# АННОТАЦИЯ

*Авторы выпускной квалификационной работы*: Тартынский Алексей Валерьевич, Красильников Павел Витальевич

*Тема ВКР:* «Разработка информационной системы автоматизации процесса учёта деятельности на комбинате питания ГПБОУ «Кемеровский областной медицинский колледж», г. Кемерово». *Специальность:* Информационные системы и технологии.

*Город:* г. Кемерово.

*Год выполнения:* 2017.

*Количество страниц: 146*

*Количество таблиц:* 28

*Количество иллюстраций:* 97

*Описание:* Данная работа описывает разработку информационной системы по автоматизации процесса учёта деятельности на комбинате питания. Система функционирует на базе СУБД SQL Server. В работе рассматриваются вопросы: аппаратной и административной интеграция, общие вопросы администрирования, вопросы информационной безопасности, разработка системы централизованного хранения и обработки данных, технология разработки и программной реализации, расчет экономической эффективности. Особенностью системы является автоматическая генерация меню на каждый день.

**СОДЕРЖАНИЕ**

[АННОТАЦИЯ 2](#_Toc485614945)

[ВВЕДЕНИЕ 6](#_Toc485614946)

[1 Разработка технического задания 8](#_Toc485614947)

[1.1 Исходное задание на проектирование 8](#_Toc485614948)

[1.2 Анализ организационно-технологической структуры предприятия 8](#_Toc485614949)

[1.2.1 Название организации, статус организации, форма юридического лица 8](#_Toc485614950)

[1.2.2 Масштаб организации 9](#_Toc485614951)

[1.2.3 Общие принципы функционирования организации 9](#_Toc485614952)

[1.2.4 Вид и профиль деятельности 9](#_Toc485614953)

[1.2.5 Цели функционирования организации 10](#_Toc485614954)

[1.2.6 Организационная структура предприятия 11](#_Toc485614955)

[1.2.7 Обследование функций, структуры и деятельности базового подразделения. 14](#_Toc485614956)

[1.2.8 Выявление бизнес-процессов базового подразделения. 17](#_Toc485614957)

[1.2.9 Анализ существующей реализации базового бизнес-процесса. 22](#_Toc485614958)

[1.3 Формулировка требований к системе 23](#_Toc485614959)

[1.3.1 Определение деятельностей, требующих автоматизации, кандидатов в требования. 23](#_Toc485614960)

[1.3.2 Определение состава функциональных требований к системе. 27](#_Toc485614961)

[1.3.3 Разработка сценариев, реализующих требования к системе. 32](#_Toc485614962)

[1.3.4 Определение требований к интерфейсу. 37](#_Toc485614963)

[1.3.5 Окончательная формулировка требований к системе. 41](#_Toc485614964)

[2 Анализ и проектирование 48](#_Toc485614965)

[2.1 Определение классов анализа 48](#_Toc485614966)

[3 Разработка системы централизованного хранения и обработки данных 62](#_Toc485614967)

[3.1 Формирование требований к БД 62](#_Toc485614968)

[3.2 Формирование отношений БД 63](#_Toc485614969)

[3.3 Даталогическое проектирование 64](#_Toc485614970)

[3.4 Разработка сценариев работы с данными 70](#_Toc485614971)

[3.5 Реализация сценариев работы с данными 72](#_Toc485614972)

[4 Использование библиотеки Nlog для учёта активности пользователей web-сайта. 76](#_Toc485614973)

[4.1 Описание технологии. 76](#_Toc485614974)

[4.2 Пример использования. 76](#_Toc485614975)

[5 Использование библиотеки MVC Site Map Provider для разграничения прав доступа пользователей к ресурсам web-сайта. 79](#_Toc485614976)

[5.1 Описание технологии. 79](#_Toc485614977)

[5.2 Пример использования. 80](#_Toc485614978)

[6 Технологии разработки и программная реализация 83](#_Toc485614979)

[6.1 Выбор операционной системы 83](#_Toc485614980)

[6.2 Выбор типа взаимодействия пользователя с системой, технологии пользовательского интерфейса 83](#_Toc485614981)

[6.3 Выбор технологии взаимодействия пользовательских компонент с данными 83](#_Toc485614982)

[6.4 Выбор языка и среды программирования 84](#_Toc485614983)

[6.5 Определение состава компонент 84](#_Toc485614984)

[6.6 Разработка схемы развертывания 85](#_Toc485614985)

[6.7 Требования к системе 85](#_Toc485614986)

[6.8 Разработка компонент 86](#_Toc485614987)

[6.9 Характеристика разработанного решения 94](#_Toc485614988)

[6.10 Разработка вопросов тестирования 96](#_Toc485614989)

[7 Аппаратная и административная интеграция ИС 106](#_Toc485614990)

[7.1 Разработка схемы развертывания 106](#_Toc485614991)

[7.2 Формулировка требований к физическим устройствам и сетевому оборудованию, состав рабочих мест 107](#_Toc485614992)

[7.3 Выбор состава аппаратных средств 107](#_Toc485614993)

[7.4 Расчет потребности персонала. 108](#_Toc485614994)

[7.5 Выбор сетевой архитектуры и технологии 108](#_Toc485614995)

[7.5.1 Выбор архитектуры 108](#_Toc485614996)

[7.5.2 Выбор технологии и аппаратных средств. Расчет сети 109](#_Toc485614997)

[8 Общие вопросы администрирования 111](#_Toc485614998)

[8.1 Определение стратегии администрирования на уровне руководства. 111](#_Toc485614999)

[8.2 Определение объектов администрирования на уровне предприятия. 111](#_Toc485615000)

[8.3 Политика администрирования на уровне предприятия. 111](#_Toc485615001)

[8.4 Определение объектов администрирования на уровне разрабатываемой ИС. 112](#_Toc485615002)

[8.5 Политика администрирования на уровне разрабатываемой ИС. 112](#_Toc485615003)

[9 Вопросы информационной безопасности 115](#_Toc485615004)

[9.1 Анализ угроз 115](#_Toc485615005)

[9.2 Информационная безопасность на уровне предприятия. 117](#_Toc485615006)

[9.2.1 Контроль доступа в помещения. 117](#_Toc485615007)

[9.2.2 Обеспечение безопасности с помощью аппаратных средств. 117](#_Toc485615008)

[9.2.3 Обеспечение безопасности с помощью программных средств. 117](#_Toc485615009)

[9.2.4 Определение политики управления доступом пользователей. 118](#_Toc485615010)

[9.2.5 Требования к административным обязанностям. 118](#_Toc485615011)

[9.2.6 Антивирусная защита информации. 119](#_Toc485615012)

[9.2.7 Информационная безопасность на уровне АИС. 120](#_Toc485615013)

[9.2.8 Защита персональных данных. 120](#_Toc485615014)

[10 Экономическое обоснование разработки ИС 122](#_Toc485615015)

[10.1 Расчет затрат на разработку системы 122](#_Toc485615016)

[10.2 Расчёт эксплуатационных затрат 127](#_Toc485615017)

[10.3 Экономическая эффективность 131](#_Toc485615018)

[10.4 Срок окупаемости 132](#_Toc485615019)

[10.5 Технико-экономические показатели проекта 132](#_Toc485615020)

[Список используемой литературы 135](#_Toc485615021)

[ЗАКЛЮЧЕНИЕ 136](#_Toc485615022)

[ПРИЛОЖЕНИЕ 1. Техническое задание 137](#_Toc485615023)

[ПРИЛОЖЕНИЕ 2. Схема ЛВС 144](#_Toc485615024)

[ПРИЛОЖЕНИЕ 3. Организационная структура колледжа 145](#_Toc485615025)

[ПРИЛОЖЕНИЕ 4. Схема управления организационными структурами КОМК 146](#_Toc485615026)

# ВВЕДЕНИЕ

Системы автоматизации деятельности данного предприятия прошли процесс становления от простейших форм бумажного учёта деятельности до современных автоматизированных систем с использованием ПК.

На предприятии уже на текущий момент времени ведётся активная интеграция собственной ИС учёта, которая замещает устаревшую систему, построенную на базе MS ACCESS и частично бумажном документообороте.

Система автоматизации обеспечения питания является важной частью в процессе перехода к более современному виду автоматизации деятельности на данном предприятии. Благодаря данной разработке оперативным образом решаются организационные вопросы по организации питания в смежных системах. Складской работник, отслеживает количество продуктов на складе. Работник по столовой, может осуществить запрос на необходимость определённых ингредиентов в определённом количестве, а при отсутствии каких-либо ингредиентов, существует возможность сформировать поставку с определённым набором ингредиентов.

Таким образом, качеств, скорость и эффективность работы современного учреждения неразрывно связано с наличием современной системой автоматизации деятельности, который позволяет решить определённый круг задач для текущей организации, такие как:

1. Автоматизация БП, следовательно, улучшение качества и ускорение деятельности предприятия;
2. Автоматизация связей между внутренними структурными подразделениями предприятия.

Цель дипломной работы: создание внутреннего веб-портала для *Кемеровского областного медицинского колледжа*. Система создаётся для сокращения временных затрат на получение и предоставление информации связанной с деятельностью *столовой предприятия* и повышение качества и скорости работы формирования меню, учёта складских поставок.

Для реализации поставленной цели необходимо решить следующие задачи:

* провести анализ деятельности предприятия;
* изучить особенности предоставления информации о деятельности образовательного учреждения;
* проанализировать и спроектировать информационную систему;
* разработать систему централизованного хранения и обработки данных;
* разработать и реализовать возможность учёта действий пользователя веб-портала в виде системы логирования;
* разработать и реализовать программную реализацию с удобным интерфейсом;
* осуществить аппаратную и административную интеграцию портала;
* разработать политику администрирования портала;
* разработать требования к информационной безопасности;
* расcчитать экономическую эффективность ИС.

# Разработка технического задания

## Исходное задание на проектирование

Требуется автоматизировать процесс формирования меню и поставок.

В системе присутствуют следующие пользователи: сотрудник кухни, сотрудник склада, менеджер, администратор.

Формирование ежедневного меню с ценами. Цена блюда складывается из стоимости ингредиентов и человеко-часов на приготовление блюда. Все составляющие цены блюда могут меняться каждый день. Необходима возможность расчета и просмотра цены блюда.

