Содержание

1. Введение
2. Основная часть
   1. Характеристика технических средств для решения задачи
   2. Характеристика программных средств для решения задачи
3. Проектная часть
   1. Описание предметной области/Постановка задачи

3.1 Техническое задание

3.2 Разработка структуры приложения

3.3 Разработка концептуальной модели базы данных

3.4 Дизайн приложения

3.5 Разработка основных и графических модулей приложения

3.6 Основные алгоритмы приложения

3.7 Безопасность данных

3.8 Тестирование и анализ результатов

3.9 Руководство пользователя

4. Охрана труда

4.1 Заключение

4.2 Список использованных источников

Введение

В какой период развития человечества и общества в целом берут свое начало информационные технологии? Если говорить без точной даты, то основопалагающим будет цифра «20», означающая двадцатый век нашей эры. Более точной датой начала информационых технологий является период с 1900 по 1950, характерными чертами которого будут простые механические устройства для облегчения бумажной работы и математических расчетов. Данный период является оснопологающим в истории развития информационных технологий в виду того, что в данный период были выдвинуты основные теории об архитектуре современного компьютера, принципов его работы и возможности дальнейшего развития.

Одной из теорий, которая являлась главной при развитии информационных технологий и электронно-вычислительных машин (далее ЭВМ) – теория «Машина Тьюринга», в концепции которой описывается идеализированная модель современного центрального процессора, который использует «бесконечную ленту» (последовательную память) для хранения промежуточных и итоговых значений. Дополнительной частью развития данной области является создание дешифровальной машины «Bombe», реализация которой также является большим толчком в развитии науки и технологий.

Теории и идеи, которые были выдвинуты в данный период времени, долгое время использовались как ориентир для различных инженеров, математиков, интузиастов и ученых, вовлеченные в создание некоторого устройства, способное облегчить математические расчеты в разных областях и решить глобальные математические задачи. В ходе улучшения технологий, способные уменьшить общий размер ЭВМ и повысить их общую эффективность – специалисты начали задаваться вопросом, основной идеей которого был переход от одноразовых носителей информации (перфокарт) и создание программы, способная автоматизировать программирование ЭВМ. В результате развития нескольких областей науки (физика, математика, химия) – человечество смогло решить проблему, связанную с программой для облегчения программирования. Данной программой был «Shortcode», предложенный Джоном Макколи и разработанный Уильямом Шмиттом в 1949-1950 г. Дополнительно, через некоторое время, человечество также смогло решить проблему хранения данных и самих программ, разработав кассеты с магнитной лентов и в дальйшем – дискеты, жесткие диски.

Данные технологии позволили:

* хранить определенную информацию на физических носителях;
* программировать ЭВМ, используя человеческие слова, за счет разработанных компиляторов и операционных систем;
* автоматизировать и отцифровать многие бизнес-процессы компаний;
* облегчить доступ к информации, за счет концепции всемироной паутины, разработанного сетевого протокола для межсетевого взаимодействия и веб-страниц;

Множественная и упорная работа различных инженеров, ученых, математиков и интузиастов – позволили перейти всему миру на новый этап развития, на этап множественной автоматизации и упрощения работы.

Тема дипломного проекта – «Разработка приложения для учета копьютерного оборудования ГБПОУ «НАМТ»

Задачами дипломного проекта является:

* Изучить способы организации и учета оборудования внутри учебного учреждения.
* Выявить недочеты используемого способа учета.
* Спроектировать и разработать базу данных приложения, на основе имеющихся сущностей в учебном учреждении
* Спроектировать дизайн пользовательского интерфейса приложения
* Разработать пользовательский интерфейс
* Спроектировать и разработать основные алгоритмы приложения
* Осуществить функциональное и нагрузочное тестирование приложения

Дипломный проект состоит из трех частей. В первом разделе дипломного проекта дано описание

1 Характеристика объекта управления

* 1. Общая характеристика предприятия

ГБПОУ «Нижегородский автомеханический техникум» — одно из старейших учебных заведений Нижегородской области, основанное в 1929 году. За почти столетнюю историю техникум зарекомендовал себя как ведущее учреждение среднего профессионального образования, готовящее высококвалифицированных специалистов для автомобильной и машиностроительной отраслей. Техникум активно развивается, внедряя современные образовательные технологии и сохраняя традиции качественной подготовки кадров.

