;
* сведение к минимуму влияния человека на бизнес в целом;
* обеспечение безопасного хранения информации с ограничением доступа к ней для каждого сотрудника;
* улучшение качества и скорости клиентского обслуживания.

## 1.2 Описание предметной области

Аптека (греческий apotheke склад, кладовая) - учреждение, осуществляющее приготовление, хранение и отпуск лекарств, перевязочных материалов, предметов ухода за больными и других медицинских товаров. Аптеку традиционно рассматривают как учреждение здравоохранения, а её деятельность формулируют как «оказание фармацевтической помощи населению». Фармацевтическая помощь включает в себя процедуру консультирования врача и пациента с целью определения наиболее эффективного, безопасного и экономически оправданного курса лечения. Аптека осуществляет следующие функции:

* реализация населению готовых лекарственных препаратов (в том числе гомеопатических препаратов) по рецептам и без рецептов врача, учреждениям здравоохранения по требованиям или заявкам;
* изготовление лекарственных препаратов по рецептам врачей и требованиям учреждений здравоохранения, изготовление внутриаптечной заготовки в соответствии с утверждёнными прописями и фасовку лекарственных препаратов и лекарственного растительного сырья с последующей их реализацией;
* отпуск лекарственных препаратов бесплатно или со скидкой отдельным группам населения в соответствии с действующим законодательством Российской Федерации и на основании заключённых договоров с территориальными органами управления здравоохранением, лечебно-профилактическими учреждениями и страховыми компаниями;
* реализация лекарственного растительного сырья в заводской упаковке; изделий медицинского назначения (в частности, предметов ухода за больными, изделий медицинской техники, в том числе профилактического назначения, диагностических средств, лечебно-профилактического белья, чулочных изделий, бандажей, предметов ухода за детьми, аптечек первой медицинской помощи и других); дезинфицирующих средств; предметов (средств) личной гигиены (в частности, средств ухода за кожей, волосами, ароматических масел и других); оптики (в частности, готовых очков, средств по уходу за очками и других); минеральных вод (натуральных и искусственных); лечебного, детского и диетического питания (в частности, пищевых добавок лечебного и профилактического назначения и других); косметическую и парфюмерную продукцию (далее - товары, разрешённые к отпуску из аптечных организаций);
* отпуск предметов через пункт проката в соответствии с установленным порядком;
* предоставление населению необходимой информации по надлежащему использованию и хранению лекарственных препаратов в домашних условиях;
* предоставление медицинским работникам учреждений здравоохранения, просвещения, социального обеспечения и др. необходимой информации об имеющихся в аптеке лекарственных препаратах, а также о новых лекарственных препаратах;
* оказание первой медицинской помощи;
* оказание консультативной помощи в целях обеспечения ответственного самолечения.

Аптека взаимодействует с поставщиками в лице менеджера, осуществляющими заказы на поставку лекарств, бытовой химии, медицинского оборудования и косметических средств. Так же в аптеке есть свой штат сотрудников: фармацевты, бухгалтера, грузчики-водители, менеджеры. Клиент, приходя в аптеку, взаимодействует с фармацевтом, который, в свою очередь, осуществляет продажу. Операции с денежными средствами и отбивку чека осуществляет так же фармацевт. При покупке товара, клиент получает чек с указанием товара и его стоимости. В аптеке ведется документация: кассовая книга (сумма проданного товара, которая отдается бухгалтеру, а затем передается в налоговую для отчетности), приходная накладная, ведется учет доходов и расходов аптеки [11].

В аптеке ведется документация: кассовая книга (сумма проданного товара, которая отдается бухгалтеру, а затем передается в налоговую для отчетности), приходная накладная, ведется учет доходов и расходов аптеки.

## 1.3 Требования к программному продукту «Автоматизация аптеки»

1. Управления процессом формирования ассортимента;

* внедрения категорийного менеджмента;
* управлением доходностью товарных групп с учетом доходности и выплат от фарм производителей;

2. Управления системой ценообразования, саморегулирующейся и конкурентоспособной;

3. Управления программой лояльности (дисконтной программой) и коммуникации с клиентом;

4. Запуска терапевтических цепочек;

5. Алгоритмизации системы оплаты труда, расчёта премии сотрудников на основании их KPI;

6. Визуализации премиальных баллов на препараты для фармацевтов;

7.Визуализации ключевых показателей эффективности работы сотрудников;

8. Получение, обработка, хранение и использование технической информации;

9. Увеличение дохода;

10. Увеличение эффективности рабочих часов за счет перевода ручного труда в автоматизированный;

11. Сокращение количества человеческих ошибок;

12. Возможность просмотра рецептов дистанционно;

13. Повышение качества работы с клиентами.

## 1.4 Анализ аналогов и прототипов

Изучены возможности следующих программных продуктов для учета лекарственных средств:

* «1С: Медицина. Больничная аптека»;
* «Учет лекарственных средств 4.1»;
* «1С: Розница 8.Аптека».

Прикладное решение «1С: Медицина. Больничная аптека» предназначено для автоматизации деятельности больничных аптек медицинских организаций любой формы собственности (хозрасчетные и бюджетные).

Возможности программы:

* ведение учета номенклатуры аптечных товаров по сериям, срокам годности, документам партии (документам поступления в аптеку);
* сплошной предметно-количественный учет и контроль движения аптечных товаров как на складах аптеки, так и в отделениях медицинской организации;
* ведение учета по складам (суммового и количественного) и по местам хранения склада (количественного);
* возможность раздельного учета по источникам финансирования;
* поддержка процесса изготовления по лекарственным прописям и фасовки готовых аптечных товаров из упаковки поставщика в упаковку больничной аптеки;
* инвентаризация товарно-материальных ценностей;
* формирование развернутых аналитических отчетов по движению аптечных товаров;
* формирование регламентированной бухгалтерской отчетности, отчетности для внешних потребителей и контролирующих органов;
* загрузка сведений о препаратах с диска «ИТС МЕДИЦИНА», который содержит материалы Государственного реестра лекарственных средств;

Программа отклонена по следующим причинам:

* сложность в освоении;
* продажа лекарственных средств осуществляется всем клиентам не зависимо от наличия рецепта;
* стоимость программного продукта на одно рабочее место 120 000 тг.

Программа «Учет лекарственных средств 4.1» позволяет вести учёт лекарственных средств с последующем выводом отчётов об окончании срока годности препаратов, количества израсходованных медикаментов за выбранный период времени, наличия медикаментов и т.д. Программа проста в управлении и не требует специального обучения для начала работы.

Программа отклонена по следующим причинам:

* поступления товаров и расход осуществляется без специальных документов;
* нельзя установить постоянную цену на номенклатуру;
* отсутствие назначения скидок на номенклатуру;
* продажа лекарственных средств осуществляется всем клиентам не зависимо от наличия рецепта;
* отсутствует справочная информация по категориям лекарств, действующим веществам, фармакологическим действиям и т.д.;

Программный продукт «1С: Розница 8. Аптека» предназначен для комплексной автоматизации торговой деятельности аптек, аптечных киосков и аптечных розничных сетей, ориентированных на продажу лекарственных средств, БАДов, средств гигиены, медицинских аппаратов и т.д.

Возможности программы:

* ведение учета номенклатуры аптечных товаров по сериям, срокам годности, документам партии (документам поступления в аптеку);
* сплошной предметно-количественный учет и контроль движения аптечных товаров, как на складах аптеки, так и в отделениях медицинской организации;
* ведение учета по складам (суммового и количественного) и по местам хранения склада (количественного);
* возможность раздельного учета по источникам финансирования;
* поддержка процесса изготовления по лекарственным прописям и фасовки готовых аптечных товаров из упаковки поставщика в упаковку больничной аптеки;
* инвентаризация товарно-материальных ценностей;
* формирование развернутых аналитических отчетов по движению аптечных товаров;
* формирование регламентированной бухгалтерской отчетности, отчетности для внешних потребителей и контролирующих органов;
* загрузка сведений о препаратах с диска «ИТС МЕДИЦИНА», который содержит материалы Государственного реестра лекарственных средств;
* возможность установки постоянных цен на номенклатуру;

Программа отклонена по следующим причинам:

* сложность в освоении;
* возможна продажа просроченного препарата;
* стоимость программного продукта на одно рабочее место 102 000тг.

# 2 Постановочная часть

## 2.1 Постановка задачи проектирования

Целью выпускной квалификационной работы является разработка автоматизированной информационной системы для сокращения временных затрат сотрудников аптеки на учет, формирование документации и отчетности аптеки. Исходя из поставленной цели, были сформулированы следующие задачи:

* провести сбор информации об объекте исследования;
* провести анализ предметной области;
* рассмотреть функциональные задачи будущих пользователей;
* изучить литературу по данной теме;
* выполнить технико-экономическое обоснование работы;
* организация условий труда;
* разработать программный продукт;
* протестировать программный продукт;
* внедрить программный продукт. На основе анализа предметной области к программному продукту были предъявлены следующие требования;
* простой и удобный пользовательский интерфейс;
* добавление номенклатуры;
* ввод и хранение данных номенклатуры;
* редактирование, изменение, удаление существующих данных номенклатуры;
* расчет налога на добавленную стоимость на номенклатуру;
* создания отчетной документации по выбранным критериям.

## 2.2 Требования к интерфейсу

При разработке программы необходимо выполнить следующие требования:

* Имена идентификаторов переменных, типов данных, подпрограмм, модулей должны соответствовать выполняемым ими функциям;
* Имена идентификаторов должны быть локальными для программного элемента (процедуры, функции, класса, модуля), в котором они используются.
* Логически связанные части кода программы, использующиеся более одного раза (например, сортировка массива), целесообразно выделять в отдельные подпрограммы;
* Набор подпрограмм, выполняющих логически связанное действие (например, работа с файлом определенного типа), и связанных с ними глобальных переменных, констант и типов данных следует выделять в отдельные модули. Это требование относится и к определениям объектных классов;
* Набор исходных текстов модулей предназначенных для выполнения какой-либо функции (например, модули запаковки и распаковки файлов) желательно размещать в отдельном каталоге, содержащемся в основном каталоге проекта;
* В программе предусмотреть реакцию на внешние ошибки (например, вывод сообщения об отсутствии файла или отсутствии необходимого объема оперативной памяти) и возможно неправильные действия пользователя (например, вывод предупреждающего сообщения при перезаписи файла).
* При разработке визуального интерфейса программы необходимо выполнить следующие требования:
* В главном окне программы должно присутствовать главное меню;
* Визуальные элементы управления должны быть расположены по функциональным признакам;
* Расположение визуальных элементов должно быть удобно для работы с программой;
* Надписи на всех элементах окон программы должны быть выполнены на одном языке;
* Цветовая гамма окон программы должна быть удобна для работы с программой;
* Желательно наличие справочной системы (доступной через главное меню программы).
* Необходимо учесть, что стандартизированный интерфейс пользователя повышает конкурентоспособность программного продукта и позволяет значительно сократить затраты на обучение пользователей, использующих программу.

## 2.3 Описание алгоритма работы аптеки

Декомпозиция контекстной диаграммы показывает, из каких этапов состоит процесс (рис. 1):

1. Прием заявки на заказ;

2. Оформление списка препаратов на заказ;

3. Передача списка препаратов на заказ директору;

4. Оформление заказа;

5. Отправление заказа поставщикам;

6. Прием заказа;

7. Продажа.