## Анализ организационно-технологической структуры предприятия

### Название организации, статус организации, форма юридического лица

Целевым предприятием является государственное бюджетное профессиональное учреждение «Кемеровский областной медицинский колледж» (далее КОМК). Официальная дата открытия колледжа – 1995 год, данная организация является юридическим лицом.

Информация об учредителе:

В соответствии с п.1.7 Устава учредителем ГБПОУ «Кемеровский областной медицинский колледж» является Кемеровская область.

Функции и полномочия учредителя Колледжа от имени Кемеровской области осуществляет департамент охраны здоровья населения Кемеровской области.

Полное наименование учредителя: Департамент охраны здоровья населения Кемеровской области (сокращенно ДОЗН КО).

Адрес: 650064, Кемеровская область, г. Кемерово, пр. Советский, 58

Руководство: начальник департамента охраны здоровья населения Кемеровской области Шан-Син Владимир Михайлович.

### Масштаб организации

КОМК является крупной организацией. Штат головного учреждения насчитывает 650 человек. Годовой доход КОМК составляет 243,872 млн. рублей. Колледж имеет филиалы в других городах. В их числе:

* Анжеро-Судженский филиал (654041, Кемеровская область, г. Анжеро-Судженск, ул. Ленина, 18а);
* Беловский филиал (652600, Кемеровская область, г. Белово, ул. Юности,19);
* Мариинский филиал (652154, Кемеровская область,г. Мариинск,ул. Добролюбова, д.1.);
* Ленинск-Кузнецкий филиал (652515, Кемеровская область, г. Ленинск-Кузнецкий, ул. Зварыгина, д. 6);
* Междуреченский филиал (652877, Кемеровская область, г. Междуреченск, ул. Кузнецкая, д. 39);
* Новокузнецкий филиал (654041, Кемеровская область, г. Новокузнецк, ул. Кузнецова, 33).

### Общие принципы функционирования организации

Деятельность КОМК проводится в соответствии со следующими общими принципами функционирования организаций:

* - Скалярный принцип (scalaris - ступенчатый) - иерархичность построения организаций.
* - Принцип функциональности - должностные обязанности на каждой ступени управления определяются настолько детально, насколько возможно.
* - Принцип диапазона контроля – руководитель умственной деятельности имеет не более 8 человек.
* - Принцип личной ответственности - ответственность руководителя за действия подчинённых ему людей в полном объёме.

### Вид и профиль деятельности

Основная деятельность организации включает:

1. Обучение абитуриентов по специальностям:

* + «Лечебное дело», профиль **фельдшер**;
  + «Акушерское дело», профиль **акушерка**;
  + «Сестринское дело», профиль **мед. сестра/брат;**
  + «Лабораторная диагностика», профиль **медицинский лабораторный техник**;
  + «Фармация», профиль **фармацевт**;
  + «Стоматология», профиль **зубной техник**.

2. Предоставление услуг по повышению квалификации младшего мед. персонала и людей смежных профессий (МЧС, Пожарные службы).

Цели функционирования организации

Основной целью КОМК является выпуск дипломированных специалистов по указанным выше специальностям на выделенные бюджетные средства. Ежегодный выпуск организацией составляет около 800 молодых специалистов. В дальнейшем планируется увеличение числа выпускников до 900 в год. В следствие этого рассчитывается убедить государственный аппарат в выделении большего объема бюджетных средств на развитие организации.

Не менее важной целью является повышение квалификации уже состоявшихся работников здравоохранительных структур, их обучение современным практикам и технологиям.

Для поддержки выполнения основных целей функционирования, выделяется отдельная цель: Обеспечение бесперебойной работы колледжа - поддержание колледжа в работоспособном режиме.

Для разработки модели «Цели функционирования предприятия» используется диаграмма классов (class diagram), которая представлена на рисунке 1.1.

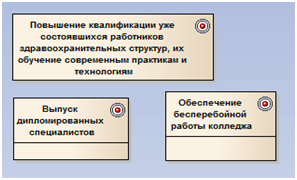


Рисунок 1.1 Цели функционирования КОМК

Организационная структура предприятия

Организационная структура Кемеровского областного медицинского коллежа приведена в приложении 3.

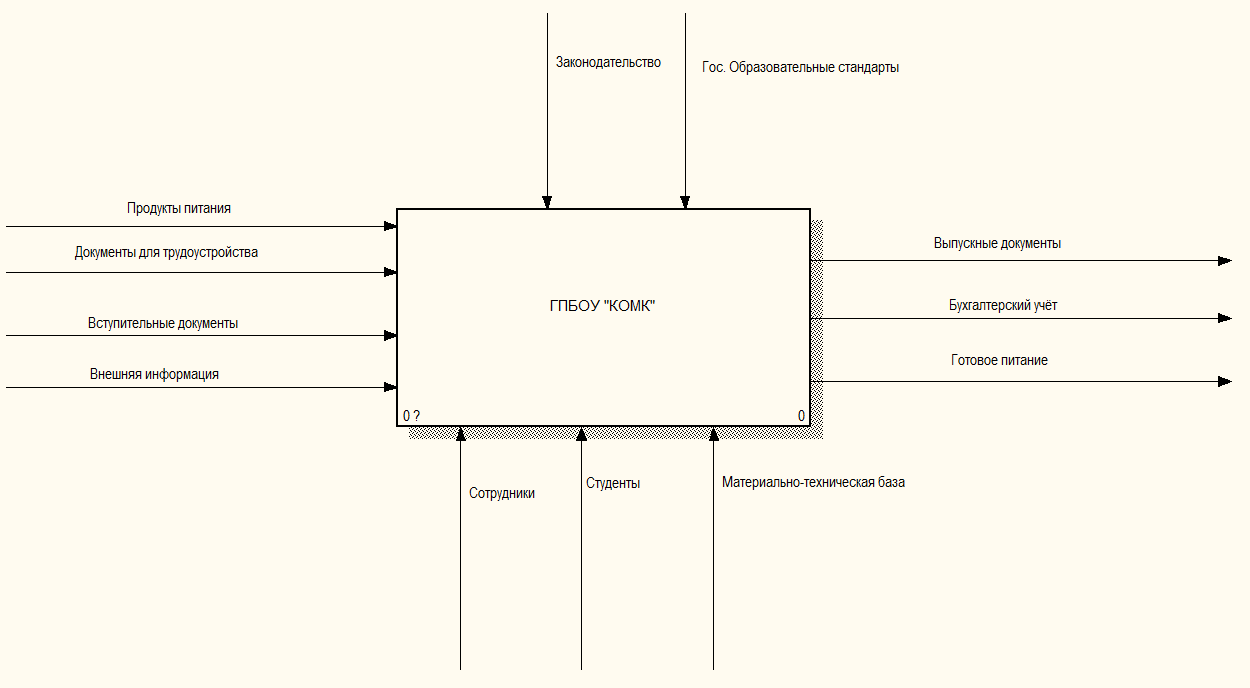
#### Форма отчета о подразделениях предприятия

Подразделение УОП (Управление обеспечением питания) занимается поддержкой рациона питания персонала и студентов, также ведёт аудит поставок продуктов, оценивает свежесть таковых на складских помещениях.

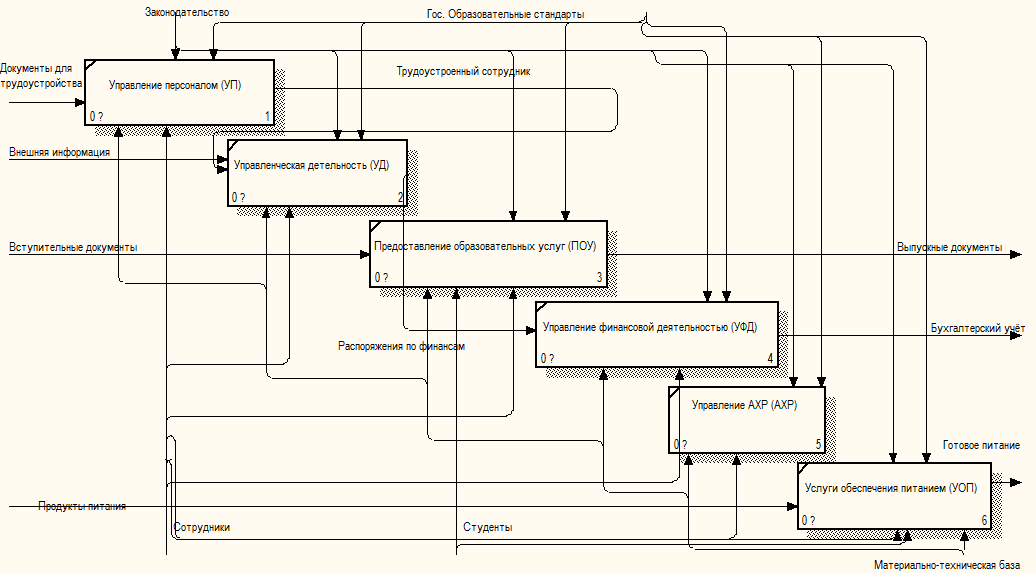
В соответствии с «Положением об отделе сопровождения системной инфраструктуры» предприятия УОП:

* Проводит работы по составлению графика поставок продуктов питания на склад предприятия.
* Проводит ревизию прибывших продуктов питания на склад на предмет качества.
* Проводит учёт продуктов питания на складах предприятия на предмет количественного наличия определённого вида продукции.
* Проводит формирование ежедневного меню на комбинате питания.
* Проводит приготовление первых и вторых блюд под сформированное ежедневное меню.

#### Контекстная диаграмма в нотации IDEF0 для всей организации и ее декомпозиция на составляющие бизнес-процессы (до уровня базового БП).