Техникум предлагает широкий спектр образовательных программ, ориентированных на потребности рынка труда. Техникум имеет такие специальности как:

* Информационные системы и программирование.
* Эксплуатация и обслуживание электрического и электромеханического оборудования (по отраслям).
* Мехатроника и мобильная робототехника (по отраслям).
* Технология машиностроения.
* Техническая эксплуатация и обслуживания роботизированного производства.
* Металлургическое производство (по видам производства).
* Автомобиле- и тракторостроение.
* Эксплуатация транспортного электрооборудования и автоматики.
* Экономика и бухгалтерский учет (по отраслям).

Техникум гордится своим педагогическим коллективом, в который входят высококвалифицированные специалисты:

Преподаватели активно участвуют в профессиональных конкурсах, научно-практических конференциях и методических разработках, что способствует постоянному повышению качества образовательного процесса.

Техникум оснащен современной материально-технической базой, которая включает:

* Учебные лаборатории и мастерские, оборудованные по последним стандартам.
* Компьютерные классы с современным оборудованием.
* Библиотеку с обширным фондом учебной и научной литературы.
* Спортивный зал и площадки для занятий физической культурой.

Инфраструктура техникума создает комфортные условия для обучения и всестороннего развития студентов.

Также, техникум включен в программу «Профессионалитет» и является ядром кластера «Машиностроение».

Профессионалитет — это образовательная программа в колледжах, которая позволит студентам стать высококвалифицированным специалистами на ведущих предприятиях региона.

Машиностроение — это отрасль тяжелой промышленности и главный двигатель индустриализации страны. В России более 40 тыс. предприятий, которые производят, проектируют, обслуживают и утилизируют машины, их детали и технологическое оборудование.

Таким образом, техникум имеет передовую программу обучения и оборудование, которое позволяет в более короткий срок выпускать специалистов для своего главного партнера и сооснователя – «автомобильный завод «ГАЗ»», которая является лидирующей компанией по производству малогрузовых и грузовых автомобилей для нужд и потребностей бизнеса.

Дополнительно, техникум и профсоюзная организация «ГАЗ» имеют сильное и налаженное сотрудничество, что позволяет поощрять студентов за различные достижения и активности как внутри, так и за пределами учебного учреждения.

За счет программы «Профессионалитет» - техникум смог спроектировать и построить учебно-производственный комплекс.

Учебно-производственный комплекс Государственного бюджетного профессионального образовательного учреждения «Нижегородский автомеханический техникум» создан 2 сентября 2024 года в рамках реализации Федерального проекта «Профессионалитет» на базе образовательно-производственного центра (кластера) машиностроения Нижегородской области.

Учебно-производственный комплекс создан в целях организации практической подготовки обучающихся по основным профессиональными образовательным программам, основным программам профессионального обучения, дополнительным профессиональным программам, предоставления работы временного характера обучающимся и выпускникам, а также производства товаров, выполнения работ и оказания услуг с использованием материально-технической базы ГБПОУ «НАМТ» по профилю реализуемых образовательных программ.

1.2 Характеристика технических и программны средств предприятия

Технические средства являются неотъемлемой и наиболее существенной составляющей информационной технологии, выполняя ту же роль, что и средства производства в трудовой деятельности.

Основное назначение техники:

* повышение уровня эффективности трудовых усилий человека;
* расширение его возможностей в процессе трудовой деятельности;
* освобождение (полное или частичное) человека от работы в условиях, опасных для здоровья.

Технические средства информационных технологий можно подразделить на следующие группы:

* оргтехника (копиры, сканеры, уничтожители бумаги, брошюровщики и т.д.). Техническое оборудование офиса, облегчающее и ускоряющее бумажную и административно-управленческую деятельность;
* коммуникационная техника (телефоны, модемы, факсы, коммутаторы, маршрутизаторы, концентраторы и т.д.). Основная задача коммуникационной техники - передавать информацию из одного или нескольких источников информации в один или несколько приемников как можно более без искажений;
* устройства и оборудование, оснащенные микропроцессорами. Автоматическая система, представляющая собой функционально законченное изделие, состоящее из одного или нескольких устройств, главным образом из микропроцессора и/или микроконтроллера;

Главным техническим средством на предприятии является компьютер.