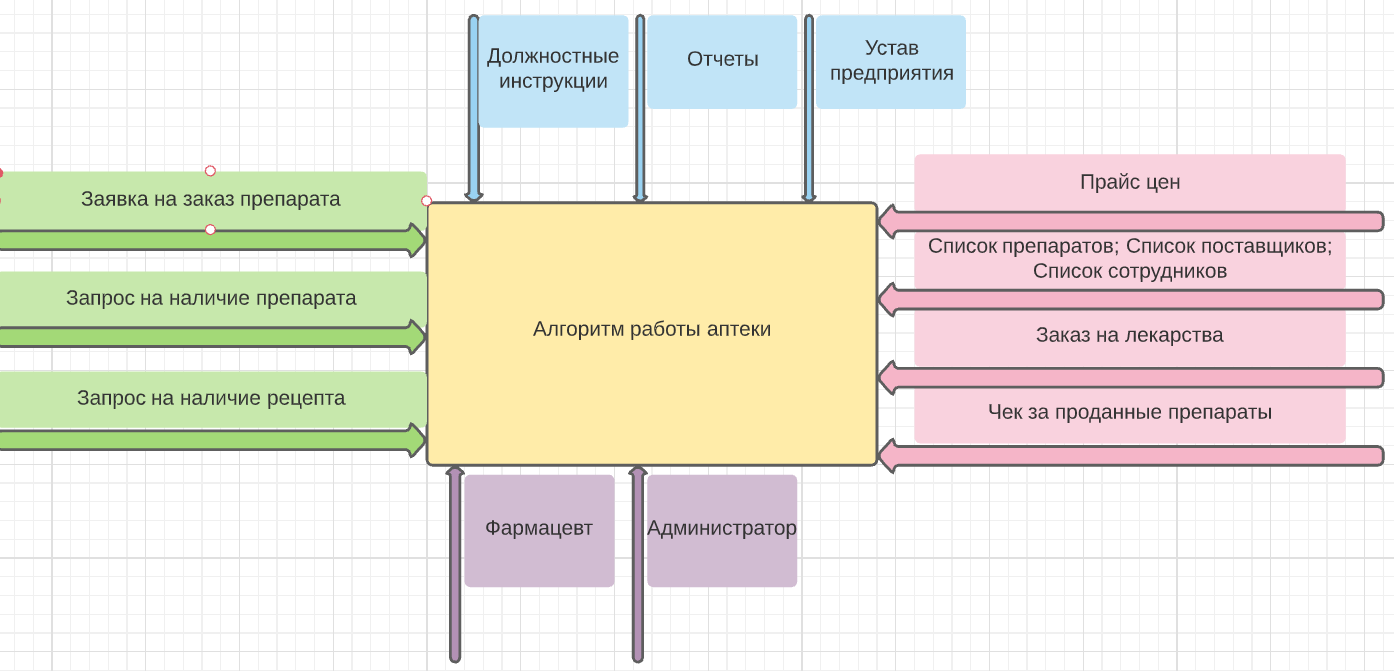


Рис. 1 Алгоритм работы аптеки

Функциональные задачи будущих пользователей

Администратор:

* создание новых пользователей;
* добавление производителей;
* приход товаров;
* регулирование БД;
* вывод отчетов;
* изменение структуры базы данных;

Фармацевт:

* заполнение документов;
* заполнение справочников;
* формирование чеков;
* ввод данных;

## 2.4 Описание входных и выходных данных

Входными данными является информация о лекарственных препаратах, или запросы от пользователя.

Входные данные:

* Заявка на заказ препарата;
* Запрос на наличие препарата;
* Запрос на срок годности;
* Запрос на наличие чека;
* Запрос на вывод отчетности;

Выходными данным является информация о лекарственных препаратах,

найденная по запросу аптекаря, и информация о реализованной продукции, указанная в отчётах

Выходные данные:

* Прайс цен;
* Список препаратов;
* Список покупателей;
* Список поставщиков;
* Заказ на лекарство;
* Чек на проданные препараты;
* Сортировка данных;
* Таблицы;
* Изображения;

Управление:

* Должностные инструкции;
* Нормативные акты;
* Устав предприятия;

Исполнительными механизмами являются:

* Фармацевт;
* Администратор;

# 3 Проектная часть

## 3.1 Краткая характеристика среды разработки Visual Studio

Microsoft Visual Studio - это программная средапоразработке приложений для ОС Windows, как консольных, так и с графическим интерфейсом.

В комплект входят следующие основные компоненты:

* Visual Basic.NET - для разработки приложений на VisualBasic;
* Visual C++ - на традиционном языке C++;
* Visual C# - на языке C# (Microsoft);
* Visual F# - на F# (Microsoft Developer Division).

Функциональная структура среды включает в себя:

* редактор исходного кода, который включает множество дополнительных функций, как автодополнение IntelliSense, рефракторинг кода и т. д.;
* отладчик кода;
* редактор форм, предназначенный для упрощённого конструирования графических интерфейсов;
* веб-редактор;
* дизайнер классов;
* дизайнерсхем баз данных.

Visual Studio также позволяет создавать и подключать сторонние дополнения (плагины) для расширения функциональности практически на каждом уровне, включая добавление поддержки систем контроля версий исходного кода (Subversion и VisualSourceSafe), добавление новых наборов инструментов (для редактирования и визуального проектирования кода на предметно-ориентированных языках программирования или инструментов для прочих аспектов процесса разработки программного обеспечения). Visual Studio внутренне не поддерживает какие-либо языки программирования, решения или инструменты; вместо этого он позволяет подключать функциональные возможности, закодированные как VSPackage. После установки функциональность доступна как услуга . IDE предоставляет три вида услуг: SVsSolution, который предоставляет возможность перечислить проекты и решения; SVsUIShell, который обеспечивает работу с окнами и функциональность пользовательского интерфейса (включая вкладки, панели инструментов и окна инструментов); и SVsShell, который занимается регистрацией VSPackages. Кроме того, IDE также отвечает за координацию и обеспечение связи между службами. Все редакторы, дизайнеры, типы проектов и другие инструменты реализованы как VSPackages. Visual Studio использует COM для доступа к пакетам VSPackages. Visual Studio SDK также включает Managed Package Framework (MPF), который представляет собой набор управляемых оболочек вокруг COM-интерфейсов, которые позволяют писать пакеты на любом языке, совместимом с CLI [15]. Однако MPF не обеспечивает всех функций, предоставляемых COM-интерфейсами Visual Studio. Затем эти службы можно использовать для создания других пакетов, которые добавляют функциональность в интегрированную среду разработки Visual Studio. Поддержка языков программирования добавляется с помощью специального пакета VSPackage, который называется языковой службой. Языковая служба определяет различные интерфейсы, которые реализация VSPackage может реализовать для добавления поддержки различных функций. Функциональные возможности, которые могут быть добавлены таким образом, включают раскраску синтаксиса, завершение операторов, сопоставление скобок, всплывающие подсказки с информацией о параметрах, списки элементов и маркеры ошибок для фоновой компиляции. Если интерфейс будет реализован, функционал будет доступен для языка [23]. Языковые услуги реализованы для каждого языка. Реализации могут повторно использовать код анализатора или компилятора языка. Языковые службы могут быть реализованы либо в машинном коде, либо в управляемом коде. Для машинного кода можно использовать либо собственные интерфейсы COM, либо Babel Framework (часть Visual Studio SDK). Для управляемого кода MPF включает оболочки для написания управляемых языковых служб. Visual Studio не включает встроенную поддержку системы управления версиями, но определяет два альтернативных способа интеграции систем управления версиями с IDE. Пакет VSPackage управления версиями может предоставлять собственный настраиваемый пользовательский интерфейс. Напротив, подключаемый модуль управления версиями, использующий MSSCCI (интерфейс управления исходным кодом Microsoft), предоставляет набор функций, которые используются для реализации различных функций управления версиями со стандартным пользовательским интерфейсом Visual Studio. MSSCCI сначала использовался для интеграции Visual SourceSafe с Visual Studio 6.0, но позже был открыт с помощью Visual Studio SDK. Visual Studio.NET 2002 использовал MSSCCI 1.1, а Visual Studio .NET 2003 использовал MSSCCI 1.2. Visual Studio 2005, 2008 и 2010 используют MSSCCI версии 1.3, в которой добавлена ​​поддержка переименования и удаления, а также асинхронного открытия [21].

## 3.2 Краткая характеристика языка SQL

SQL - это язык структурированных запросов, который является компьютерным языком для хранения, обработки и извлечения данных, хранящихся в реляционной базе данных.

SQL является стандартным языком для реляционной системы баз данных. Все системы управления реляционными базами данных (RDMS), такие как MySQL, MS Access, Oracle, Sybase, Informix, Postgres и SQL Server, используют SQL в качестве стандартного языка баз данных.

Кроме того, они используют разные диалекты, такие как:

* MS SQL Server с использованием T-SQL;
* Oracle использует PL / SQL;
* MS Access версия SQL называется JET SQL (собственный формат);

Основные категории команд языка SQL:

Определение структур базы данных (DDL)

Язык определения данных (Data Definition Language, DDL) позволяет создавать и изменять структуру объектов базы данных, например, создавать и удалять таблицы. Основными командами языка DDL являются следующие: CREATE TABLE, ALTER TABLE, DROP TABLE, CREATE INDEX, ALTER INDEX, DROP INDEX.

Манипулирование данными (DML)

Язык манипулирования данными (Data Manipulation Language, DML) используется для манипулирования информацией внутри объектов реляционной базы данных посредством трех основных команд: INSERT, UPDATE, DELETE.

Выборка данных (DQL)

Язык запросов DQL наиболее известен пользователям реляционной базы данных, несмотря на то, что он включает всего одну команду SELECT. Эта команда вместе со своими многочисленными опциями и предложениями используется для формирования запросов к реляционной базе данных.

Язык управления данными (DCL - Data Control Language)

Команды управления данными позволяют управлять доступом к информации, находящейся внутри базы данных. Как правило, они используются для создания объектов, связанных с доступом к данным, а также служат для контроля над распределением привилегий между пользователями. Команды управления данными следующие: GRANT, REVOKE.

Команды администрирования данных

С помощью команд администрирования данных пользователь осуществляет контроль за выполняемыми действиями и анализирует операции базы данных; они также могут оказаться полезными при анализе производительности системы. Не следует путать администрирование данных с администрированием базы данных, которое представляет собой общее управление базой данных и подразумевает использование команд всех уровней.

Команды управления транзакциями

Существуют следующие команды, позволяющие управлять транзакциями базы данных: COMMIT, ROLLBACK,

## 3.3 Краткая характеристика языка программирования C#

C#, C-sharp, си-шарп - язык программирования, сочетающий объектно-ориентированные и аспектно-ориентированные концепции. Разработан в 1998-2001 годах группой инженеров под руководством Андерсa Хейлсбергa в компании Microsoft как основной язык разработки приложений для платформы Microsoft .NET. Компилятор с C# входит в стандартную установку самой .NET, поэтому программы на нём можно создавать и компилировать даже без инструментальных средств вроде Visual Studio [4].

C# относится к семье языков с C-подобным синтаксисом, из них его синтаксис наиболее близок к С++ и Java. Язык имеет строгую статическую типизацию, поддерживает полиморфизм, перегрузку операторов, указатели на функции-члены классов, атрибуты, события, свойства, исключения, комментарии в формате XML. Переняв многое от своих предшественников - языков С++, Java, Delphi, Модула и Smalltalk - С#, опираясь на практику их использования, исключает некоторые модели, зарекомендовавшие себя как проблематичные при разработке программных систем: так, C# не поддерживает множественное наследование классов (в отличие от C++).