**Рисунок 1.2 Контекстная диаграмма**



**Рисунок 1.3 Декомпозиция контекстной диаграммы**

#### Общий перечень бизнес-процессов организации и их краткая характеристика, описание.

Таблица 1.1 Перечень бизнес-процессов КОМК.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Код бизнес-процесса БП | Наименование бизнес-процесса БП | Степень важности процесса | Отдел владелец БП | Отдел исполнитель БП | БП – Источник информации | БП – Приемник выходной информации |
| Основные | | | | | | |
| УП | Управленческая деятельность | А | Д | АХР | АХР | Д, Б |
| ПОУ | Предоставление образовательных услуг | А | УМ, У | УМ, ДО |  |  |
| Обеспечивающие | | | | | | |
| УП | Управление персоналом | Б | К | К |  | Д |
| УФД | Управление финансовой деятельностью | Б | Б | Б | Д |  |
| АХР | Управление АХР | В | АХР | АХР |  |  |
| УОП | Услуги обеспечения питанием | А | УОП | УОП |  |  |

#### Описание средства реализации процессов обмена между подразделениями и с внешними объектами.

Взаимодействие с внешними организациями происходит средствами:

* Бумажные носители;
* Средствами электронной почты;
* Государственные порталы (СУФД, Сбербанк АСТ, Парус, Госуслуги, ГИВЦ, РТС-Тендер и др.);
* ПО «Документооборот КОМК» 0.9.2.5. (частично реализует обмен документами между подразделениями).

Виды передаваемой информации:

* Отчеты в надзорные органы (Роспотребнадзор, Рособранадзор, ДОЗН, Роскомнадзор и др.)
* Информация о приемной компании (ФИС ГИА).
* Закупочная документация (Торговые площадки).
* Зарплатный пакет (Банки).

#### Описание схемы управления предприятием, бизнес-процесс управления на уровне предприятия, его организационную, методологическую, программную и аппаратную реализацию.

Управление КОМК осуществляется под руководством Директора. От него поступают указания заместителям по деятельностям КОМК (прим. Зам. директора по Учебной Работе). Они в свою очередь координируют работу подчиненных им отделений. Полная схема управления предприятием представлена в приложении 3. В пределах предприятия частично автоматизирован документооборот. Данную функцию выполняет ПО «Документооборот КОМК» 0.9.2.5.

#### Краткая характеристика базового бизнес-процесса и обоснование его выбора.

В качестве автоматизируемого бизнес-процесса был выбран процесс услуги обеспечения питанием. Данный бизнес-процесс был выбран по причине отсутствие ПО предназначенного для данной деятельности на предприятии как такового.

### Обследование функций, структуры и деятельности базового подразделения.

#### Название подразделения, цели подразделения, выполняемые функции, результаты деятельности подразделения

Базовым подразделением, на котором лежит обеспечение выбранного для автоматизации бизнес-процесса, является «Управление обеспечения питанием».

Целями данного подразделения являются:

* Обеспечение питания персонала колледжа и его студентов;
* Учёт поставок продуктов;
* Учёт продуктов на складских помещениях;
* Ведение документации и подготовка отчетных данных колледжа по вопросам, связанными с данным процессом;
* Формирование меню.

Подразделение выполняет следующие функции:

* Составление графика поставок продуктов питания;
* Ревизия прибывших продуктов питания;
* Учёт продуктов питания на складах;
* Формирование меню.

Результатом функционирования подразделения являются:

* Сметы о поставках;
* Документация по учёту количества продуктов питания;
* Сформированное ежедневное меню.

#### Общедоступные средства ввода-вывода, отображения, обмена информацией.

Обмен данными между подразделениями, в частности документооборот Частично автоматизированы ПО «Документооборот КОМК» 0.9.2.5. Ручная передача документов так же все еще в ходу.

#### Описание процесса обмена информацией между рабочими местами и с внешними объектами.

Процесс обмена информации между местами происходит за счет локальной сети, которая подключена ко всем рабочим станциям, так же обмен информацией происходит через БД, которая расположена на главном сервере КОМК.

На серверах КОМК осуществлены следующие средства защиты:

На Windows серверах установлен и настроен Kaspersky endpoint security 10, на linux – настроен iptables в качестве брандмауэра. На пограничном маршрутизаторе mikrotik также настроен брандмауэр. Присутствует разграничение пользователей по ролям при помощи Active directory, а так же разграничение трафика по vlan.

#### Описание штатного расписания сотрудников и выполняемых ими функций.

В соответствии с должностной инструкцией работника в системе управления питания предприятия, данный работник должен:

* + Вести рабочую деятельность в соответствии с санитарными нормами
  + Рассчитывать количество ингридиентов в блюдах соответственно указанным нормам
  + Выполнять готовку блюда за определённый – установленный срок времени
  + Добросовестно вести учёт продукции на складах
  + Контролировать движения ингредиентов со складка.
  + Контролировать работу своих подчинённых в соответствии с нормами подразделения

Работает в режиме нормированного рабочего дня с 8.00 до 15.00, с понедельника по субботу.

* ***Документы, регламентирующие деятельность;***

Деятельность подразделения регламентируют следующие документы:

* Положение об отделе УОП;
* Должностная инструкция начальника отдела УОП
* Должностная инструкция работника УОП
* ***Входящие документы, исходящие документы в виде схемы документооборота и табеля документооборота.***

Схема документооборота подразделения, табл. 1.2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Номер документа | Наименование документа | Откуда приходит / исходит документ | Куда уходит | Операции, выполняемые над документом | Ответственный за выполнение операций над документом |
| СЗ | Служебная записка | Источник любой | АХР | Формирование записки | Сотрудник УОП |
| АПОС | Акт о перемещении ОС | Начальник УОП | Б (Бухгалтерия) | Составление акта | Глав. бух. |

Табель документооборота, табл. 1.3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Номер документа | Наименование документа | Тип документа | Частота документа за временной период (раз/месяц, раз/раб. день) | Ответственный за документ |
| СЗ | Служебная записка | Внутренний, исходящий | > 2 раз/раб.день | Сотрудник УОП |
| АПОС | Акт о перемещении ОС | Внутренний, исходящий | 5 раз/месяц | Глав. бух. |

* ***Перечень имеющихся организационных документов.***
* *Положение об организации рационального питания в столовой предприятия.*
* *Дневное меню столовой КОМК.*

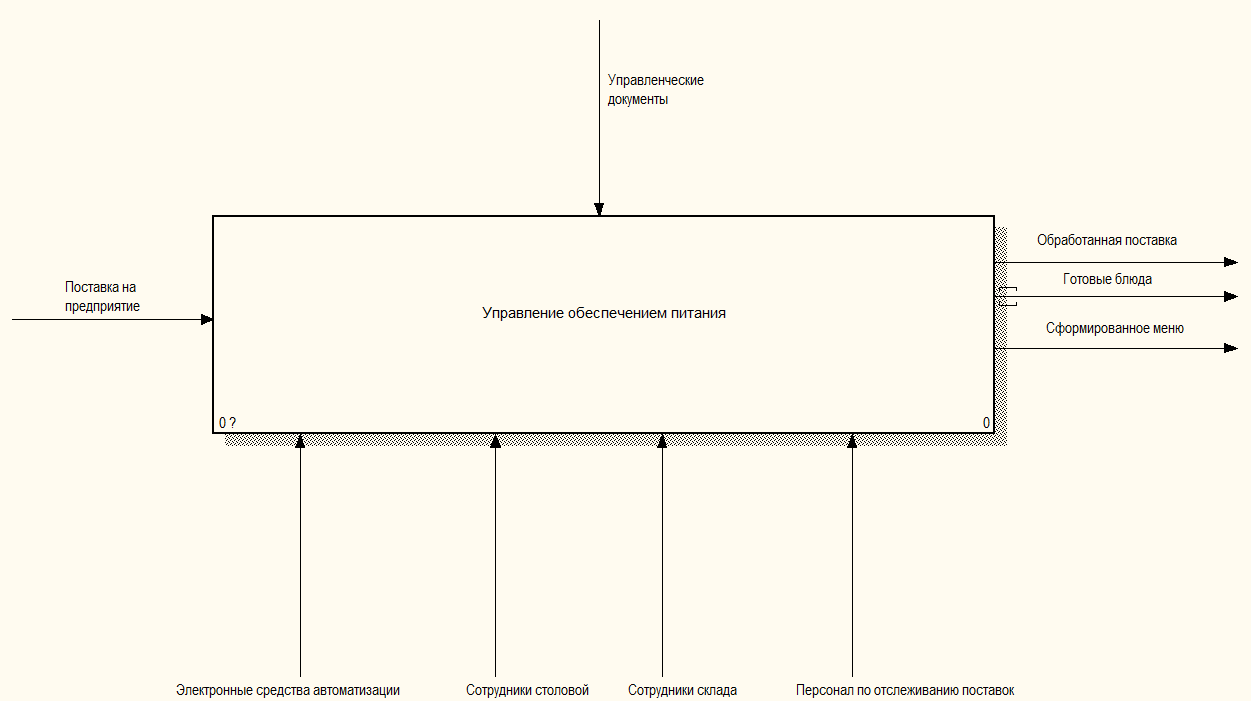
### Выявление бизнес-процессов базового подразделения.

#### Перечень и описание бизнес-процессов, выполняемых подразделением, представленные в табличном виде.

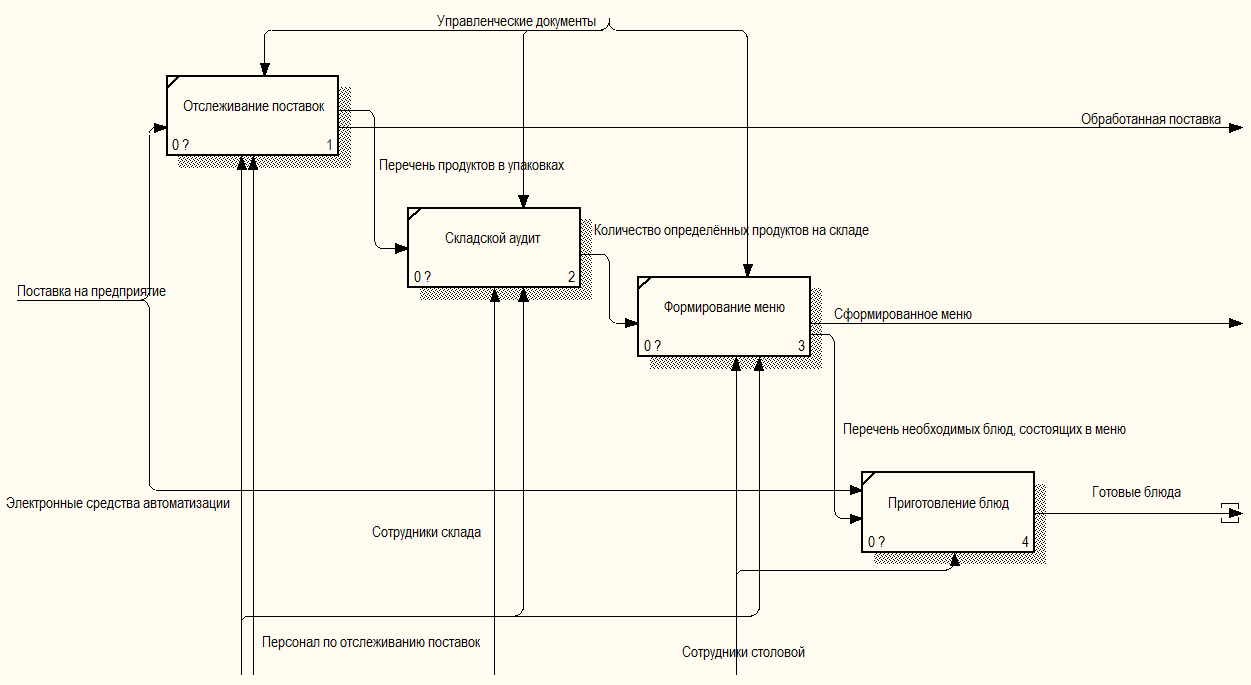
*Столовая предприятия обеспечивает следующие разновидности бизнес-процессов, табл. 1.4*