Характеристики компьютера, используемого для разработки приложения:

* процессор: AMD Radeon 5600, 3.5 GHz. Электронный блок либо интегральная схема, исполняющая машинные инструкции, главная часть аппаратного обеспечения компьютера или программируемого логического контроллера;
* оперативная память: 32Gb DDR4. В большинстве случаев, это часть системы [компьютерной памяти](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%BF%D0%B0%D0%BC%D1%8F%D1%82%D1%8C), в которой во время работы компьютера хранится выполняемый машинный код ([программы](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0%D1%8F_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0)), а также входные, выходные и промежуточные данные, обрабатываемые [процессором](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81%D0%BE%D1%80);
* монитор. Устройство оперативной визуальной связи пользователя с управляющим устройством и отображением данных, передаваемых с клавиатуры, мыши или центрального процессора;
* видеокарта: NVIDIA GeForce RTX 3060. Устройство, преобразующее графический образ, хранящийся как содержимое памяти компьютера, в форму, пригодную для дальнейшего вывода на экран монитора;
* тип системы: 64-x разрядная версия Windows 10;
* клавиатура. Устройство, позволяющее пользователю вводить информацию в компьютер;
* мышь. Координатное устройство для управления курсором и отдачи различных команд компьютеру.

Компьютерная сеть (компьютерная сеть, сеть передачи данных) - система связи компьютеров или компьютерного оборудования (серверы, маршрутизаторы и другое оборудование), предназначенная для обмена информацией между компьютерами в сети без использования промежуточных носителей данных.

Основным назначением локальной сети организации является:

* хранение данных;
* передача данных с одного компьютера на другой;
* передача данных со считывающих устройств (CD, DVD ROM);
* передача файлов для вывода на принтер, подключенный к одному из компьютеров сети;
* доступ к модему, подключенному к одному из компьютеров, для выхода в Internet.

Для функционирования компьютера (выполнения приложений, или прикладных программ) требуются программные средства, или программное обеспечение. Программные средства – это средства вычислительной техники, реализованные в виде программ.

Программные средства, используемые на предприятии:

* операционная система – Windows 10 – данная операционная система обеспечивает автоматическую настройку необходимых параметров сети при подключении компьютеров к сети;
* среда разработки VS Code - редактор исходного кода, разработанный Microsoft для Windows, Linux и macOS;
* среда разработки Visual Studio - мощная и многофункциональная среда разработки, которая подходит для различных типов проектов и языков программирования. Она предоставляет множество инструментов для повышения продуктивности и качества кода, но требует значительных ресурсов системы и может быть сложной для новичков;
* браузер (Google Chrome) - прикладное программное обеспечение для просмотра страниц, содержания web-документов, компьютерных файлов и их каталогов; управления веб-приложениями; а также для решения других задач;
* язык программирования С# - это современный язык программирования, созданный компанией Microsoft. Он используется для разработки приложений, которые работают на самых разных платформах: компьютерах, телефонах, игровых консолях и в интернете;
* язык программирования C++ - данный язык широко используется для разработки программного обеспечения, являясь одним из самых популярных языков программирования. Область его применения включает создание операционных систем, разнообразных прикладных программ, драйверов устройств, приложений для встраиваемых систем, высокопроизводительных серверов, а также компьютерных игр;
* язык программирования Python - это высокоуровневый язык программирования, отличающийся эффективностью, простотой и универсальностью использования. Он широко применяется в разработке веб-приложений и прикладного программного обеспечения, а также в машинном обучении и обработке больших данных;
* Язык программирования Visual Basic - это объектно-ориентированный язык программирования, разработанный корпорацией Майкрософт. Использование Visual Basic позволяет быстро и легко создавать безопасные приложения .NET;
* Пакет офисных программ Microsoft Office - в состав этого пакета входит программное обеспечение для работы с различными типами документов: текстами, электронными таблицами, базами данных и др;
* Комплекс прикладных программ и драйверов от Nvidia – данные программы обеспечивает контроль над параметрами графических драйверов NVIDIA и другими служебными программами NVIDIA, которые могут быть установлены на компьютере.
  1. Понятие информационной системы и ее компоненты

Информационная система – определенная система или совокупность множества компонентов, которые способна обеспечивать непрерывный доступ к определенной информации и осуществлять взаимодействие с ней для достижения определенной цели.