C# разрабатывался как язык программирования прикладного уровня для CLR и, как таковой, зависит, прежде всего, от возможностей самой CLR. Это касается, прежде всего, системы типов C#, которая отражает FCL. Присутствие или отсутствие тех или иных выразительных особенностей языка диктуется тем, может ли конкретная языковая особенность быть транслирована в соответствующие конструкции CLR. Так, с развитием CLR от версии 1.1 к 2.0 значительно обогатился и сам C#; подобного взаимодействия следует ожидать и в дальнейшем. (Однако эта закономерность была нарушена с выходом C# 3.0, представляющим собой расширения языка, не опирающиеся на расширения платформы .NET.) CLR предоставляет C#, как и всем другим .NET-ориентированным языкам, многие возможности, которых лишены «классические» языки программирования. Например, сборка мусора не реализована в самом C#, а производится CLR для программ, написанных на C# точно так же, как это делается для программ на VB.NET, J# и др.

Когда говорят C#, нередко имеют в виду технологии платформы .NET (Windows Forms, WPF, ASP.NET, Xamarin). И, наоборот, когда говорят .NET, нередко имеют в виду C#. Однако, хотя эти понятия связаны, отождествлять их неверно. Язык C# был создан специально для работы с фреймворком .NET, однако само понятие .NET несколько шире.

Как-то Билл Гейтс сказал, что платформа .NET - это лучшее, что создала компания Microsoft. Возможно, он был прав. Фреймворк .NET представляет мощную платформу для создания приложений. Можно выделить следующие ее основные черты:

* Поддержка нескольких языков. Основой платформы является общеязыковая среда исполнения Common Language Runtime (CLR), благодаря чему .NET поддерживает несколько языков: наряду с C# это также VB.NET, C++, F#, а также различные диалекты других языков, привязанные к .NET, например, Delphi.NET. При компиляции код на любом из этих языков компилируется в сборку на общем языке CIL (Common Intermediate Language) - своего рода ассемблер платформы .NET. Поэтому при определенных условиях мы можем сделать отдельные модули одного приложения на отдельных языках.
* Кроссплатформенность. .NET является переносимой платформой (с некоторыми ограничениями). Например, последняя версия платформы на данный момент - .NET 5 поддерживается на большинстве современных ОС Windows, MacOS, Linux. Используя различные технологии на платформе .NET, можно разрабатывать приложения на языке C# для самых разных платформ - Windows, MacOS, Linux, Android, iOS, Tizen [1].
* Мощная библиотека классов. .NET представляет единую для всех поддерживаемых языков библиотеку классов. И какое бы приложение мы не собирались писать на C# - текстовый редактор, чат или сложный веб-сайт - так или иначе мы задействуем библиотеку классов .NET.
* Разнообразие технологий. Общеязыковая среда исполнения CLR и базовая библиотека классов являются основой для целого стека технологий, которые разработчики могут задействовать при построении тех или иных приложений. Например, для работы с базами данных в этом стеке технологий предназначена технология ADO.NET и Entity Framework Core. Для построения графических приложений с богатым насыщенным интерфейсом - технология WPF и UWP, для создания более простых графических приложений - Windows Forms. Для разработки мобильных приложений - Xamarin. Для создания веб-сайтов и веб-приложений - ASP.NET и т.д.
* К этому стоит добавить активной развивающийся и набирающий популяность Blazor - фреймворк, который работает поверх .NET и который позволяет создавать веб-приложения как на стороне сервера, так и на стороне клиента. А в будущем будет поддерживать создание мобильных приложений и, возможно, десктоп-приложений.
* Производительность. Согласно ряду тестов веб-приложения на .NET 5 в ряде категорий сильно опережают веб-приложения, построенные с помощью других технологий. Приложения на .NET 5 в принципе отличаются высокой производительностью.

Также еще следует отметить такую особенность языка C# и фреймворка .NET, как автоматическая сборка мусора. А это значит, что нам в большинстве случаев не придется, в отличие от С++, заботиться об освобождении памяти. Вышеупомянутая общеязыковая среда CLR сама вызовет сборщик мусора и очистит память. Известно как минимум о трёх независимых реализациях C#, базирующихся на этой спецификации и находящихся в настоящее время на различных стадиях разработки: Mono, начата компанией Ximian, продолжена её покупателем и преемником Novell, а затем Xamarin; dotGNU и Portable.NET, разрабатываемые Free Software Foundation [3].

## 3.4 Краткая характеристика БД

SQL Server является одной из наиболее популярных систем управления базами данных (СУБД) в мире. Данная СУБД подходит для самых различных проектов: от небольших приложений до больших высоконагруженных проектов.

SQL Server был создан компанией Microsoft. Первая версия вышла в 1987 году. А текущей версией является версия 2019, которая вышла в 2019 году и которая будет использоваться в текущем руководстве.

SQL Server долгое время был исключительно системой управления базами данных для Windows, однако начиная с версии 16 эта система доступна и на Linux [9].

SQL Server характеризуется такими особенностями как:

* Производительность. SQL Server работает очень быстро.
* Надежность и безопасность. SQL Server предоставляет шифрование данных.
* Простота. С данной СУБД относительно легко работать и вести администрирование.

Центральным аспектом в MS SQL Server, как и в любой СУБД, является база данных. База данных представляет хранилище данных, организованных определенным способом. Нередко физически база данных представляет файл на жестком диске, хотя такое соответствие необязательно. Для хранения и администрирования баз данных применяются системы управления базами данных (database management system) или СУБД (DBMS). И как раз MS SQL Server является одной из такой СУБД.

Для организации баз данных MS SQL Server использует реляционную модель. Эта модель баз данных была разработана еще в 1970 году Эдгаром Коддом. А на сегодняшний день она фактически является стандартом для организации баз данных [10].

Реляционная модель предполагает хранение данных в виде таблиц, каждая из которых состоит из строк и столбцов. Каждая строка хранит отдельный объект, а в столбцах размещаются атрибуты этого объекта.

Для идентификации каждой строки в рамках таблицы применяется первичный ключ (primary key). В качестве первичного ключа может выступать один или несколько столбцов. Используя первичный ключ, мы можем ссылаться на определенную строку в таблице. Соответственно две строки не могут иметь один и тот же первичный ключ [16].

Через ключи одна таблица может быть связана с другой, то есть между двумя таблицами могут быть организованы связи. А сама таблица может быть представлена в виде отношения («relation»).

Главная страница SSMS состоит из Object Explorer, страницы запроса, панели инструментов, верхнего меню (рис. 2). В SSMS мы можем создать диаграмму БД, создать роли пользователей, триггеры, функции.

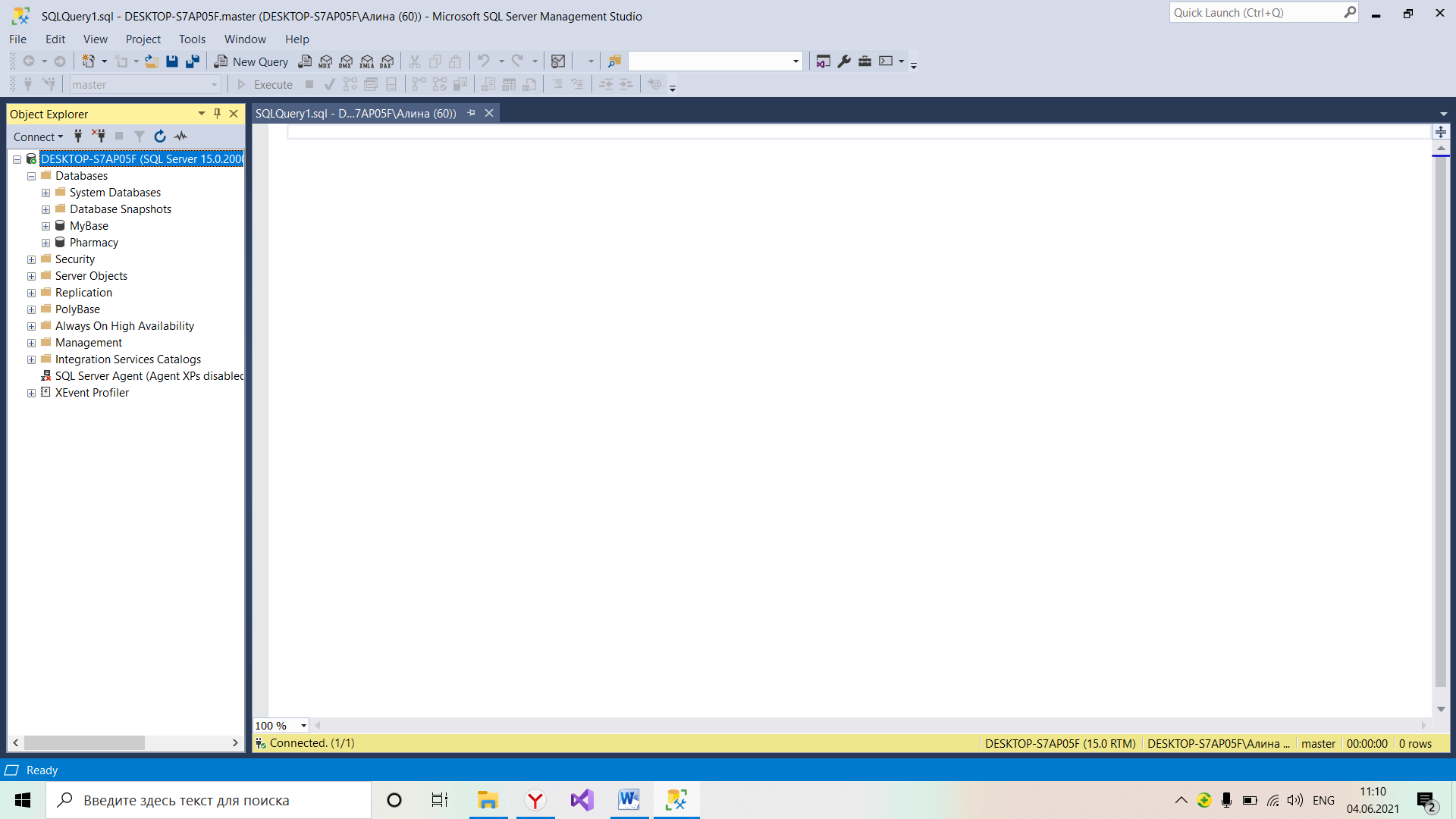
,

Рис. 2 Основная страница SSMS

Для взаимодействия с базой данных применяется язык SQL (Structured Query Language). Клиент (например, внешняя программа) отправляет запрос на языке SQL посредством специального API. СУБД должным образом интерпретирует и выполняет запрос, а затем посылает клиенту результат выполнения.

Изначально язык SQL был разработан в компании IBM для системы баз данных, которая называлась System/R. При этом сам язык назывался SEQUEL (Structured English Query Language). Хотя в итоге ни база данных, ни сам язык не были впоследствии официально опубликованы, по традиции сам термин SQL нередко произносят как «сиквел».

В 1979 году компания Relational Software Inc. разработала первую систему управления баз данных, которая называлась Oracle и которая использовала язык SQL. В связи с успехом данного продукта компания была переименована в Oracle.

Впоследствии стали появляться другие системы баз данных, которые использовали SQL. В итоге в 1989 году Американский Национальный Институт Стандартов (ANSI) кодифицировал язык и опубликовал его первый стандарт. После этого стандарт периодически обновлялся и дополнялся. Последнее его обновление состоялось в 2011 году. Но несмотря на наличие стандарта нередко производители СУБД используют свои собственные реализации языка SQL, которые немного отличаются друг от друга.

Выделяются две разновидности языка SQL: PL-SQL и T-SQL. PL-SQL используется в таких СУБД как Oracle и MySQL. T-SQL (Transact-SQL) применяется в SQL Server. Собственно поэтому в рамках текущего руководства будет рассматриваться именно T-SQL.