|  |  |  |  |
| --- | --- | --- | --- |
| Код БП | Название БП | Описание БП | Средства автоматизации |
| ОП | Отслеживание поставок | Процесс обеспечивающий учёт необходимого количества продукции | «КОМК» (MS Access) |
| ФМ | Формирование меню | Процесс обеспечения формированием меню на каждый день | «КОМК» (MS Access) |
| СУ | Складской аудит | Процесс, обеспечивающий  контроль целостности запасов продуктов питания | «КОМК» (MS Access) |

#### Состав и взаимосвязь бизнес-процессов базового подразделения, представленная в виде функциональной диаграммы в нотации IDEF0.

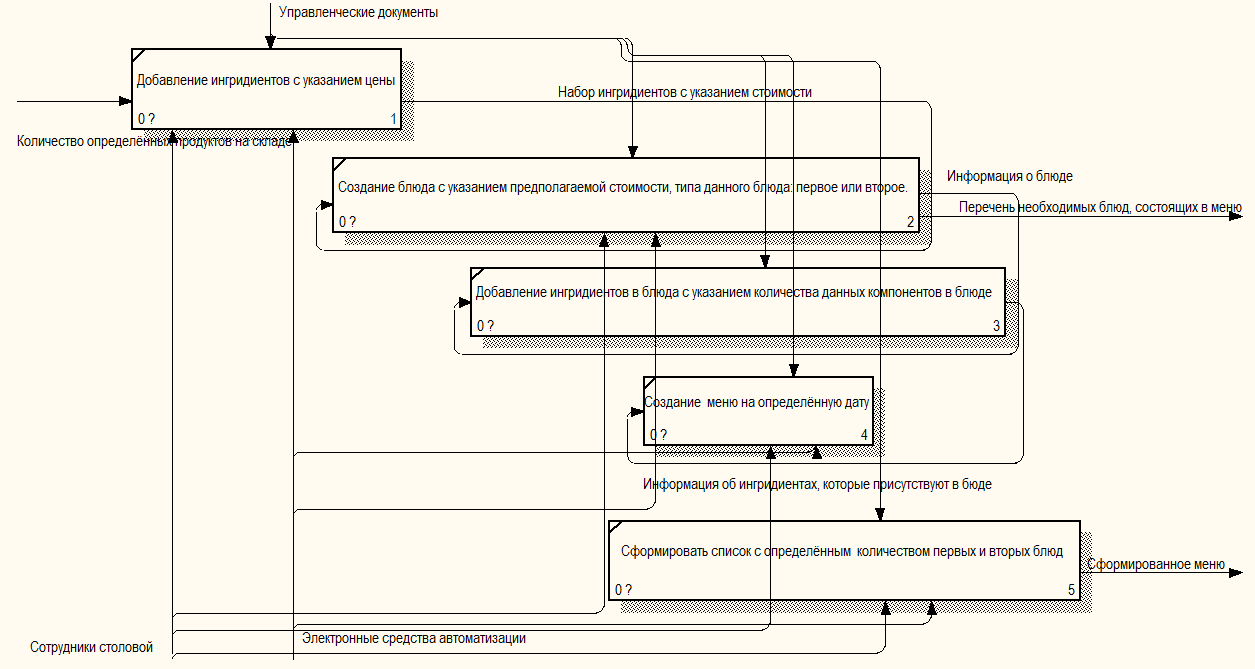


**Рисунок 1.4 Контекстная диаграмма Управления обеспечением питания**



**Рисунок 1.5 Декомпозиция деятельности «Управление обеспечением питания»**

Базовым бизнес-процессом является: «Формирование меню». Процесс: «Формирование меню» обеспечивает столовую предприятия свежим меню на определённые дни. Входящими документами являются документы о перечне продуктов питания, количество определённых продуктов питания на складах. Выходным документом является сформированное меню. Целью бизнес-процесса является формирование меню.



**Рисунок 1.6 Декомпозиция базового БП: «Формирование меню»**

#### Подробное описание и анализ базового БП в табличном виде.

Далее выполняется подробное описание базового бизнес-процесса. Для этого необходимо привести следующие данные, табл. 1.5

|  |  |
| --- | --- |
| Необходимые данные | Значение |
| Условия начала выполнения бизнес-процесса | Поступление поставки ингредиентов на склад сигнализирует о необходимости формирования меню. |
| Документы и данные, необходимые для выполнения бизнес-процесса, их источники. | Необходимые документы и данные определяются индивидуально для каждой поставки. При поступлении поставки осуществляется добавление информации об ингредиентах с указанием их стоимости. |
| Документы, создаваемые в результате выполнения бизнес-процесса, и их получатели. | В результате выполнения БП формируется запись в БД КОМК о сформированных меню. Возможен доступ к просмотру уже добавленных ранее в БД видах меню. |
| Действующие лица, принимающие участие в выполнении бизнес-процесса. | Действующих лиц можно разделить на 2 группы:   * Сотрудники столовой КОМК * Сотрудник склада КОМК |
| Материальные ценности, необходимые для выполнения бизнес-процесса, если таковые есть. | Для формирования меню необходимо наличие заранее добавленных ингредиентов и добавленных блюд в БД. |
| Материальные ценности – результат выполнения бизнес-процесса, если таковые есть. | - |
| Результаты выполнения бизнес-процесса (кроме материальных ценностей). | Результатом выполнения БП считается является сформированное меню. |
| Цель данного бизнес-процесса, его место и роль в общих задачах (процессах) компании. | Ускорение процесса формирования меню, повышение производительности работы столовой. |
| Проблемы, возникающие при выполнении бизнес-процесса. | Нехватка ингредиентов на складах, нарушение условий по срокам поставок со стороны поставщиков |
| Нештатное завершение (выполнение) бизнес-процесса. | Поломки на предприятии, отключение электроэнергии, водоснабжения. |
| Последовательность действий выполнения бизнес-процесса. | * *Прибытие поставки на склад* * *Добавление ингредиентов в список* * *Добавление в список блюда* * *Добавление ингредиентов в блюда* * *Создание меню на определённую дату* * *Формирование списка блюд: первых и вторых в определённом количестве* |

#### IDEF0 диаграмма для базового БП.

### Анализ существующей реализации базового бизнес-процесса.

В качестве базового (автоматизируемого) процесса был выбран процесс «Обеспечения питания», который предусматривает формирование блюд из ингредиентов, меню из блюд, а также поставку необходимых ингредиентов в нужном составе. В КОМК отсутствует автоматизация данного процесса.

Существующее средства автоматизации бизнес-процесса «Формирование меню» реализована на базе MS Excel 2008. Структура, используемой базы данных системы, не нормализирована и не структурирована и хранит в себе избыточные данные, что влечет к быстрому увеличению места, необходимого для хранения базы данных, на жестком диске сервера.

При добавлении продуктов питания или формировании нового меню система не учитывает зависимые атрибуты сущностей, что приводит к неудобству ввода данных, вводит дополнительные трудозатраты по отслеживанию связей.

Требуется спроектировать базу данных таким образом, чтобы ее структура соответствовала общей структуре в ЕИС «КОМК» и в дальнейшем могла быть интегрирована с существующей базой данных, находящейся в работе.

## Формулировка требований к системе

Определение деятельностей, требующих автоматизации, кандидатов в требования.

#### Состав бизнес-процессов предприятия.

Состав бизнес-процессов приведен на рисунке 1.10.

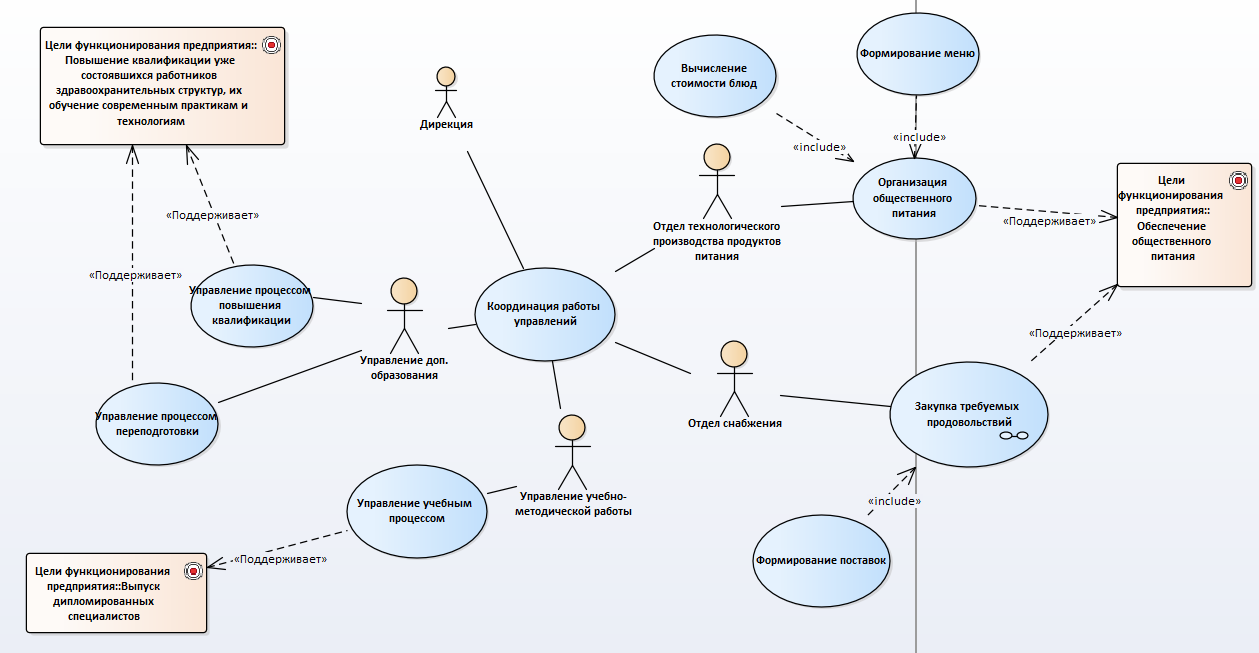


Рисунок 1.7. Состав бизнес-процессов

* Дирекция - осуществляет координацию работы управлений;
* Отдел технологического производства продуктов питания - разработка основных направлений развития технологического производства продуктов питания; Обеспечение качества производства продуктов питания;
* Отдел разработки и сопровождения программного обеспечения - осуществляет информационное обеспечение;
* Отдел технологического производства продуктов питания – осуществляет организацию общественного питания.
  + Автоматизации процесса учёта деятельности на комбинате питания.
* Отдел управление доп образованием осуществляет управление процессом переподготовки и процессом повышения квалификации
* Отдел управление учебно-методической работой осуществляет управление учебным процессом