Основными составными компонентами или частями данной системы являются:

* Аппаратное обеспечение: серверы, компьютеры и различные технические устройста для хранения информации и функционирования самой системы.
* Программной обеспечение: набор программ, обеспечивающие доступ к данным и возможность взаимодействия с ними.
* Данные: основопологающий компонент, который хранится в аппаратной части, обрабатывается программной и является описанием определенной области жизни.
* Человек, люди: определенная целевая аудитория, которая имеет определенную проблему или же запрос, решением которого являются возможности самой системы.
  + 1. Топология информационных систем

Информационные системы бывают таких видов, как:

* Транзационные: основной фукнцией данной системы является отслеживание всех операций компании.
* Управленческие: система, используемая руководителями для принятия стратегических решения, на основе выборки и анализа определенного количества данных.
* Экспертные информационные системы: данные системы используются для решения задач повышенной сложности, на основе базы знаний и искусственного интелекта, который внедрен в данную систему.
* Географические информационные системы: данные системы собирают и обрабатывают множество георграфических данных для решения задач, связанные с экологией, логистикой и городском планировании.
* Системы учета: основной функцией данной системы является ведения учета определенных категорий объектов на территории, в имуществе у самой компании.
  1. Жизненный цикл разработки программного обеспечения

В сфере разработки программного обеспечения используется такой вид жизненного цикла, как SDLC (Software Development Life Cycle), состоящих из 6 основных пунктов, из которых также состоит жизненный цикл проекта:

* Планирование и анализ требований: на данном этапе осущестляется генерация основных идей программного обеспечения и его желаемый функционал.
* Определение требований: данный этап подразумевает сегментацию идей, относительно функционала программы и возможности ее масштабирования.
* Проектирование архитектуры: за счет определения основного функционала программного обеспечения, на данном этапе осуществляет проектирование структруры исходного кода приложения, составление дорожной карты пользователя, базы данных и ее сущностей.
* Разработка: на данном этапе происходит написание исходного кода приложения и бота, в соответствии с поставленными требованиями и спроектированной структурой.
* Тестирование: в ходе самой разработки и после нее программный продукт проходит различные виды тестирования, включая функциональное и нагрузочное, для оценки общей производельности программы и работы ее основных модулей.
* Развертывание и сопровождение: данный этап характеризуется внедрением разработанного программного обеспечения и дальнейшее его сопровождение.

Для разработки проекта использовалась инкрементная модель разработки программного обеспечения, которая позволят за счет оценки продукта внутри одной интерации выявить его недочеты или создать новый функционал, который может быть реализован в следующей итерации.

* 1. Модели и методы проектирования информационных систем

Перед посроением полноценной системы, во всех случаях, нужно ее описать или же создать информационную модель, которая отображает основные сущности выбранной области.

Структура системы – это совокупность необходимых и достаточных элементов, отношений или же сущноностей для достижения конкретной цели.

В общей сложности, существуют такие модели, как:

* модель «Черный ящик»;
* модель «Состава системы»;
* структурная модель

1.5.1 Модели проектирования информационных систем

1.5.1.1 Модель «Черный ящик»

Данная модель используется для описания внешнего воздествия системы на окружающий мир, в виду ненадобности ее внутренного устройства. Сама модель состоит из входа (воздействия окружающего мира на объект), самой системы и выхода (воздействие, оказываемое системой на окружающий мир). Общая модел системы представлена на рисунке 1.