В зависимости от задачи, которую выполняет команда T-SQL, он может принадлежать к одному из следующих типов:

DDL (Data Definition Language / Язык определения данных). К этому типу относятся различные команды, которые создают базу данных, таблицы, индексы, хранимые процедуры и т.д. В общем определяют данные.

В частности, к этому типу мы можем отнести следующие команды:

* CREATE: создает объекты базы данных (саму базу даных, таблицы, индексы и т.д.);
* ALTER: изменяет объекты базы данных;
* DROP: удаляет объекты базы данных;
* TRUNCATE: удаляет все данные из таблиц.

DML (Data Manipulation Language / Язык манипуляции данными). К этому типу относят команды на выбору данных, их обновление, добавление, удаление - в общем все те команды, с помощью которыми мы можем управлять данными.

К этому типу относятся следующие команды:

* SELECT: извлекает данные из БД;
* UPDATE: обновляет данные;
* INSERT: добавляет новые данные;
* DELETE: удаляет данные.

DCL (Data Control Language / Язык управления доступа к данным). К этому типу относят команды, которые управляют правами по доступу к данным. В частности, это следующие команды:

* GRANT: предоставляет права для доступа к данным;
* REVOKE: отзывает права на доступ к данным;

SQL Server Management Studio (SSMS) - это интегрированная среда для управления любой инфраструктурой SQL. Используйте SSMS для доступа, настройки, администрирования и разработки всех компонентов SQL Server, Базы данных SQL Azure и Azure Synapse Analytics, а также управления ими. Среда SSMS предоставляет единую полнофункциональную служебную программу, которая сочетает в себе обширную группу графических инструментов с рядом отличных редакторов сценариев для доступа к службе SQL Server для разработчиков и администраторов баз данных всех профессиональных уровней.

SQL Server Management Studio позволяет управлять объектами служб Службы Analysis Services, например выполнять их резервное копирование и обработку.

Среда Management Studio позволяет создавать проекты скриптов служб Службы Analysis Services, в которых выполняются разработка и сохранение скриптов с использованием многомерных выражений (MDX), расширений интеллектуального анализа данных (DMX) и XML для аналитики (XMLA). Проекты скриптов служб Службы Analysis Services используются для выполнения задач управления или повторного создания баз данных, кубов и других объектов в экземплярах служб Службы Analysis Services. Например, можно разработать скрипт XMLA в проекте скрипта служб Службы Analysis Services, который создает объекты непосредственно в существующем экземпляре служб Службы Analysis Services . Проекты скриптов служб Службы Analysis Services могут быть сохранены в составе решения и интегрироваться с контролем исходного кода [5].

SQL Server Management Studio позволяет использовать службу Службы Integration Services для управления пакетами и наблюдения за выполняющимися пакетами. В среде Среда Management Studio можно организовать пакеты в папки, выполнять, импортировать и экспортировать пакеты, переносить пакеты служб DTS и обновлять пакеты служб Службы Integration Services.

Блокировка разных языков установки снята. Например, немецкий SSMS можно установить на французскую версию Windows. Если язык операционной системы не соответствует языку SSMS, пользователю следует изменить язык в разделе «Сервис» - «Параметры» - «Международные настройки». В противном случае SSMS будет отображать интерфейс пользователя на английском языке.

## 3.5 Windows Forms

Windows Forms (WinForms) - это бесплатная библиотека классов с графическим интерфейсом (GUI) с открытым исходным кодом, входящая в состав Microsoft .NET Framework или Mono Framework, предоставляющая платформу для написания многофункциональных клиентских приложений для настольных, портативных и планшетных ПК. Хотя он рассматривается как замена более ранней и более сложной библиотеки классов Microsoft Foundation на основе C ++, он не предлагает сопоставимой парадигмы и действует только как платформа для уровня пользовательского интерфейса в многоуровневом решении. На мероприятии Microsoft Connect 4 декабря 2018 года Microsoft объявила о выпуске Windows Forms в качестве проекта с открытым исходным кодом на GitHub . Он выпущен под лицензией MIT . В этом выпуске Windows Forms стала доступна для проектов, ориентированных на платформу .NET Core. Однако фреймворк по-прежнему доступен только на платформе Windows, а неполная реализация Windows Forms в Mono остается единственной кроссплатформенной реализацией [20].

Приложение Windows Forms является применением управляемых событий при поддержке компании Microsoft .NET Framework. В отличие от пакетной программы, она проводит большую часть своего времени, просто ожидая, пока пользователь что-то сделает, например, заполнит текстовое поле или нажмет кнопку. Windows Forms обеспечивает доступ к собственным общим элементам управления пользовательского интерфейса Windows, заключая существующий Windows API в управляемый код. С помощью Windows Forms .NET Framework предоставляет более полную абстракцию над Win32 API, чем Visual Basic или MFC. Windows Forms похожа на библиотеку Microsoft Foundation Class (MFC) при разработке клиентских приложений. Он предоставляет оболочку, состоящую из набора классов C ++ для разработки приложений Windows. Однако он не предоставляет платформу приложений по умолчанию, такую ​​как MFC. Каждый элемент управления в приложении Windows Forms - это конкретный экземпляр класса [2].

Все визуальные элементы в библиотеке классов Windows Forms являются производными от класса Control. Это обеспечивает минимальную функциональность элемента пользовательского интерфейса, такую ​​как расположение, размер, цвет, шрифт, текст, а также общие события, такие как щелчок и перетаскивание. Класс Control также имеет поддержку закрепления, позволяющую элементу управления переупорядочивать свое положение под своим родительским элементом. Поддержка Microsoft Active Accessibility в классе Control также помогает ослабленным пользователям лучше использовать Windows Forms. Помимо предоставления доступа к собственным элементам управления Windows, таким как кнопка, текстовое поле, флажок и список, Windows Forms добавила свои собственные элементы управления для размещения ActiveX, компоновки, проверки и привязки данных. Эти элементы управления отображаются с использованием GDI + [14].

Для будущего развития Microsoft заменила Windows Forms с помощью записи графического интерфейса на основе XAML с использованием таких платформ, как WPF и UWP. Однако размещение компонентов графического интерфейса с помощью перетаскивания аналогично Windows Forms по-прежнему обеспечивается в XAML путем замены корневого элемента XAML страницы/окна на элемент управления пользовательского интерфейса «Canvas». При внесении этого изменения пользователь может создать окно аналогично Windows Forms, напрямую перетаскивая компоненты с помощью графического интерфейса Visual Studio. В то время как XAML обеспечивает обратную совместимость размещения перетаскиванием через элемент управления Canvas, элементы управления XAML похожи только на элементы управления Windows Forms и не являются взаимно однозначно обратно совместимыми. Они выполняют аналогичные функции и имеют похожий внешний вид, но свойства и методы достаточно разные, чтобы потребовать переназначения одного API на другой [12].

# 4 Экономическая часть

## 4.1 Расчет расходов на программу

Для расчета расходов на заработанную платы необходимо умножить среднюю часовую ставку программиста на трудоемкость работы, чел/час по каждому из этапов разработки системы.

Средняя часовая ставка взята по формуле:

(1)

где - средняя часовая ставка программиста.

- средняя месячная ставка начинающего программиста (60000 тенге).

= 60000 /168 = 360 тенге.

= 500 тенге - среднечасовая ставка руководителя ВКР

= 500 тенге - среднечасовая ставка консультанта по организационно-экономическому разделу. Исходя из полученных данных, можно вычислить заработную плату по всем этапам разработки, результат в таблице 2.

Таблица 2

Расчет основной и дополнительной заработной платы

|  |  |  |  |
| --- | --- | --- | --- |
| п/п | Содержание работы | Трудоемкость работы чел/час | Основная заработная плата (тенге) |
| 1. | Анализ предметной области | 8 | 2880 |
| 2. | Постановка задачи | 8 | 2880 |
| 3. | Разработка технического задания | 12 | 4320 |
| 4. | Проектирование конфигурации | 80 | 28800 |
| 5. | Разработка интерфейса программы | 51 | 18360 |
|  |  | продолжение таблицы 2 | |
| 6. | Разработка модулей | 100 | 36000 |
| 7. | Тестирование системы | 80 | 28800 |
| 8. | Документирование | 11 | 3960 |
| 9. | Руководство ВКР | 20 | 10000 |
| 10. | ИТОГО | 370 | 136000 |

Дополнительную заработную плату разработчиков определяют в процентах от итоговой суммы основной заработной платы (15 %).

Зп доп.= 136000 ∙ 0,15 =20400 (тг.).

Таблица 3

Элементы затрат

|  |  |  |
| --- | --- | --- |
| п/п | Элементы затрат | Сумма, тг. |
| 1. | Затраты на основную заработную плату | 136600 |
| 2. | Затраты на дополнительную заработную плату | 20400 |
| 3. | Отчисление на социальные нужды | 5495 |
| ИТОГО | | 161895 |

Оценка экономической эффективности

Без использования приложения разработанного в рамках выпускной квалификационной работы происходит нерациональная трата рабочего времени сотрудников аптеки, деятельность которых связана с учетом большого количества операций, множество лекарственных средств и покупателей серьезно замедляют работу кассиров. Сложность поиска нужного препарата в каталоге занимает длительное время. Используя разработанное приложение, кассир рациональнее использует трудовые ресурсы.

Разрабатываемое программное обеспечение предназначено для облегчения работы сотрудника аптеки, поэтому дополнительных рабочих мест, а соответственно и увеличения заработной платы не планируется. Соответственно, дополнительных затрат на оплаты труда не будет.

Необходимо учесть значение средней стоимости часа работника выполняющего операции.

Для нахождения средней стоимости часа Зч, необходимо разделить среднюю заработанную плату кассира за смену (5000 тг) на количество рабочих часов (12 часов).

Зч = 5000 /12 = 416.6 тг.

При использовании ручного способа на работу с документами, составление отчетов, различных списков тратится tп1 = 4ч, а с помощью программы работа с документами должна сократиться примерно до tп2 = 1,5ч.

Экономия времени составит:

tп1 – tп2 = 4ч – 1,5ч = 2,5 часа в смену для одного кассира. За месяц экономия времени составит: Tpм1 = 30 ∙ 2,5 часа = 75 ч. С учетом, что средняя заработная плата кассира, который будет пользоваться разработанным ПО, составляет 416.6 тг/час, аптека работает без выходных, поэтому экономия на заработной плате за 1год составит:

12 ∙ 30 ∙ 2,5ч ∙ 416.6 тг/час = 374940 тг.

В результате экономия составляет 374940 рублей. Срок окупаемости рассчитывается по формуле:

(2)

где Ток - срок окупаемости;

Зр – затраты на ручную обработку информации, руб.;

За – затраты на автоматизированную обработку информации, руб.;

К – полная стоимость владения системой, руб.

В данном случае Ток равен:

Ток=224964/(374940) = 0,6 года

Срок окупаемости = 6 месяцев.

Затем рассчитаем фактический коэффициент экономической эффективности разработки (Еф) и сопоставим его с нормативным значением коэффициента эффективности капитальных вложений Ен = 0,33:

Еф=1/Tок = 1/0,6 = 1,6.

Фактический коэффициент экономической эффективности разработки получился больше, чем нормативный, поэтому разработка и внедрение разрабатываемого продукта является эффективной.

Таким образом, в ходе проделанной работы найдены все необходимые данные, доказывающие целесообразность и эффективность данной разработки. Приведем эти данные в сводной таблице 4.