#### Содержание бизнес-процессов.

Для подробного анализа выбран следующий процесс: “Вычисление стоимости блюд”.

Содержание бизнес-процесса “Вычисление стоимости блюд” приведено на рисунке 1.8.



Рисунок 1.8. Вычисление стоимости блюд

Работы, выполняемые документоведом:

* Выбор блюда – сотрудник выбирает блюдо, стоимость которого необходимо рассчитать;
* Поиск компонентов блюда – сотрудник находит информацию о необходимых для приготовления блюда компонентах;
* Получение цен компонентов;
* Расчет стоимости блюда – сотрудник вычисляет стоимость блюда суммируя стоимость всех компонентов;

Описание объектов:

* Блюда – является входным объектом выбора блюда; выходным объектом для расчета стоимости блюда;
* Состав блюд – является входным объектом для составления списка работ;
* Ингредиенты – является входным объектом для получения цен компонентов.

#### Классы-объекты предметной области.

Описание классов-объектов предметной области приведено на рисунке 1.9.

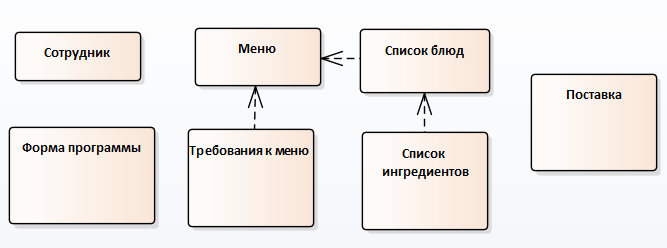


Рисунок 1.9. Описание классов-объектов предметной области

Выявленные классы:

* Меню – информационный объект, представляет собой меню на каждый день;
* Требования к меню– информационный объект, содержит данные о необходимом количестве типов блюд в меню;
* Список блюд – информационный объект, представляет данные о блюдах;
* Список ингредиентов – информационный объект, представляет данные о ингредиентах;
* Сотрудник – информационный объект, представляет собой данные о сотруднике;
* Список сотрудников - информационный объект, представляет собой список всех исполнителе;
* Поставка – информационный объект, представляет собой данные о поставках;
* Форма программы – информационный объект, представляет собой интерфейс программы.

#### Автоматизируемые элементы.

Предполагаемые автоматизируемые элементы приведены на рисунке 1.10.

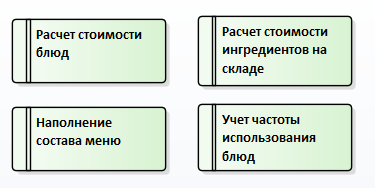


Рисунок 1.10. Автоматизируемые элементы бизнес-процессов

Расчет стоимости блюд соответствует функции «Формирование блюд». Автоматизация данного процесса позволит сократить время формирования блюд и уменьшит вероятность ошибок.

Расчет стоимости ингредиентов на складе соответствует функции «Учет поставок». Автоматизация данного процесса позволит быстро вычислить цену ингредиентов на складе.

Наполнение состава меню соответствует функции «Формирование меню». Автоматизация данного процесса позволит значительно ускорить создание меню на каждый день.

Учет частоты использования блюд соответствует функции «Формирование блюд». Автоматизация данного процесса позволит исключить однообразие меню.

Определение состава функциональных требований к системе.

Соответствие функциональных требований автоматизируемым бизнес-решениям отобразим в виде матрицы трассировки (таблица 1).

Таблица 1.6. Требования

|  |  |  |
| --- | --- | --- |
| **№** | **Бизнес-процесс** | **Требование** |
| 1 | Расчет стоимости блюд | Обеспечение возможности вычисления стоимости блюд из стоимости компонентов. |
| 2 | Расчет стоимости ингредиентов на складе | Обеспечение возможности изменения стоимости блюд после поставки. |
| 3 | Наполнение состава меню | Обеспечение возможности создания требований к меню;  Обеспечение формирования типов блюд; |
| 4 | Учет частоты использования блюд | Обеспечение возможности вычисления частоты использования блюд. |

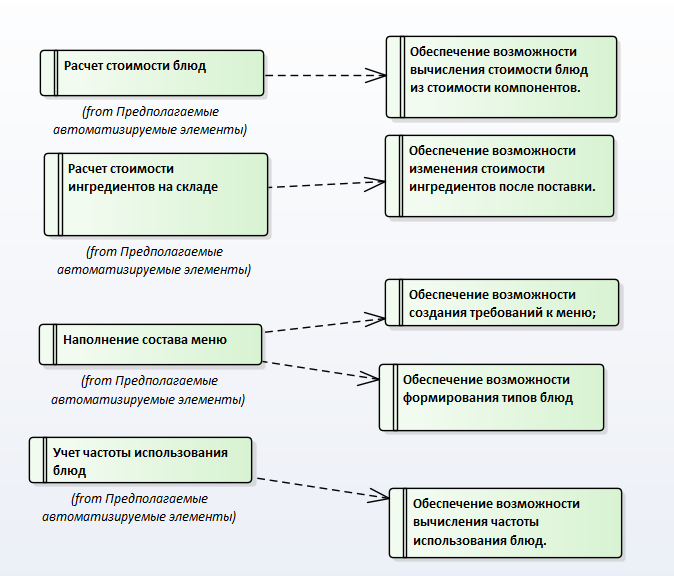
Диаграмма соответствия функциональных требований автоматизируемым бизнес-решениям приведена на рисунке 1.11.

Рисунок 1.11. Диаграмма соответствия функциональных требований автоматизируемым бизнес-решениям

#### Состав подсистем.

Для отображения выявленных подсистем построим матрицу трассировки «подсистема – функциональные требования» (таблица 1.7).

Таблица 1.7. Подсистемы

|  |  |
| --- | --- |
| **Подсистема** | **Требование** |
| Подсистема работы с меню | Обеспечение возможности создания требований к меню; |
| Обеспечение возможности вычисления частоты использования блюд. |
| Подсистема работы с блюдами | Обеспечение возможности вычисления стоимости блюд из стоимости компонентов. |
| Обеспечение формирования типов блюд; |
| Подсистема работы с поставками | Обеспечение возможности изменения стоимости блюд после поставки. |

Диаграмма выявленных подсистем приведена на рисунке 1.12.

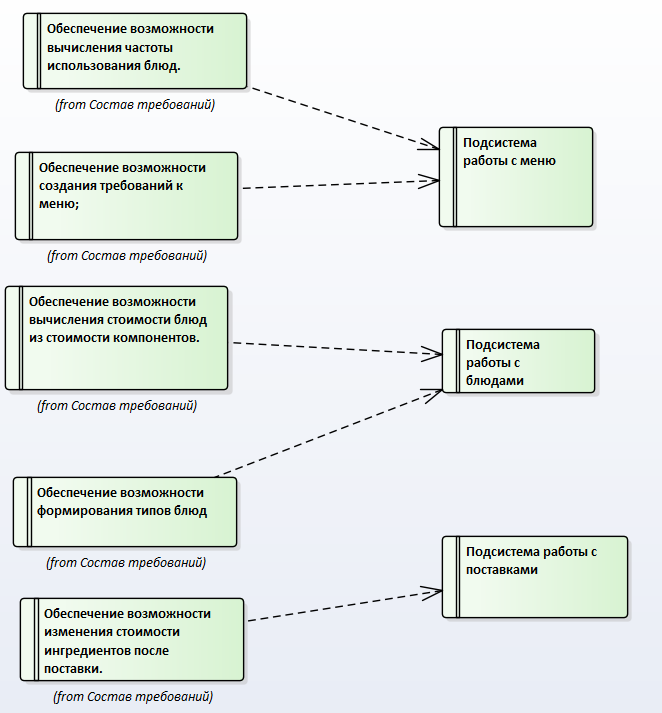


Рисунок 1.12. Диаграмма выявленных подсистем

#### Варианты использования.

Диаграмма вариантов использования, отражающая функциональные требования к системе и сценарии, их реализующие, приведены на рисунке 1.13.