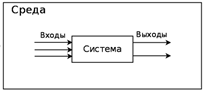


Рисунок 1 – Общая вид модель «Черный ящик»

1.5.1.2 Модель «Состава системы»

Основной задачей данной модели является описание внутренней структуры системы, не имея связей между ее сущностями и дополнительно, отображая иерархию вложенности между ними. Общий вид и пример данной модели представлен на рисунке 2.

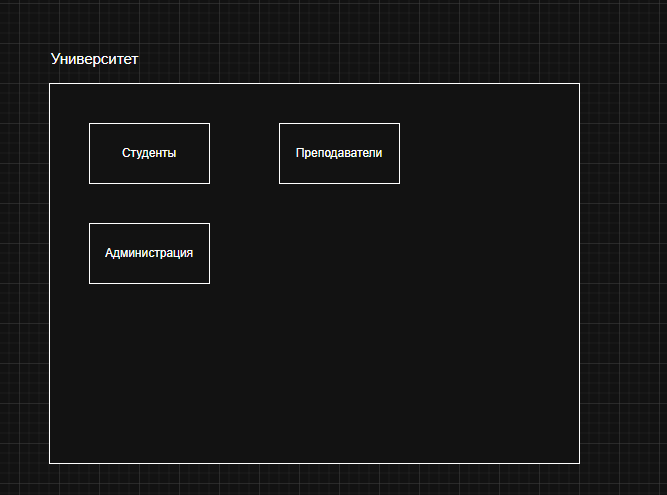


Рисунок 2 – Общие пример модели «Состава системы»

1.5.1.3 Структурная модель

Структурная модель является основной из основных моделей, которая используется для описания информационной системы. Общий вид данной модели предствляет из себя набор фигур или таблиц, каждая из которых описывает определенную сущность и в совокупности, за счет связей, являются описанием определенной области жизни или же предметной области. Одним из представлений струтурной модели является ER-диаграмма, которая описывает определенные сущности в виде таблиц, имеющие логическую и строгую структуру. Связи между сущностями осуществляются за счет организации первичных ключей (уникальный идентификатор записи) и внешних ключей, которые содержат в себе информацию первичного ключа. Общий пример структурной модели, в виде ER – диаграмы, представлен на рисунке 3.

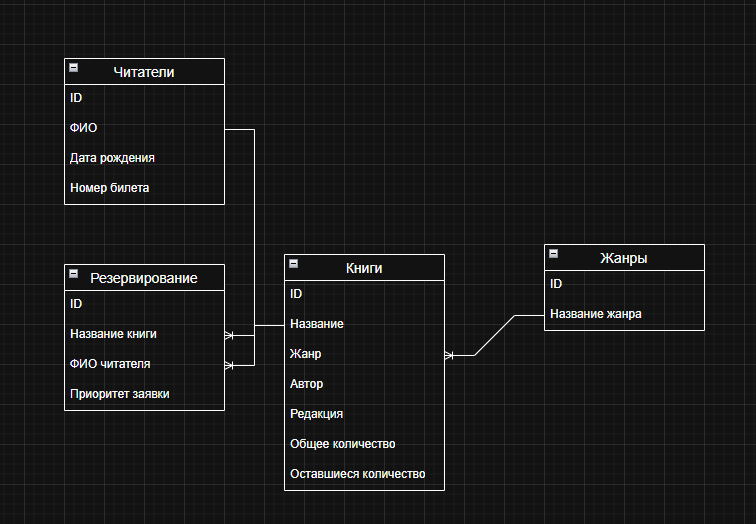


Рисунок 3 – Пример структурной модели

1.5.2 Архитектуры информационных систем

В современном мире информационные системы играют ключевую роль в обеспечении эффективности и конкурентоспособности организаций. Они позволяют автоматизировать бизнес-процессы, оптимизировать работу с данными и предоставлять пользователям необходимые услуги. Одной из важнейших составляющих информационных систем является их архитектура, которая определяет структуру, компоненты и взаимодействие между ними.