Таблица 4

Результаты экономического обоснования работы

|  |  |
| --- | --- |
| Характеристики проекта | Значение |
| Затраты на разработку и внедрение программного продукта | 161895 |
| Экономический эффект, тг. | 374940 |
| Коэффициент экономической эффективности | 1,6 |
| Срок окупаемости, лет | 0,6 |

# 5 Эксплуатация

## 5.1 Требования к аппаратному обеспечению

Минимальная комплектация:

* операционная система: Windows 7/8/10;
* тактовая частота процессора: не менее 2 Ghz;
* оперативная память (ОЗУ): не менее 2 Gb;
* жёсткий диск: не менее 10 Gb свободного дискового пространства;
* экран: не менее 17”, разрешение не менее 1024×768.

Рекомендуемая комплектация:

* операционная система: Windows 7/8/10;
* тактовая частота процессора: не менее 2 GHz. Dual Core;
* оперативная память (ОЗУ): не менее 4 Gb;
* жёсткий диск: не менее 15 Gb свободного дискового пространства;
* экран: не менее 17”, разрешение не менее 1280×1024.

Оптимальная комплектация (рабочие места делопроизводителей и ответственных руководителей):

* операционная система: Windows 7/8/10 64 bit;
* тактовая частота процессора: не менее 2 GHz. Dual Core;
* оперативная память (ОЗУ): не менее 8 Gb. (ОС 64 bit);
* жёсткий диск: не менее 25 Gb свободного дискового пространства;
* экран: не менее 17”, разрешение не менее 1920×1080.

**5.2 Роли пользователей**

Роли - это общие объекты конфигурации. Они предназначены для реализации ограничения прав доступа в прикладных решениях. Роль в конфигурации может соответствовать должностям или видам деятельности различных групп пользователей, для работы которых предназначена данная конфигурация.

В конфигурации разработаны следующие роли:

1. «Администратор»;

2. «Пользователь».

«Администратор». Роль разработана для пользователя «Администратор». Права администратора представлены в диаграмме вариантов использования (рис. 3)

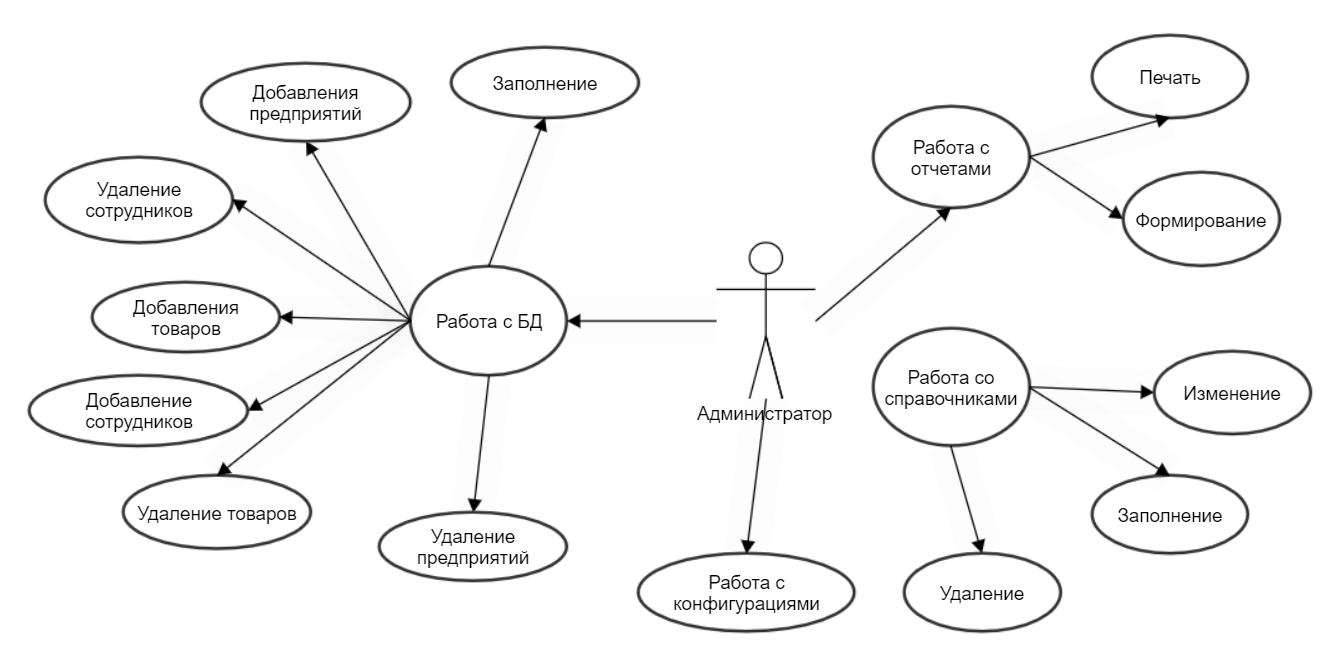


Рис. 3 Диаграмма вариантов использования администратора

«Пользователь». Роль разработана для пользователя «Фармацевт». Права пользователя представлены в диаграмме вариантов использования (рис. 4).

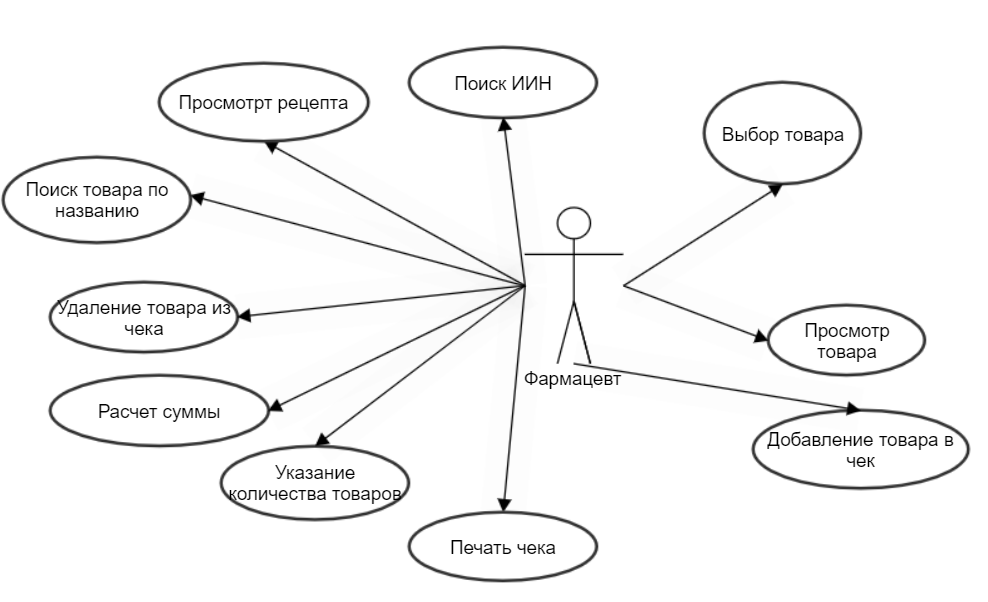


Рис. 4 Диаграмма вариантов использования фармацевта

## 5.3 Инструкция пользователю

При запуске программы откроется окно (рис. 5):

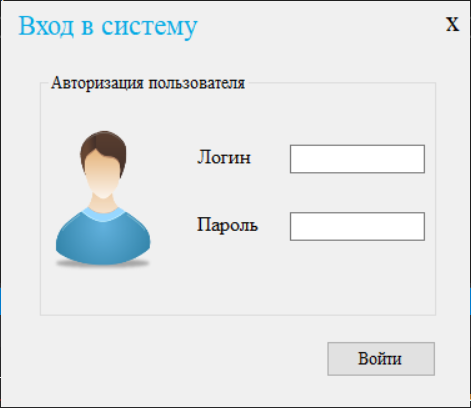


Рис. 5 Первая форма

Введите данные, по умолчанию в БД занесены для двух сотрудников (администратора и фармацевта), нажмите «Войти».

Инструкция по пользованию фармацевту:

Лекарства выведены в таблице в таком виде (рис. 6):



Рис. 6 Таблица с продуктами

Поиск товаров происходит по названию, для этого необходимо ввести название товара в рамку (рис. 7) и нажать кнопку.



Рис. 7 Поиск продуктов

Для вывода информации о товаре необходимо нажать по товару левой кнопкой мыши, после чего появиться панель (рис. 8). Здесь можно ввести желаемое количество товара и добавить его к покупке.



Рис. 8 Информация о продукте

Здесь отображаются все товары, которые собираются купить (рисунок 9). Товар также можно добавить двойным нажатием левой кнопки мыши, после чего он автоматически в единичном количестве попадет в таблицу с покупками. После нажатия кнопки оформить формируется чек, и таблица с товарами очищается, это действие отменить нельзя. Кнопка «Удалить» удаляет выделенный товар в таблице.

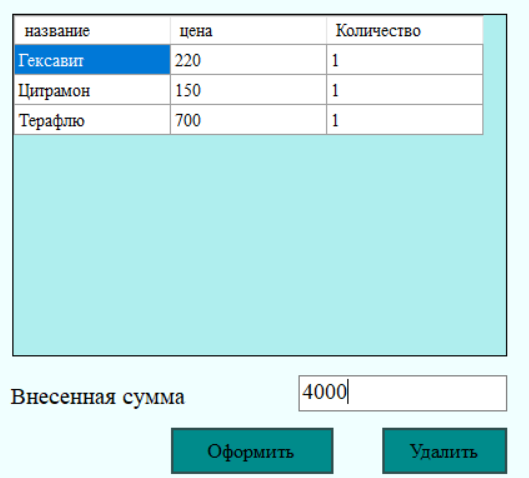


Рис. 9 Таблица товаров покупки

Для просмотра рецепта необходимо открыть верхнее меню и выбрать пункт «Посмотреть рецепт». После щелчка левой кнопкой мыши по выбранному пункту появится окно, на котором будет расположена поле для ввода данных и пустое изображение. По окончанию ввода данных клиента, а именно ИИН (Идентификационный номер налогоплательщика), в случае наличия рецепта на месте пустого изображения высветится электронный вариант рецепта, в противном случае отобразится уведомление об отсутствие рецепта на препараты.

Инструкция администратору

Основная страница будет состоять из трех вкладок (рисунок 10), для перемещения по вкладкам необходимо кликнуть левой кнопкой мыши по ним. Каждая вкладка оснащена своим функционалом:

1. Добавить лекарства в БД,

2. Добавить сотрудника,

3. Добавить производителя.