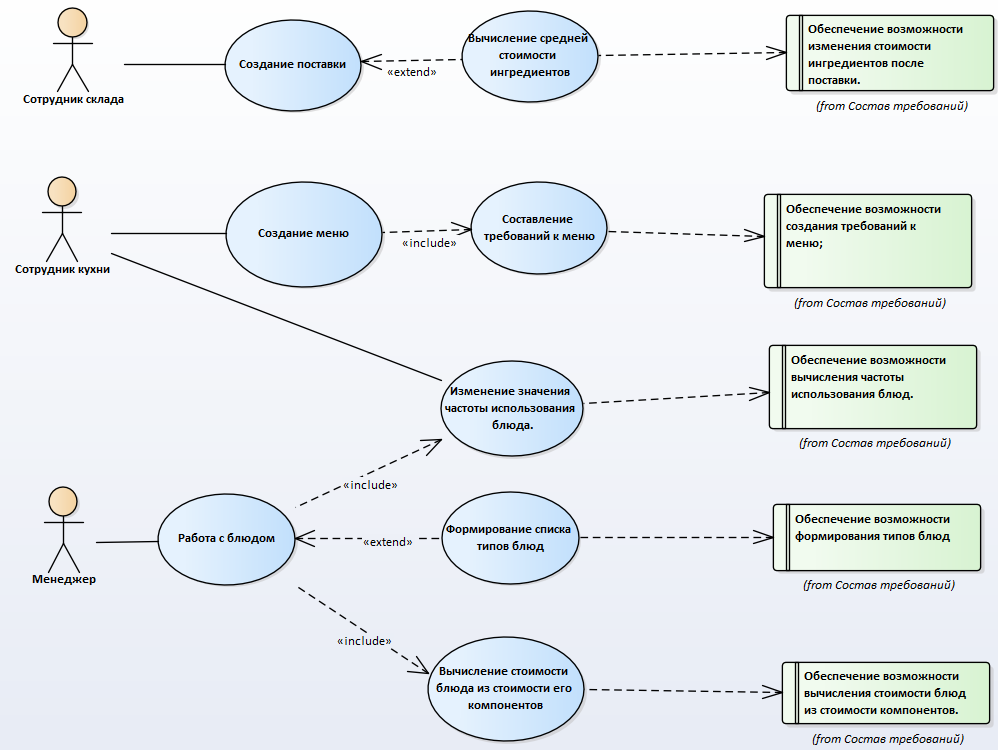


Рисунок 1.13. Диаграмма вариантов использования

Функции, выполняемые сотрудником склада:

* Создание поставки, расширяемая вычислением средней стоимости ингредиентов.

Функции, выполняемые сотрудником кухни:

* Создание меню, включающая составление требований к меню.
* Изменение значения частоты использования блюда.

Функции, выполняемые менеджером:

* Работа с блюдом, которая включает в себя: изменение значения частоты использования блюда и вычисление стоимости блюда из стоимости его компонентов. Расширяется формированием списка типов блюд.

### Разработка сценариев, реализующих требования к системе.

#### Содержание сценариев.

Проведем анализ следующих сценариев: «Создание поставки», «Создание меню», «Вычисление стоимости блюда из стоимости его компонентов», «Авторизация пользователя».

Сценарии выполнения остальных ВИ рассмотрены не будут, т.к. они имеют косвенное отношение к работе системы.

Диаграмма деятельности сценария «Создание поставки» приведена на рисунке 1.14.

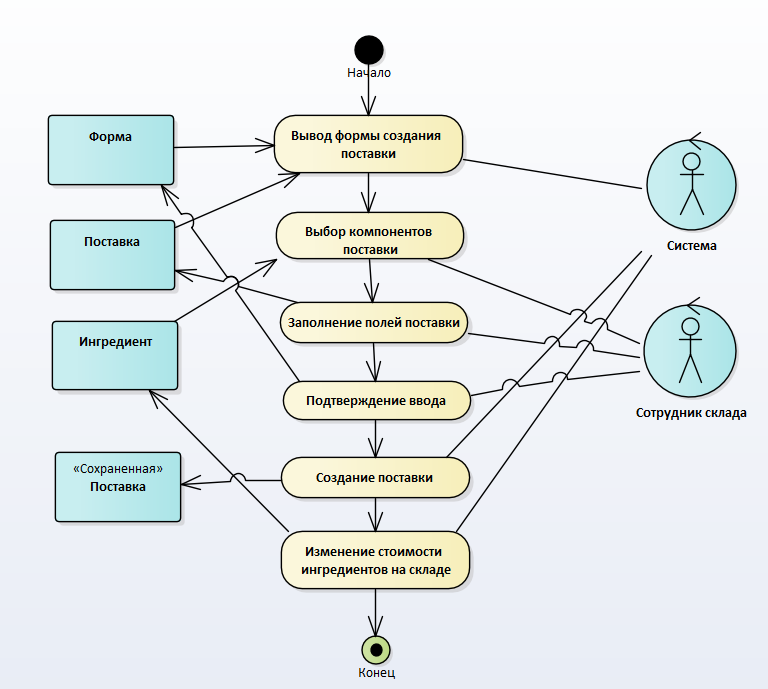


Рисунок 1.14. Создание поставки

В результате работы процесса выявлены следующие объекты:

* Граничные объекты:
* Форма – содержит формы «Форма создания поставки». Позволяет создавать, изменять, удалять, искать и выгружать документ в Excel.
* Сущности:
* Поставка – содержит шаблон для создания поставки.
* Сохраненная поставка – содержит описание поставки с датой, включаемыми компонентами, их количествами и стоимостями.

Сценарии выполняются сотрудником склада и системой:

* Вывод формы заявки, открывается форма создания заявки с полем текста заявки и значениями текущей даты и описания заявителя (выполняется системой);
* Выбор компонентов поставки (выполняется сотрудником склада);
* Заполнение полей поставки (выполняется сотрудником склада);
* Подтверждение ввода (выполняется сотрудником склада);
* Создание поставки в БД (выполняется системой);
* Изменение стоимости ингредиентов на складе (выполняется системой);

Диаграмма деятельности сценария «Создание меню» приведена на рисунке 1.15.

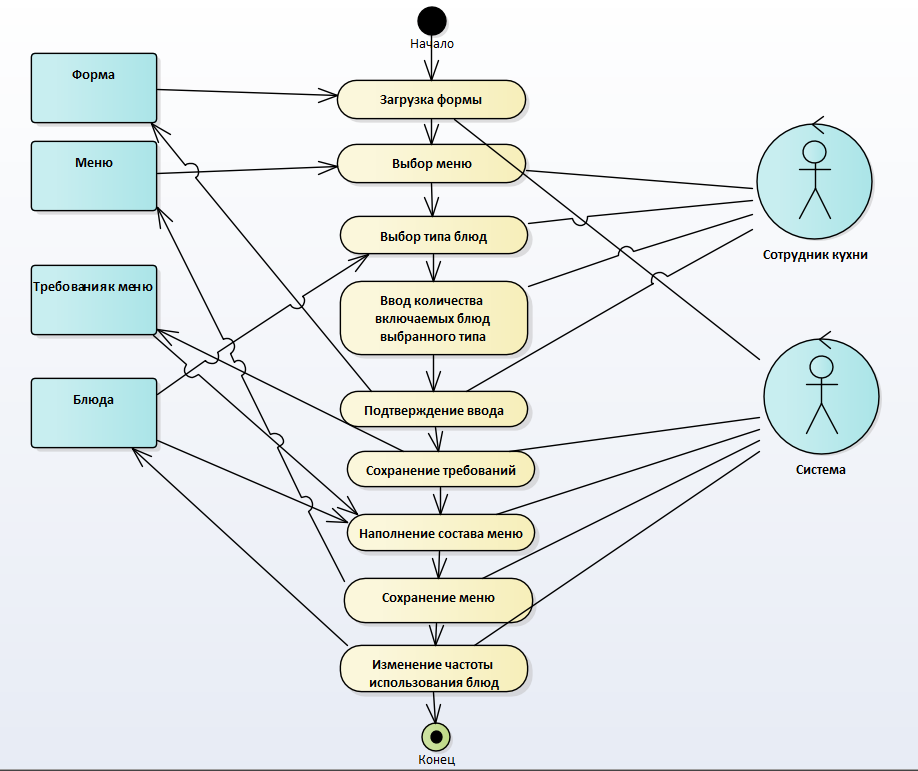


Рисунок 1.15. Создание меню

В результате работы процесса выявлены следующие объекты:

* Граничные объекты:
* Форма – содержит формы «Добавить требования», «Создать меню». Позволяет генерировать меню.
* Сущности:
* Меню – содержит данные о меню;
* Требования к меню – содержит данные о требованиях к составу конкретного меню;
* Блюда – содержит данные о блюдах.

В выполнении сценариев участвуют система и сотрудник кухни:

* Выводится форма создания меню;
* Сотрудник выбирает меню;
* Сотрудник выбирает включаемые типы блюд;
* Сотрудник вводит количество блюд;
* Сотрудник подтверждает введенные требования;
* Система сохраняет требования;
* Система наполняет состав меню:
  + Получает требования к меню;
  + Выбирает соответствующие блюда;
* Система сохраняет меню;
* Система изменяет частоту блюд.

Диаграмма деятельности сценария «Вычисление стоимости блюда из стоимости его компонентов» приведена на рисунке 1.16.

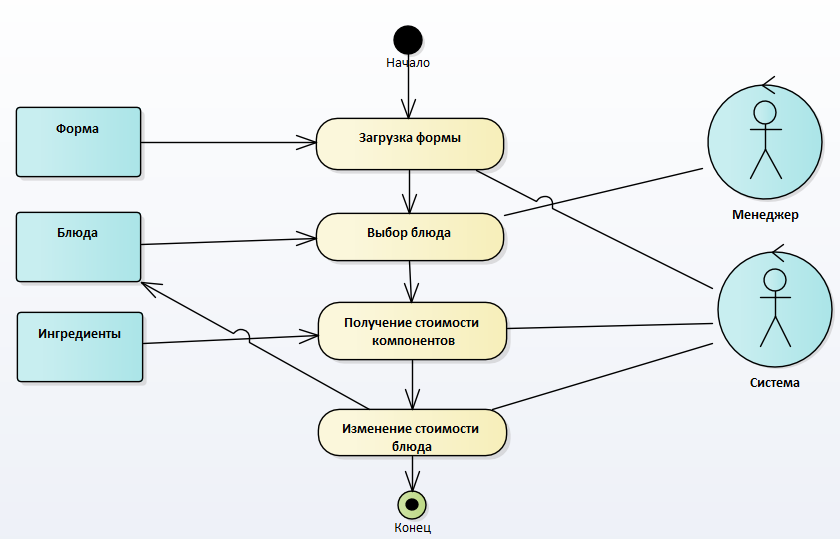


Рисунок 1.16. Вычисление стоимости блюда из стоимости его компонентов

В результате работы процесса выявлены следующие объекты:

* Граничные объекты:
* Форма – содержит формы «Вычислить стоимость блюда». Позволяет рассчитать стоимость блюда.
* Сущности:
* Блюда - содержит данные о блюдах: их стоимость, название и состав.
* Ингредиенты - содержит данные о стоимости ингредиентов.