Архитектура приложения – это некоторый способ описания самого приложения, его компонентов и их взаимодействия между собой. Архитектура описывает способы взаимодействия, передачу и обработку данных между различными модулями приложения. Цель, которая стоит перед построением архитектуры приложения, являтся обеспечения необходимой производительности работы самого приложения, возможности его расширения и удобности в чтении исходного кода программы. В случае, если архитектура приложения составлена верно – программа будет иметь необходимый уровень производительности и более облегченную возможность добавления нового функционала в приложение.

При построении самой архитектуры выделяют несколько основных принципов:

* Четкое делегирование ответственности: каждый компонент приложения имеет определенную область действия и обязанностей, которые он должен выполнять.
* Наличие модульности: приложение должно состоять из отдельных модулей (файлов), совокупность которых и является самим приложением.
* Составление и применение архитектурных шаблонов: в системе имееются шаблоны, которые применяюся для решения типовых проблем и организуют программу более эффективно.
* Инкапсуляция: приложение имеет сокрытие деталей, которые обеспечивают необходимый уровень взаимодействия между модулями и предотвращения критических ситуаций в работе программы.
* Расширяемость: построенная архитектура должна иметь возможность быстрого добавления нового функционала, без использования большего количества кода.
* Разделение на уровни представления: за счет данного разделения у общей команды должно быть общее представление о фукциях каждого модуля, которое позволяет упорядочить сам код и облегчить их использование.

1.5.2.1 Архитектура «Клиент – сервер»

Клиент-серверная архитектура является одной из самых популярных и широко используемых в информационных системах. Она основана на разделении функций между клиентом и сервером, что позволяет обеспечить эффективное взаимодействие между пользователями и серверами, а также упростить разработку и поддержку системы.

Основные компоненты клиент-серверной архитектуры:

1. Клиент — это программа или устройство, которое отправляет запросы к серверу и обрабатывает полученные данные. Клиенты могут быть представлены в виде веб-браузеров, мобильных приложений, настольных программ и т. д.
2. Сервер — это компьютер или группа компьютеров, которые обрабатывают запросы от клиентов и предоставляют им необходимые данные или услуги. Серверы могут выполнять различные функции, такие как хранение данных, обработка запросов, управление бизнес-логикой и т. п.

Преимущества клиент-серверной архитектуры:

* Централизованное управление данными: все данные хранятся на сервере, что упрощает их управление и обеспечение целостности.
* Масштабируемость: при необходимости можно добавить дополнительные серверы или клиенты, чтобы увеличить производительность системы.
* Удобство обновления и поддержки: изменения в системе можно внести на сервере, и они автоматически будут доступны всем клиентам.
* Недостатки клиент-серверной архитектуры:
* Зависимость от сервера: если сервер выходит из строя, то клиенты не смогут получить доступ к данным или услугам.
* Необходимость обеспечения безопасности: данные, передаваемые между клиентом и сервером, могут быть перехвачены или изменены злоумышленниками.

1.5.2.2 Многослойная архитектура

Многослойная архитектура (или n-tier архитектура) является более сложной и гибкой, чем клиент-серверная. Она предполагает разделение системы на несколько слоёв, каждый из которых выполняет определённые функции. Это позволяет добиться более высокой модульности, масштабируемости и удобства поддержки системы.

Основные слои многослойной архитектуры:

1. Presentation layer (уровень представления) — отвечает за взаимодействие с пользователем, отображение данных и обработку ввода.
2. Business logic layer (уровень бизнес-логики) — содержит алгоритмы и правила, которые определяют поведение системы и обработку данных.
3. Data access layer (уровень доступа к данным) — обеспечивает взаимодействие с базой данных или другими источниками данных.

Количество слоёв и их функции могут варьироваться в зависимости от конкретной системы и её требований.

Преимущества многослойной архитектуры:

* Модульность: каждый слой можно разрабатывать, тестировать и поддерживать независимо от других слоёв.
* Масштабируемость: при необходимости можно добавить новые слои или изменить существующие, чтобы улучшить производительность или функциональность системы.
* Удобство повторного использования кода: код, написанный для одного слоя, можно использовать в других проектах или системах.

Недостатки многослойной архитектуры:

* Сложность разработки и настройки: требуется больше времени и усилий для проектирования и реализации многослойной архитектуры.
* Увеличение overhead (накладных расходов): взаимодействие между слоями может привести к дополнительным затратам на обработку запросов и передачу данных.