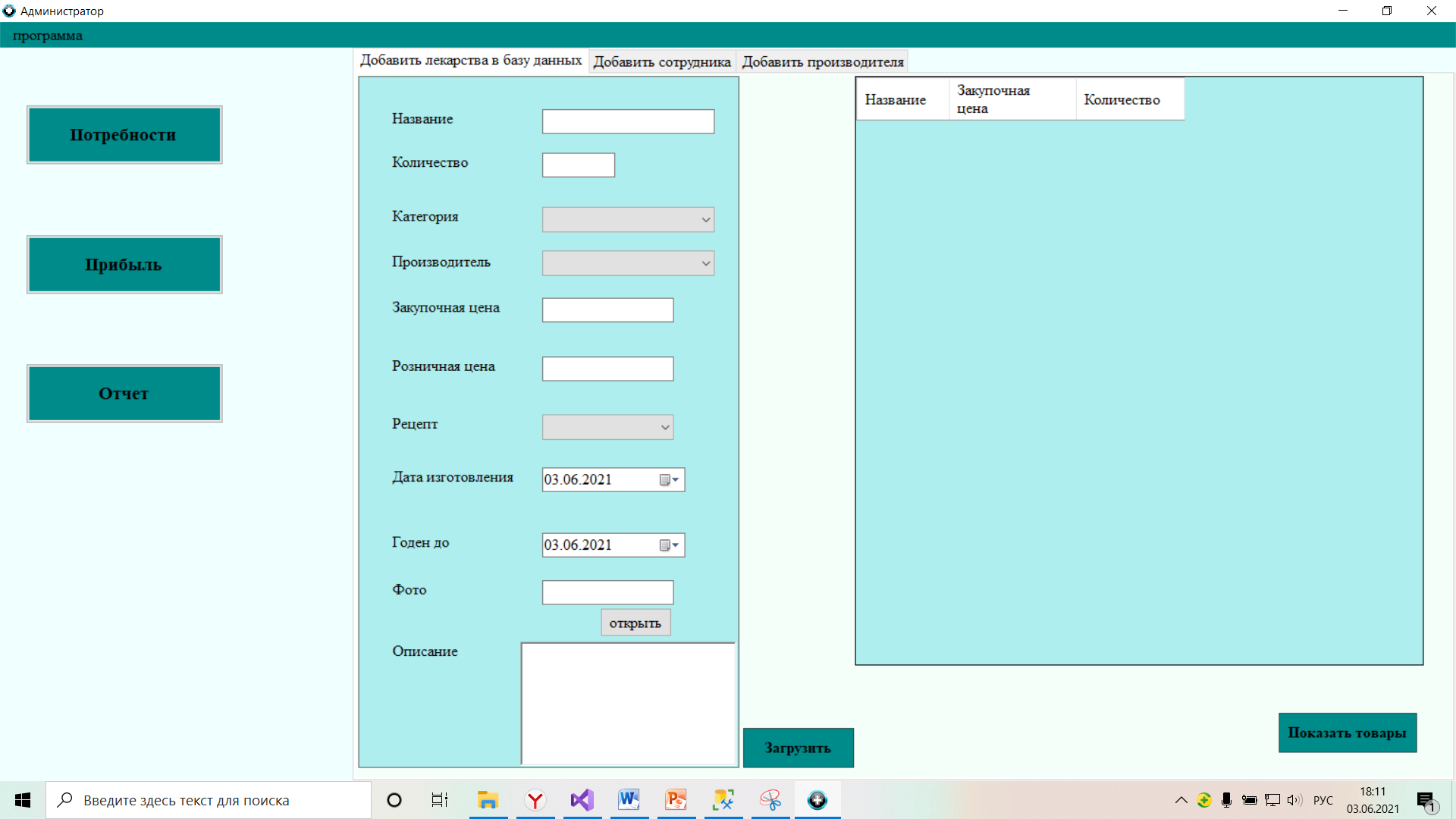


Рис. 10 Основная страница администратора

Кнопки, расположенные справа вызываются одним нажатием по ним левой кнопкой мыши. Кнопка «Потребности» - проводит анализ товаров и выводит те чье количество меньше пяти. Кнопка «Прибыль» производит расчет суммы всех купленных товаров за определенный период: день, неделя, месяц, год. Можно также произвести сортировку, посмотрев самый продаваемый товар в аптеки.

Программа оснащена дополнительными возможностями, такими как просмотр товаров, просмотр сотрудников, изменение данных о сотруднике, удалить сотрудника, удалить производителя.

Примечание: если вы выбрали сотрудника и нажали кнопку «изменить данные сотрудника», то обязательно нужно загрузить сотрудника обратно, иначе все данные будут утеряны.

Для вывода на печать необходимо нажать на кнопку «Печать», после чего откроется окно, в котором нужно будет выбрать параметры печати, количество страниц, формат и цвет шрифта.

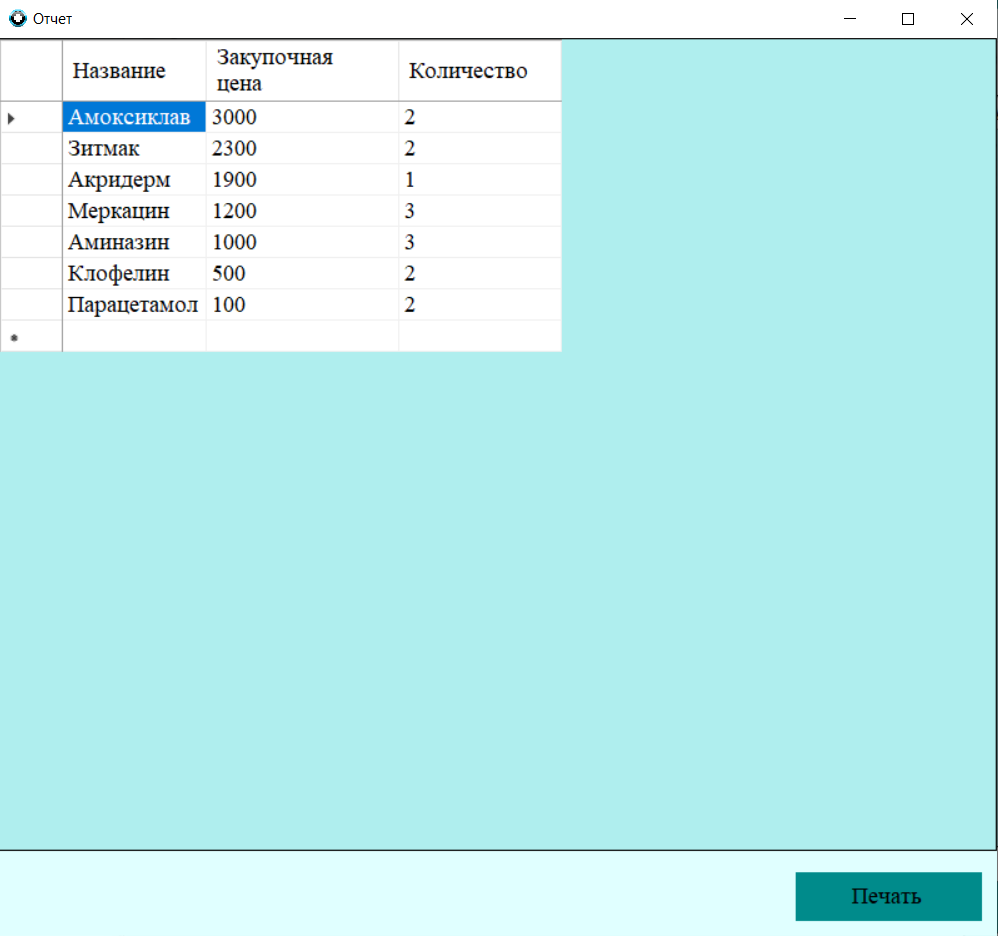


Рис. 11 Отчет

Для увеличения количества товаров в БД необходимо в верхнем меню выбрать пункт «Приход товаров», после чего откроется окно, на котором будут отображены таблица со всеми товарами и кнопка «Печать». Для изменения количества определенного товара в БД необходимо кликнуть по товару левой кнопкой мыши. Затем должно появиться изображения продукта, в поле для ввода необходимо будет ввести количество товара и нажать кнопку «Добавить», после успешного добавление отобразится уведомление об успешной операции (рис. 12).

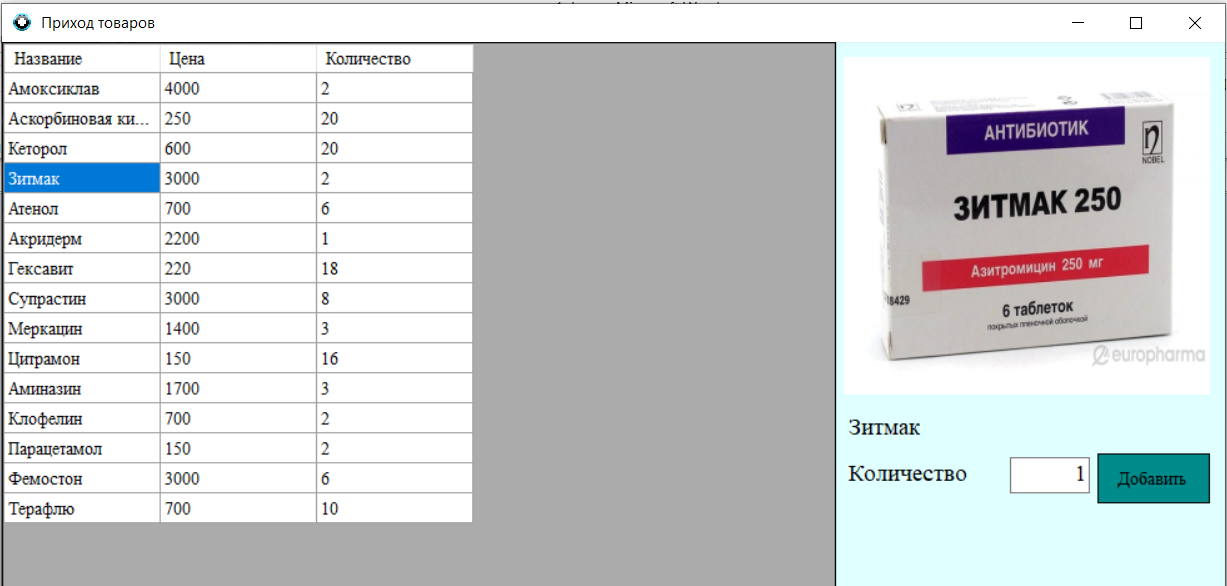


Рис. 12 Приход товаров

**5.4 Инструкция программисту**

Работа программы начинается с запуска .exe файла. Логин и Пароль для входа будут прикреплены вместе с программой в текстовом файле. Готовая БД также будет располагаться со всей программой, необходимо будет лишь ее восстановить с помощью back файла.

Весь код поделен по папкам, подключение к БД было осуществлено с помощью ADO.Net (рис. 13).

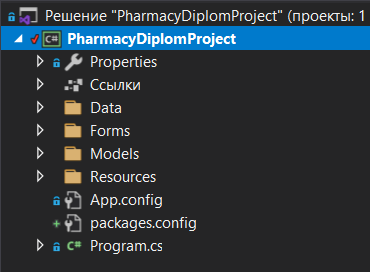


Рис. 13 Состав проекта

БД состоит из семи таблиц (рис. 14);

Таблица Products хранит информацию о товарах, которые есть в аптеки. Основные поля: Название, Закупочная цена, Розничная цена, Категория, Производитель, Количество, Фото, Рецепт, Дата производства.

Таблица Employees хранит информацию о всех сотрудниках аптеки. Основные поля: ФИО, Должность, Зарплата, Телефон, Логин и Пароль для входа, Адрес, Дата трудоустройства, Фото.

Таблица Manufacturer хранит информацию о производителях. Основные поля: Название, Страна, Город, Адрес, Телефон.

Таблица Receipt хранит информацию о чеке.