В выполнении сценария участвуют система и исполнитель:

* Загрузка формы (выполняется системой);
* Выбор блюда (выполняется исполнителем);
* Получение стоимости компонентов (выполняется системой);
* Изменение стоимости блюда (выполняется системой).

Диаграмма деятельности сценария «Авторизация пользователя» приведена на рисунке 1.17.

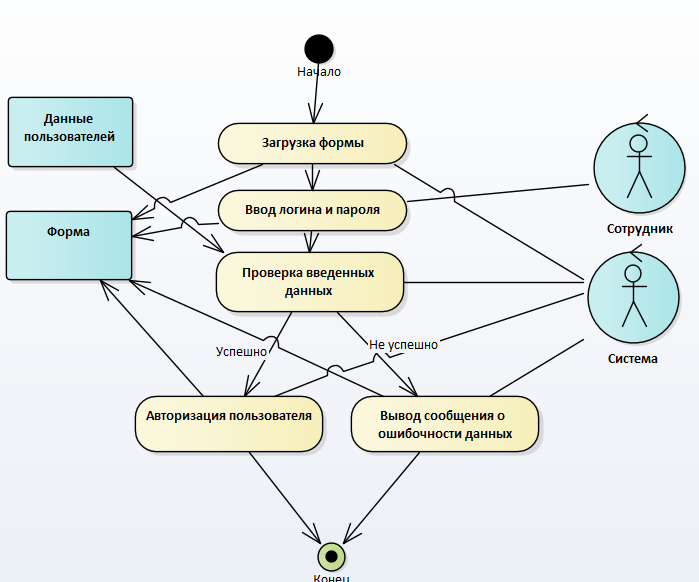


Рисунок 1.17. Авторизация пользователя

В результате работы процесса выявлены следующие объекты:

* Граничные объекты:
* Форма – содержит форму «Авторизация пользователя». Позволяет пользователю авторизоваться за свой аккаунт.
* Сущности:
* Данные пользователя - содержит данные о пользователях зарегистрированных в системе.

В выполнении сценария участвуют система и сотрудник:

* Вывод формы авторизации (выполняется системой);
* Ввод данных (выполняется сотрудником);
* Проверка данных (выполняется системой):
  + Если проверка пройдена: Авторизация пользователя в системе (выполняется системой);
  + Если проверка провалена: Вывод сообщения об ошибочности введенных данных (выполняется системой).

Определение требований к интерфейсу.

Требования к интерфейсу:

* Возможность создания поставок;
* Возможность создания меню;
* Возможность составления требований к меню;
* Возможность создания типов блюд;
* Возможность изменения стоимости блюд;
* Возможность изменения цены ингредиентов;

Прототип пользовательского интерфейса в виде диаграммы экранных форм приведен на рисунке 1.18.

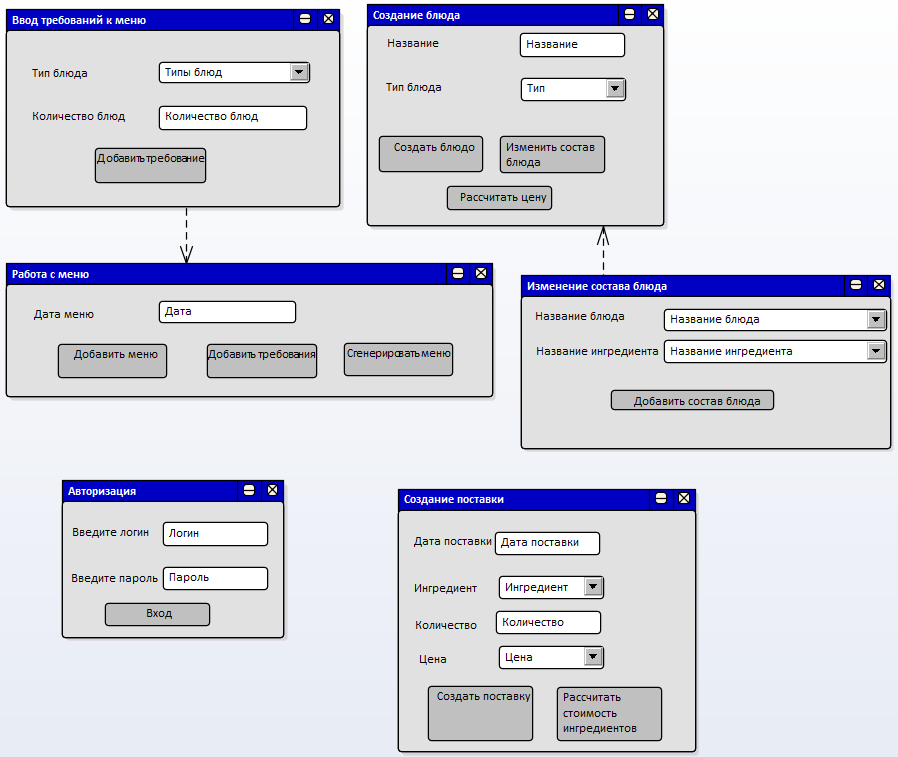
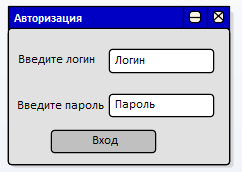


Рисунок 1.18. Интерфейс подсистемы «Учет деятельности на комбинате питания»

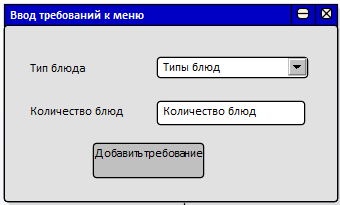
Состав форм:



**Рисунок 1.19 - Форма “Авторизация”**

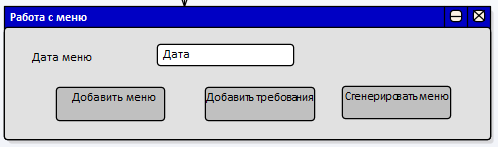
Форма “Авторизация”, предназначена для авторизации пользователей.

Содержит два поля для ввода логина (в конкретном случае использован адрес электронной почты пользователя) и пароля, а также кнопку подтверждения ввода.



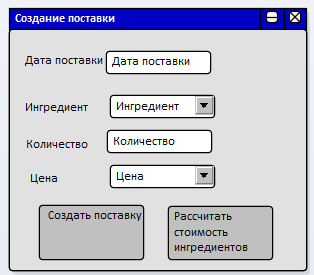
**Рисунок 1.20 - Форма “Ввод требований к меню”**

Форма “Ввод требований к меню”, предназначена для создания требований к меню. На форме необходимо выбрать тип блюда и количество блюд этого типа.



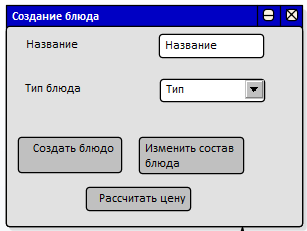
**Рисунок 1.21 - Форма “Работа с меню”**

Форма “Работа с меню”, предназначена для создания меню. На форме необходимо ввести дату меню, затем перейти на форму «Ввод требований к меню», после сгенерировать меню.



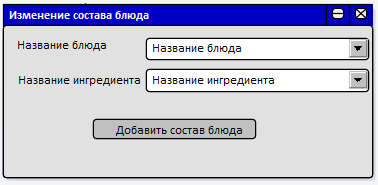
**Рисунок 1.22 - Форма “Создание поставки”**

Форма “Создание поставки”, предназначена для создания поставки. На форме требуется ввести дату поставки, выбрать ингредиент, его количество в поставке и цену.



**Рисунок 1.23 - Форма “Создание блюда”**

Форма “Создание блюда”, предназначена для создания блюда. На форме требуется ввести название блюда, выбрать тип блюда и перейти на форму «Изменение состава блюда».



**Рисунок 1.24 - Форма “Изменение состава блюда”**

Форма “Изменение состава блюда”, предназначена для формирования блюда. На форме требуется выбрать блюдо и его компонент.

Итоговый состав форм может отличаться от указанного, но заявленная функциональность при этом не должна быть потеряна.

Окончательная формулировка требований к системе.

#### Функциональные требования к системе.

1. Обеспечение возможности создания поставки в фиксированной форме (кр. «Создание поставки»).

* Описание: В программе должна быть реализована возможность создания поставки пользователем. При этом должна быть обеспечена возможность задания пользователем состава поставки. Приложение при создании поставки должно автоматически добавлять к ней идентификатор поставки и пересчитывать стоимость и количество ингредиента на складе;
* Статус требования: принято;
* Тестовый пример: Создать поставку с картошкой от имени сотрудника склада Сергея Складовского.
  + Исходные данные:
    - Сергей Складовский авторизован в системе;
    - Дата поставки – 20.06.2017 09:00:00;
    - Ингредиент – «Картошка»;
    - Количество – 40кг.;
    - Цена – 30 рублей.
  + Результаты:
    - В приложении создана поставка со следующими параметрами: Дата поставки – 20.06.2017 09:00:00, Ингредиент – «Картошка», Количество – 40кг., Цена – 30 рублей.
    - В БД создана соответствующая запись поставки.
    - В БД изменена соответствующая запись ингредиента.