2. Анализ предметной области и постановка задачи

2.1 Описание предметной области

Государственное бюджетное профессиональное образовательное учреждение «Нижегородский автомеханический техникум» является образовательным учреждением, основанный при министрестве образования и науки Нижегородской области и включенный в федеральную программу «Профессионалитет», которая направлена на практико-ориентированный подход при обучении. Данный проект позволяет за счет современных методов обучения и оборудования получить специальность, которая востребована на производствах страны и дополнительно, получить опыт применения теоритических знаний совместно с формированием практических навыков.

Сам же техникум выпускает специалистов, которые связаны с областью машиностроения и главным партнером которого является «Горьковский автомобильный завод». В виду наличия данного партнера - у студентов и самого образовательного процесса имеется возможность получения практических навыков по своей специальности и более углубленного изучения работы сферы производства.

За счет множества направлений, по которым происходит обучение, образовательное учреждение имеет определенную структуру, которая позволяет эффективно организовывать как сам учебный, так и рабочий процесс. В структуре данного техникума можно выделить следующие составляющие:

* Администрация.
* Учебная часть.
* Отдел практик.
* Воспитательный отдел.
* Бухгалтерия.
* Хозяйственный отдел.

4 Охрана труда

4.1 Общие требования безопасности

Требования охраны труда распространяются на персонал, эксплуатирующий средства вычислительной техники и периферийное оборудование. Инструкция содержит общие указания по безопасности применения электрооборудования в учреждении. Требования настоящей инструкции являются обязательными, отступления от неё не допускаются. К самостоятельной эксплуатации электроаппаратуры допускаются только специально обученный персонал не моложе 18 лет, пригодный по состоянию здоровья и квалификации к выполнению указанных работ.

4.2 Требования безопасности перед работой на компьютере

Осмотреть и привести в порядок рабочее место. Отрегулировать освещение на рабочем месте, убедится в отсутствии потока встречного света. Проверить правильность подключения электрооборудования в сети. Протереть салфеткой поверхность экрана. Проверить правильность установки стола и клавиатуры.

4.3 Требования безопасности во время работы

Во время работы быть внимательным, не отвлекаться посторонними делами и разговорами. Рабочее место должно быть оборудовано так, чтобы исключить неудобные позы и длительные статические напряжения тела. При работе на ПК должна быть исключена возможность одновременного прикосновения к оборудованию и частям помещения или оборудования, имеющим соединение с землёй (радиатора батарей, металлоконструкций). Во время работы нельзя класть на монитор бумаги, книги и другие предметы, которые могут закрыть его вентиляционные отверстия. Запрещается оставлять без присмотра включенное оборудование; вскрывать устройства ПК.

4.4 Требования безопасности после окончания работы

Отключить ПК от сети, штепсельную вилку при этом держась за корпус. Запрещается отключать ПК за электропривод. При отключении ПК со съёмным шнуром питания сначала необходимо отключить вилку от розетки, а затем отключить питающий шнур от ПК. Привести в порядок рабочее место. Чистку ПК от пыли необходимо производить только после отключения ПК от сети.

4.5 Требования безопасности в аварийных ситуациях

При обнаружении неисправности немедленно обесточить электрооборудование, оповестить администрацию. Продолжение работы возможно только после устранения неисправности.

При обнаружении оборвавшегося провода необходимо немедленно сообщить об этом администрации, принять меры по исключению контакта с ним людей. Прикосновение к проводу опасно для жизни.

Во всех случаях поражения человека электрическим током немедленно вызывают врача. До прибытия врача нужно, не теряя времени, приступить к оказанию первой помощи пострадавшему.

Необходимо немедленно начать производить искусственное дыхание, наиболее эффективным из которых является метод рот в рот или рот в нос, а также наружный массаж сердца. Искусственное дыхание поражённому электрическим током производится вплоть до прибытия врача.

На рабочем месте запрещается иметь огнеопасные вещества.