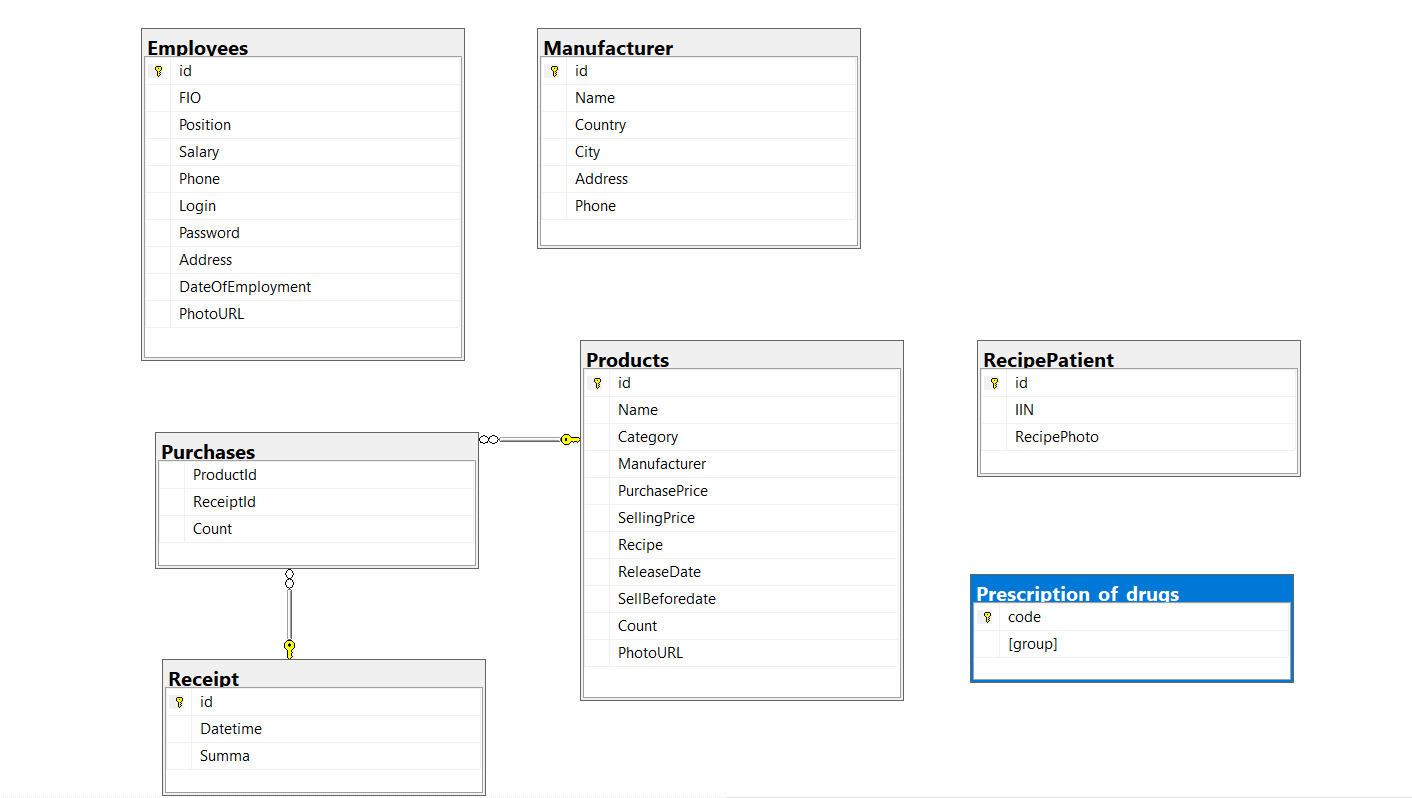


Рис. 14 БД аптеки

# 6 Мероприятия по технике безопасности и охране окружающей среды

## 6.1 Общие принципы и законодательство по охране труда

Закон РК об охране труда в Республики Казахстан. Настоящий закон направлен на обеспечение права работников на охрану труда, устанавливает основные принципы национальной политики в этой области в целях предупреждения несчастных случаев и повреждения здоровья на производстве, сведения к минимуму опасных и вредных производственных факторов и распространяется на все виды хозяйственной деятельности и предприятий независимо от форм собственности.

Охрана труда представляет собой действующую на основании соответствующих законодательных и иных нормативных актов систему социально - экономических, организационных, технических, гигиенических и лечебно - профилактических мероприятий и средств, обеспечивающих безопасность, сохранения здоровья и работоспособности человека в процессе труда.

Право на охрану труда - имеют все работники, состоящие в трудовых отношениях с предприятиями, учреждениями, кооперативами, фермерскими (крестьянскими) хозяйствами и другими организациями (в дальнейшем - предприятия) различных форм собственности и хозяйствования, в том числе: с отдельными нанимателями; члены кооперативов; студенты высших учебных заведений и учащиеся средних специальных учебных заведений и общеобразовательных школ, проходящие производственную практику и производственное обучение; военнослужащие, привлеченные к работам, не связанные с несением воинской службы; лица отбывающие наказание по приговору суда, в период их работы на предприятиях, определяемых органами, ведущими исполнением приговоров, а также участники любых других видов трудовой деятельности, организуемой в интересах общества и государства.

Национальная политика в области охраны труда предусматривает единство действий органов государственной власти и управления всех уровней при участии профсоюзов и работодателей и основывается на следующих принципах:

1. приоритета жизни и здоровья работника по отношению к результатам производственной деятельности предприятия;
2. полной ответственности собственника либо уполномоченного им представителя (в дальнейшем - работодатель);
3. комплексного решения задач охраны труда на базе государственных программ по этим вопросам и координации деятельности в области охраны труда с другими направлениями экономической и социальной политики;
4. установление единых требований в области охраны труда для всех предприятий, независимо от форм собственности и хозяйствования;
5. осуществления государственного надзора и контроля за повсеместным выполнением требований охраны труда и техники безопасности на предприятиях;
6. широкого использования достижения науки, техники и передового национального и зарубежного опыта по охране труда;
7. стимулирования разработки и внедрения безопасной техники, технологии и средств защиты работающих, научно - исследовательской работы по охране труда;
8. участия государства в финансировании охраны труда;
9. экономической заинтересованности предприятий в обеспечении здоровых и безопасных условий труда, а работников - в соблюдении правил и норм охраны труда и техники безопасности.

## 6.2 Техника безопасности при работе с персональным компьютером

При выполнении работ нужно соблюдать следующие правила:

1. запрещается класть на корпус и дисплей компьютера посторонние предметы, прикасаться к элементам аппаратуры мокрыми руками, производить чистку корпуса оборудования, находящегося под напряжением, располагать технику близко к жилищно-коммунальным инженерным системам;
2. в случае обнаружения неисправности компьютера немедленно прекратить работу и сообщить об этом непосредственному руководителю;
3. эксплуатировать компьютер только с соблюдением инструкции, установленной производителем;
4. избегать частого и необоснованного включения и выключения компьютера во время работы.

Правильное расположение за компьютером

Требования к расположению работника за компьютером нацелены на обеспечение его комфорта в течение всей рабочей смены и отсутствие негативных следствий длительной работы. Они действуют для любых рабочих мест, будь то бухгалтерия, обучающий класс или кабинет информатики. Они включают следующие правила:

1. полная опора ступнями на пол при посадке;
2. использование компьютерной мебели, отвечающей нормам СанПиН 2.2.2/2.4.1340-03;
3. отказ от скрещивания конечностей, способного затруднить кровообращение;
4. соблюдение расстояние до монитора компьютера не меньше 45 сантиметров;
5. правильная установка освещения, которое не должно светить в глаза и оставлять блики на рабочем мониторе.

Для облегчения длительной работы эксперты рекомендуют использовать вспомогательное оборудование, например, особые очки, антибликовые мониторы, коврики для мыши и проч. Нарушения техники безопасности при работе на компьютере способны вызвать стойкие расстройства здоровья, которые потом будет трудно ликвидировать.

Режим труда и отдыха

Длительность непрерывного труда и отдыха для сотрудников, постоянно работающих за персональной офисной техникой, регламентирована на законодательном уровне. Соответствующие нормы приводятся в СанПиН 2.2.2/2.4.1340-03, который устанавливает обязательную продолжительность перерывов в такой работе. Они зависят от длины рабочей смены сотрудника и определяются так: при 8-часовой рабочей смене в зависимости от тяжести и напряженности труда работника общая продолжительность перерывов за рабочий день должна составлять от 50 до 90 минут; при 12-часовой рабочей смене - от 80 до 140 минут.

## 6.3 Противопожарная безопасность при работе с персональным компьютером

Пожары в вычислительном центре представляют особую опасность, так как поряжены с большими материальными потерями. Характерная особенность небольшие площади помещений. Как известно, пожар может возникнуть при взаимодействии веществ, окислителя, и источников зажигания. Горючими компонентами являются: строительные материалы для акустической и эстетической отделки помещений, перегородки, полы, двери, изоляция силовых, сигнальных кабелей и т.д. Для отвода теплоты от ЭВМ действует мощная система кондиционирования. Поэтому кислород, какокислитель процессов горения, имеется в любой точке помещений ВЦ.

Особенностью компьютеров является очень высокая плотность расположения микросхем. При прохождении электрического тока по проводникам и деталям выделяется тепло, что в условиях их высокой плотности может привести к перегреву. Надежная работа отдельных элементов и микросхем в целом обеспечивается только в определенных интервалах температуры, влажности и при заданных электрических параметрах. При отклонении реальных условий эксплуатации от расчетных могут возникнуть пожароопасные ситуации.

Кабельные линии являются наиболее пожароопасным местом. Наличие горючего изоляционного материала, вероятных источников зажигания в виде электрических искр и дуг, разветвленность и недоступность делают кабельные линии местом наиболее вероятного возникновения и развития пожара. Для понижения воспламеняемости и способности распространять пламя кабели покрывают огнезащитными покрытиями.

Пожар - это неконтролируемое горение вне специального очага, который наносит огромный ущерб. Главная причина пожара неэлектрического характера является неосторожное обращение с огнем, а также взрывы газо-воздушных и паро-воздушных смесей. Электрическое горение - это замыкание, перегрузка электрического тока на электрическом оборудовании, грозовая молния.

## 6.4 Мероприятия, проводимые для охраны окружающей среды

Охрана окружающей среды - комплекс научно-технических, производственно-хозяйственных и административно-правовых мероприятии, направленных на сохранение и надлежащее использование земли и ее недр, водных ресурсов, растительного и животного мира, обеспечение чистоты воздуха и воды, воспроизводства природных богатств, гармоничные взаимоотношения между обществом и природой. Не сложно представить себе процветающую экономику, при которой люди из года в год становятся все более больными в результате неправильного подхода к своему здоровью и загрязненной окружающей среды. По мере того, как мы строим свое общество, необходимо применять нарастающие усилия в том, чтобы наши граждане были здоровыми на протяжении всей своей жизни, и их окружала здоровая природная среда. Источником загрязнения атмосферного воздуха называется технологический агрегат, выделяющий в процессе эксплуатации вредные вещества.

Количество выделенных веществ зависит от типа и мощности производства, его технической оснащенности и определяется путем инструментальных замеров или расчетов с использованием специальных отраслевых методик. Основными мероприятиями по снижению выбросов вредных веществ в, атмосферу являются совершенствование технологических процессов, включая снижение производственных выбросов; строительство новых и повышение эффективности существующих очистных устройств; ликвидация источников загрязнения, перепрофилирование производства. Применительно к предприятиям машиностроительного производства наиболее значимым представляется газа - и пылеулавливание вентиляционных выбросов, особенно при открытой разливке металлов.

Очистку и обезвреживание газовых составляющих выбросов промышленных производств осуществляют методами, выбор которых определяется составом, концентрации загрязняющих веществ, типа производства, условиями выброса.

Очистку вентиляционных выбросов от механических примесей осуществляют аппаратами, мокрого и сухого пылеулавливания, волокнистыми фильтрами и электрофильтрами.

В качестве фильтров используют различные фильтрующие, тонкие грубо волокнистые материалы. Кроме того, на предприятиях машиностроения широкое применение получили электрофильтры, которые в зависимости от способа удаления осажденных на электродах частиц подразделяются на сухие и мокрые.

Защита вод питьевого назначения также имеет глобальное значение.