В помещениях запрещается:

* зажигать огонь;
* включать электрооборудование, если в помещении пахнет газом;
* курить;
* сушить что-либо на отопительных приборах;
* закрывать вентиляционные отверстия в электроаппаратуре.

Источниками воспламенения являются:

* искра при разряде статического электричества;
* искры от электрооборудования;
* искры от удара и трения;
* открытое пламя.

При возникновении пожароопасной ситуации или пожара персонал должен немедленно принять необходимые меры для его ликвидации, одновременно оповестить о пожаре администрацию. Помещения с электрооборудованием должны быть оснащены огнетушителями типа ОУ-2 или ОУБ-3.

Заключение

В ходе выполнения курсового проекта было разработано приложение, осуществляющие взаимодействие с базой данных «Железнодорожный вокзал» и обеспечивающие автоматизация бизнес процессов, которые происходят на самом вокзале. Также, за счет разработанных алгоритмов подгрузки данных в элементы управления comboBox и автоматизации поиска в большом количестве данных - обеспечивается более интуитивное и удобно управление.

Цели и задачи курсового проекта выполнены успешно.

Список использованных источников

1. Положение по организации выполнения и защиты курсовой работы (проекта) в ГБПОУ «НАМТ», 2024 г.
2. Стандарт предприятия ГБПОУ «НАМТ», 2017 г.
3. Методические указания по выполнению курсового проектирования по МДК.01.01. Разработка программных модулей для студентов специальности 09.02.07 Информационные системы и программирование, 2023 .
4. Бодров О.А., Медведев Р.Е. Предметно-ориентированные экономические информационные системы. - М.: Горячая линия - Телеком, 2013. -250 С.
5. Громов, А. Современные подходы к автоматизации управления железнодорожным транспортом / Громов, А. // Транспорт и логистика. -2020. -№ 86. С. 45-54.
6. Иванов, С. Оптимизация работы железнодорожных вокзалов с использованием информационных технологий / Иванов, С // Информационные технологии. -2018. - № 12. С. 56-68.
7. Петров, В. Проектирование информационных систем на Visual Basic. / Петров, В // Москва: ДМК Пресс. -2015. - № 4. С. 5-10.
8. Рудаков А.В. Технология разработки программных продуктов. Практикум: учеб. пособие для студ. учреждений сред. проф. образования / А.В. Рудаков, Г.Н. Федорова. - 4-е изд. Стер. - М.: Издательский центр «Академия», 2014. -356 С.
9. Сидоров, К. Применение Visual Basic для автоматизации процессов на железнодорожных вокзалах / Сидоров, К // Программные решения. -2019. -№ 67. С. 5-20.
10. Федорова Г.Н Разработка модулей программноо обеспечения для компьютерных систем: учебник для студентов среднего профессионального образования / Г.Н. Федорова. – 3-е ездание. –М.: Издательский центр «Академия», 2019. -384 С.
11. Шилоносов, М. Программирование баз данных на Visual Basic, Москва: МК-Пресс, 2010. -257 С.

Приложение 1

Imports MySql.Data.MySqlClient

Module SQL

Public constring As MySqlConnection

Public command As MySqlCommand

Sub connection(ByRef connBool As Boolean)

Try

constring = New MySqlConnection("server=127.0.0.1; database=железнодорожный\_вокзал; username=root")

constring.Open()

connBool = True

Catch ex As Exception

MsgBox("Connection was failed [ " & ex.Message & " ]")

connBool = False

End Try

End Sub

End Module

Public Class main

Dim connBool As Boolean

Dim i As Integer

Dim dr As MySqlDataReader

Dim picker As String

Private Sub main\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

Try

connection(connBool)

If connBool Then

Dim SucColor = RGB(0, 255, 0)

lblCon.Text = "Connection: True"

lblCon.BackColor = Color.LightGreen

End If

Catch ex As Exception

MsgBox(ex.Message)

lblCon.Text = "Connection: False"

lblCon.BackColor = Color.Red

End Try

loadData\_rice()

loadData\_Schedule()

End Sub

Public Sub loadData\_rice()

Try

Dim cmd As New MySqlCommand("Select \* from рейсы", constring)