Степень очистки сточных вод устанавливается в зависимости от местных условий с учетом возможного использования очищенных сточных вод - для промышленных и сельскохозяйственных нужд. На предприятиях металлургической и машиностроительной промышленности очистка сточных вод осуществляется, как правило, в отстойниках, шлаконакопителях, нефте-маслоловушках с использованием ряде случаев коагулянтов. Полученный шлак, содержащий большое количество металлов, утилизируется и включается в состав шихты. Очищенные воды в большинстве случаев используются в системах оборотного водоснабжения. При этом вода основного источника или из других циклов водопользования идет на компенсацию потерь оборотной воды. Важным направлением охраны окружающей среды является охрана почв от водной и ветровой эрозии, борьба с их засолением путем введения соответствующих севооборотов, создания лесозащитных полос, закрепления и облесения оврагов и балок, использования средств мелиорации.

В целях снижения загрязнения почв различными промышленными отходами в практике охраны земельных ресурсов предусматриваются следующие мероприятия: утилизация, обезвреживание методом сжигания, захоронение на специальных полигонах, организация усовершенствованных свалок.

## 6.5 Требования к освещению помещений и рабочих мест

В помещения с ПЭВМ должно быть естественное и искусственное освещение. Естественное освещение обеспечивается через оконные проемы с коэффициентом естественного освещения КЕО не ниже 1,2% в зонах с устойчивым снежным покровом и не ниже 1,5% на остальной территории. Световой поток из оконного проема должен падать на рабочее место оператора с левой стороны. Искусственное освещение в помещениях с ПЭВМ должно осуществляться системой общего равномерного освещения. Освещенность на поверхности стола в зоне размещения документа должна быть 300-500 лк. Допускается установка светильников местного освещения для подсветки документов. Местное освещение не должно создавать бликов на поверхности экрана и увеличивать освещенность экрана более 300 лк. Прямую блескость от источников освещения следует ограничить. Яркость светящихся поверхностей (окна, светильники), находящихся в поле зрения, должна быть не более 200 кд/м 2. Яркость бликов на экране монитора не должна превышать 40 кд/м2. Показатель ослепленности для источников общего искусственного освещения в помещениях должен быть не более 20, показатель дискомфорта в административнообщественных помещениях не более 40. Соотношение яркости между рабочими поверхностями не должно превышать 3:1 - 5:1, а между рабочими поверхностями и поверхностями стен и оборудования 10:1. Для искусственного освещения помещений с ПЭВМ следует применять светильники типа ЛПО36 с зеркализованными решетками, укомплектованные высокочастотными пускорегулирующими аппаратами. Допускается применять светильники прямого света, преимущественно отраженного света типа ЛПО13, ЛПО5, ЛСО4, ЛПО34, ЛПО31 с люминисцентными лампами типа ЛБ. Допускается применение светильников местного освещения с лампами накаливания. Светильники должны располагаться в виде сплошных или прерывистых линий сбоку от рабочих мест параллельно линии зрения пользователя при разном расположении компьютеров. При периметральном расположении – линии светильников должны располагаться локализованно над рабочим столом ближе к его переднему краю, обращенному к оператору. Защитный угол светильников должен быть не менее 40 градусов. Светильники местного освещения должны иметь непросвечивающийся отражатель с защитным углом не менее 40 градусов. Для обеспечения нормативных значений освещенности в помещениях следует проводить чистку стекол оконных проемов и светильников не реже двух раз в год и проводить своевременную замену перегоревших ламп.

# 6.6 Режим труда и отдыха при работе за компьютером

Режим труда и отдыха предусматривает соблюдение определенной длительности непрерывной работы на ПК и перерывов, регламентированных с учетом продолжительности рабочей смены, видов и категории трудовой деятельности. Виды трудовой деятельности на ПК разделяются на 3 группы: группа А - работа по считыванию информации с экрана с предварительным запросом; группа Б - работа по вводу информации; группа В - творческая работа в режиме диалога с ПК. Если в течение рабочей смены пользователь выполняет разные виды работ, то его деятельность относят к той группе работ, на выполнение которой тратится не менее 50% времени рабочей смены.

Категории тяжести и напряженности работы на ПК определяются уровнем нагрузки за рабочую смену: для группы А - по суммарному числу считываемых знаков; для группы Б - по суммарному числу считываемых или вводимых знаков; для группы В - по суммарному времени непосредственной работы на ПК. В таблице приведены категории тяжести и напряженности работ в зависимости от уровня нагрузки за рабочую смену. Виды категорий представлены в таблице 5.

Таблица 5

Виды категорий трудовой деятельности с ПЭВМ

|  |  |  |  |
| --- | --- | --- | --- |
| Категория работы по тяжести и напряженности | Уровень нагрузки за рабочую смену при видах работы на ПК | | |
| Группа А  Количество знаков | Группа Б  Количество знаков | Группа В  Время работы, ч |
|  |  | Продолжение таблицы 5 | |
| 1  2  3 | До 20000  До 40000  До 60000 | До 15000  До 30000  До 40000 | До 2,0  До 4,0  До 6,0 |

Количество и длительность регламентированных перерывов, их распределение в течение рабочей смены устанавливается в зависимости от категории работ на ПК и продолжительности рабочей смены.

При 8-часовой рабочей смене и работе на ПК регламентированные перерывы следует устанавливать:

* для первой категории работ через 2 часа от начала смены и через 2 часа после обеденного перерыва продолжительностью 15 минут каждый;
* для второй категории работ - через 2 часа от начала рабочей смены и через 1,5-2,0 часа после обеденного перерыва продолжительностью 15 минут каждый или продолжительностью 10 минут через каждый час работы;
* для третьей категории работ - через 1,5- 2,0 часа от начала рабочей смены и через 1,5-2,0 часа после обеденного перерыва продолжительностью 20 минут каждый или продолжительностью 15 минут через каждый час работы.

При 12-часовой рабочей смене регламентированные перерывы должны устанавливаться в первые 8 часов работы аналогично перерывам при 8-часовой рабочей смене, а в течение последних 4 часов работы, независимо от категории и вида работ, каждый час продолжительностью 15 минут.

Продолжительность непрерывной работы с ПЭВМ без регламентированного перерыва не должна превышать 2 часа. При работе с ПЭВМ в ночную смену продолжительность регламентированных перерывов увеличивается на 60 минут независимо от категории и вида трудовой деятельности.

Эффективными являются нерегламентированные перерывы (микропаузы) длительностью 1-3 минуты. Регламентированные перерывы и микропаузы целесообразно использовать для выполнения комплекса упражнений и гимнастики для глаз, пальцев рук, а также массажа. Комплексы упражнений целесообразно менять через 2-3 недели.

Пользователям ПЭВМ, выполняющим работу с высоким уровнем напряженности, показана психологическая разгрузка во время регламентированных перерывов и в конце рабочего дня в специально оборудованных помещениях (комнатах психологической разгрузки).

# Заключение

Человек - не бессмертен. Каждый день человеческий организм встречается с кучей внешних факторов, которые угрожают его жизни: проезжающий автобус, гололед и микробы… Еще совсем недавно человек мог умереть от обычного кашля или простуды. Все это было возможно по причине не развитой медицины. И что же мы делаем, когда вдруг обнаружили першение во рту, правильно - идем в аптеку.

Аптечный бизнес процветает везде, ведь все мы можем заболеть. Но на дворе 21 век - век технологий и интернета, и этому направлению не обойтись без информационных технологий. Именно поэтому темой для моего дипломного проекта является «Автоматизация аптечного учета».

Изучив ведение бизнеса в этой сфере, мне удалось создать приложение, которое способно покрыть множество нюансов работы. Оно будет являться мощным подспорьем, в продвижении развития бизнеса и увеличение прибыли, ведь оно позволяет уменьшить время, необходимое на этапы, которые можно автоматизировать.

Приложение для автоматизации - окупается в 95% случаев. Почему в 95%? Остается небольшой шанс того, что этим приложением просто не будут пользоваться после покупки. В этом, и только в это случае, есть шанс не окупаемости данного программного продукта.

Очень многие сферы деятельности требуют комплексного подхода к вопросам автоматизации. Для меня тема «Аптечного бизнеса» оказалась самой близкой по духу. Доработав сделанное мною приложение - им можно пользоваться в промышленных масштабах - и получать от этого выгоду.

Я и дальше буду развиваться в данном направлении. Анализ проблематики, выявление не оптимальных моментов в работе, постановка проблемы и нахождение способов ее решения - показались мне очень интересными.

Разрабатывая проект, я узнала много нового для себя. Пользуясь технологией, которую я применяла во время разработки моей программы, позволит мне с большей уверенностью найти работу в данной сфере.

В результате выполнения выпускной работы, поставленные задачи и цели достигнуты:

1. выявить актуальные проблемы существующих продуктов
2. разработать автоматизированную информационную систему для учета лекарств в аптеке на C#
3. провести теоретический анализ научно-методической литературы по аптечному бизнесу
4. автоматическое формирование потребности аптеки решение проблем мошенничества среди персонала

# Список использованной литературы

1. Албахари Б. C# 7.0. Справочник. Полное описание языка C#, 2017 - 400с.
2. Горелов С.В. «Современные технологии программирования: разработка Windows-приложений на языке С#. Том 1», 2019 - 363с.
3. Джуст Виссер. «Разработка обслуживаемых программ на языке С#», 2017 - 194с.
4. Джеффри Чилберто, Гаурав Арораа, Паттерны проектирования для C# и платформы .NET Core, 2021 - 352c.
5. Жилинский А. Самоучитель Misrosoft SQL Server, 2008 - 228с.
6. Казанский А.А., Программирование на Visual C# 2-е изд., пер. и доп. Учебное пособие для СПО, 2020 - 193с.
7. Кнут Д., Искусство программирования. Том 3. Сортировка и поиск, 2018 - 834с.
8. Комлев Н. Ю., Объектно-ориентированное программирование, 2014 - 299c.
9. Костюк А. И., Беспалов Д. А., Администрирование баз данных и компьютерных сетей, 2020 - 129c.
10. Кондрашов Ю. Н. Анализ данных и машинное обучение на платформе MS SQL Server, 2019 - 304с.
11. Лисовский П. Аптека в плюсе. Прибыльный аптечный бизнес, 2018 - 246с.
12. Лозинский Ю., Б., SQL Server: Новые возможности для разработчиков, 2006 - 210с.
13. Лоскутова Е.Е., Косова И.В., Лагуткина Т.П., Дорофеева В.В., Максимкина Е.А., Теодорович А.А. Управление и экономика фармации. Том 1. Фармацевтическая деятельность. Организация и регулирование. 2008 - 400с.
14. Марк Дж. Прайс. «C# 7 и .NET Core. Кросс-платформенная разработка для профессионалов», 2018 - 640с.
15. Марк Дж. Прайс, C# 8 и .NET Core. Разработка и оптимизация, 2021 - 816c.
16. Мирошниченко Г. Реляционные базы данных: практические приемы оптимальных решений, 2005 - 400с.
17. Немченко А.С. Основы экономики и системы учета в фармации. 2008 - 480с.
18. Прайс. C# 7 и .NET Core. Кросс-платформенная разработка для профессионалов, 2017 - 464с.
19. Роберт Мартин, Мика Мартин, Принципы, паттерны и методики гибкой разработки на языке C#, 2011 – 753с.
20. Скит, C# для профессионалов. Тонкости программирования, 2014 - 602с.
21. Смит Б., Бренд-терапия. 15 методов для создания стратегии бренда в индустрии фармацевтики и медицинских технологий, 2018 - 210с.
22. Стивен Клири, Конкурентность в C#. Асинхронное, параллельное и многопоточное программирование, 2019 - 304c.
23. Уотсон Б., Высокопроизводительный код на платформе .NET, 2018 - 416с.

# ПРИЛОЖЕНИЕ