1. Обеспечение возможности создания требований к меню (кр. «Создание требований»);
   * Описание: В программе должна быть реализована возможность создания требований к меню. При этом должна быть обеспечена возможность выбора меню и типа блюд, к которому относятся требования, количества блюд данного типа. Приложение при создании требований должно автоматически добавлять к ней идентификатор меню и типа блюд;
   * Статус требования: принято;
   * Тестовый пример: Создать требование к меню о включение двух первых блюд от имени сотрудника кухни Семена Столового.
     + Исходные данные:
       - Семен Столовый авторизован в системе;
       - Дата меню - 20.06.2017;
       - Тип блюда – Первое;
       - Количество блюд – 2;
     + Результаты:
       - В приложении создано требование со следующими параметрами: Дата меню - 20.06.2017, Тип блюда – Первое, Количество блюд – 2.
       - В БД создана соответствующая запись требований.
2. Обеспечение возможности создания меню (кр. «Создание меню»);
   * Описание: В программе должна быть реализована возможность создания меню и автоматического наполнения его состава на основе требований;
   * Статус требования: принято;
   * Тестовый пример: Создать меню на 20.06.2017 от имени сотрудника кухни Семена Столового.
     + Исходные данные:
       - Семен Столовый авторизован в системе;
       - Дата меню - 20.06.2017;
       - Требования к меню – 2 первых блюда;
     + Результаты:
       - В приложении создано меню с параметрами: Дата меню- 20.06.2017, Блюдо – Борщ, Блюдо – Солянка.
       - В БД создана соответствующая запись меню.
       - В БД изменены соответствующие записи блюд
3. Обеспечение возможности создания ингредиентов (кр. «Создание блюда»);
   * Описание: В программе должна быть реализована возможность создания ингредиента. При создании ингредиента приложение должно автоматически заполнять цену и количество ингредиента;
   * Статус требования: принято;
   * Тестовый пример: От имени менеджера Уолтера Управленцева создать ингредиент картошка.
     + Исходные данные:
       - Уолтер Управленцев авторизован в системе;
       - Наименование ингредиента – «Картошка»;
     + Результаты:
       - В приложении создан ингредиент с параметрами: Название – «Картошка», Цена – 0, Количество – 0.
       - В БД создана соответствующая запись ингредиентов.
4. Обеспечение возможности формирования блюд (кр. «Создание блюда»);
   * Описание: В программе должна быть реализована возможность создания состава блюда. При создании блюда приложение должно автоматически вычислять его стоимость;
   * Статус требования: принято;
   * Тестовый пример: Создать омлет из яиц и соли от имени менеджера Уолтера Управленцева.
     + Исходные данные:
       - Уолтер Управленцев авторизован в системе;
       - Блюдо – Омлет;
       - Тип блюда – Второе;
       - Ингредиент 1 – Яйцо;
       - Количество ингредиента 1 – 2;
       - Ингредиент 2 – Соль;
       - Количество ингредиента 2 – 5 грамм;
     + Результаты:
       - В приложении создано блюдо с параметрами: Наименование – Омлет, Тип блюда – Второе, Ингредиент 1 – Яйцо, Количество ингредиента 1 – 2, Ингредиент 2 – Соль, Количество ингредиента 2 – 5 грамм, Цена – 25 рублей. Частота использования – 0.
       - В БД создана соответствующая запись блюда.
5. Обеспечение возможности разграничения прав пользователей (кр. «Разграничение прав пользователей»);
   * Описание: В программе должна быть реализована возможность разграничения прав пользователей. Пользователь не имеет доступа к системе. Сотрудник склада может: создавать поставки, изменять стоимость ингредиентов, просматривать созданные поставки. Сотрудник кухни может: Создавать меню, создавать требования к меню, генерировать меню, просматривать созданные меню. Менеджер может: создавать ингредиенты, создавать блюда, формировать блюда, создавать типы блюд, вычислять стоимость блюда. Администратор может: изменять права пользователей;
   * Статус требования: принято;
   * Тестовый пример №1: Менеджер Уолтер Управленцев авторизуется в системе.
     + Исходные данные:
       - Уолтер Управленцев не авторизован в системе;
     + Результаты:
       - Авторизованному пользователю доступны вкладки: «Блюда», «Ингредиенты», «Типы блюд».
   * Тестовый пример №2: Сотрудник кухни Семен Столовый авторизуется в системе.
     + Исходные данные:
       - Семен Столовый не авторизован в системе;
     + Результаты:
       - Авторизованному пользователю доступны вкладки: «Меню», «Требования к меню».
6. Обеспечение формирования типов блюд (кр. «Формирование типов блюд»).
   * Описание: В программе должна быть реализована возможность форматирования списка типов блюд. Менеджер должен иметь возможность конфигурировать список типов блюд;
   * Статус требования: принято;
   * Тестовый пример №1: Создать новый тип блюд от имени менеджера Уолтера Управленцева.
     + Исходные данные:
       - Уолтер Управленцев авторизован в системе;
       - Наименование типа блюд – Второе.
     + Результаты:
       - В приложении создан тип блюд с параметрами: Наименование – «Второе»;
       - В БД создана соответствующая запись типов блюд.

#### Нефункциональные требования к системе.

* Производительность системы – обеспечить достаточно быстрый поиск и вывод нужной информации для удобства использования;
  + Тестовый пример: Процесс назначение исполнителя на работу не должно занимать больше 7 секунд.
* Безопасность данных – разделения прав доступа к информации в БД;
  + Тестовый пример: Пользователь не должен иметь доступа к заявкам других пользователей;
* Удобство использования – дружественный, интуитивно понятный пользовательский интерфейс;
  + Тестовый пример: Для того чтобы разобраться с интерфейсом системы, сотрудник тех. поддержки должен потратить не более 10 минут.
* Стоимость разработки не должна превышать 15 000 руб;

#### Проектные требования к системе.

* Ограничение ОС Windows 7;
* Пользователь должен иметь навыки работы с компьютером;
* Администратор должен иметь необходимую квалификацию;
* Сервер SQL server;
* Средства разработки на C#.

#### Требования к процессу хранения данных.

Данные приложения, а именно информацию о пользователях, сотрудниках тех. поддержки и исполнителях, заявки пользователей на обслуживание, работы по данным заявкам, а также константные значения, такие как типы работ, статусы заявок/работ и т.п. требуется хранить на удаленном сервере. Хранение данных на сервере должно организовываться с использованием СУБД MS SQL Server.

Предполагается, что приложение будет иметь клиент-серверную архитектуру (рисунок 1.25).



**Рисунок 1.25 Клиент-серверная архитектура**

Сотрудник предприятия через приложение на рабочем месте формирует заявку на обслуживание и сохраняет ее, сформированная заявка сохраняется сервером приложение на сервере данных. Отравленная заявка обрабатывается работником отдела тех. обеспечения, формируется список работ соответствующий заявке. По мере выполнения работ добавляются новые записи в БД об изменении статуса работ и заявок.

#### Определение приоритетов требований.

Табл. 1.13.

|  |  |
| --- | --- |
| Требование | Приоритет |
| Функциональные | |
| Создание поставки | Реализуется |
| Создание требований к меню | Реализуется |
| Создание меню | Реализуется |
| Создание ингредиентов | Реализуется |
| Разграничение прав пользователей | Реализуется |
| Формирование блюд | В проекте |
| Формирование типов блюд | В проекте |
| Нефункциональные | |
| Производительность системы | Ограничение |
| Безопасность данных | Ограничение |
| Удобство использования | Ограничение |
| Стоимость разработки | Ограничение |

# Анализ и проектирование

## Определение классов анализа

Выявлены классы:

Классы граничных объектов:

* Класс "Форма" – определён на основе объекта «Форма» сценариев «Создание поставки», «Создание меню», «Создание блюда».
* Класс "Форма создания поставки" – определён на основе объекта «Форма» сценария «Создание поставки», является подокном «Формы».
* Класс "Форма создания меню" – определён на основе объекта «Форма» сценария «Работа с меню», является подокном «Формы».
* Класс "Форма создания блюда" – определён на основе объекта «Форма» сценария «Работа с блюдом», является подокном «Формы».
* Класс "Форма создания состава блюда" – определён на основе объекта «Форма» сценария «Создание состава блюда», является подокном «Формы создания блюда».
* Класс "Форма требований" – определён на основе объекта «Форма» сценария «Создание блюда», является подокном «Формы создания меню».
* Класс "Форма авторизации" - определен на основе объекта «Форма» сценария «Авторизация пользователя», является подокном «Формы».

Классы сущностей:

* Класс "Поставка" – определён на основе объекта «Поставка» сценариев «Создание поставки». Атрибуты:
  + Идентификатор, тип int - искусственный атрибут, хранит в себе идентификатор объекта;
  + Дата поставки, тип date - содержит в себе дату прихода поставки;
  + Ингредиент, тип int- искусственный атрибут, хранит в себе идентификатор ингредиента;
  + Количество, тип float – содержит в себе количество пришедшего ингредиента;
  + Цена, тип float – содержит в себе стоимость пришедшего ингредиента;
* Класс "Ингредиент" – определён на основе объекта «Ингредиент» сценария «Создание блюда». Атрибуты:
  + Идентификатор, тип int - искусственный атрибут, хранит в себе идентификатор объекта;
  + Название, тип char- содержит в себе название ингредиента;
  + Цена, тип float – содержит в себе цену ингредиента на складе;
  + Количество, тип float – содержит в себе количество ингредиента на складе;
* Класс «Данные пользователей» - определен на основе объекта «Данные пользователей» сценария «Авторизация пользователя». Атрибуты:
  + Идентификатор, тип int - искусственный атрибут, хранит в себе идентификатор пользователя;
  + Логин, тип char - содержит в себе логин пользователя;
  + Пароль, тип char - содержит пароль пользователя.
* Класс «Блюдо» - определен на основе объекта «Блюдо» сценария «Создание блюда». Атрибуты:
  + Идентификатор, тип int - искусственный атрибут, хранит в себе идентификатор объекта;
  + Наименование, тип char - содержит в себе наименование блюда.
  + Тип, тип char – хранит в себе название типа блюд;
  + Цена, тип float – содержит в себе цену блюда;
  + Частота, тип int – содержит в себе количество включений блюда в меню;
  + Ингредиент, тип int – искусственный атрибут, хранит в себе идентификатор ингредиента.
* Класс «Меню» - определен на основе объекта «Меню» сценария «Меню». Атрибуты:
  + Идентификатор, тип int - искусственный атрибут, хранит в себе идентификатор объекта;
  + Дата, тип date - содержит дату использования меню;
  + Блюдо, тип int – искусственный атрибут, хранит в себе идентификатор блюда.
* Класс «Требования к меню» - определен на основе объекта «Требования к меню» сценария «Создание меню». Атрибуты:
  + Идентификатор, тип int - искусственный атрибут, хранит в себе идентификатор объекта;
  + Меню, тип int- искусственный атрибут, хранит в себе идентификатор меню;
  + Тип блюда, тип int - искусственный атрибут, хранит в себе идентификатор меню;
  + Количество, тип int – хранит требуемое количество блюд в меню;

Определение методов объектов

Диаграммы последовательности сценариев: «Создание поставки», «Создание меню», «Создание блюда».

Диаграмма последовательности сценария «Создание поставки» приведена на рисунке 2.1.

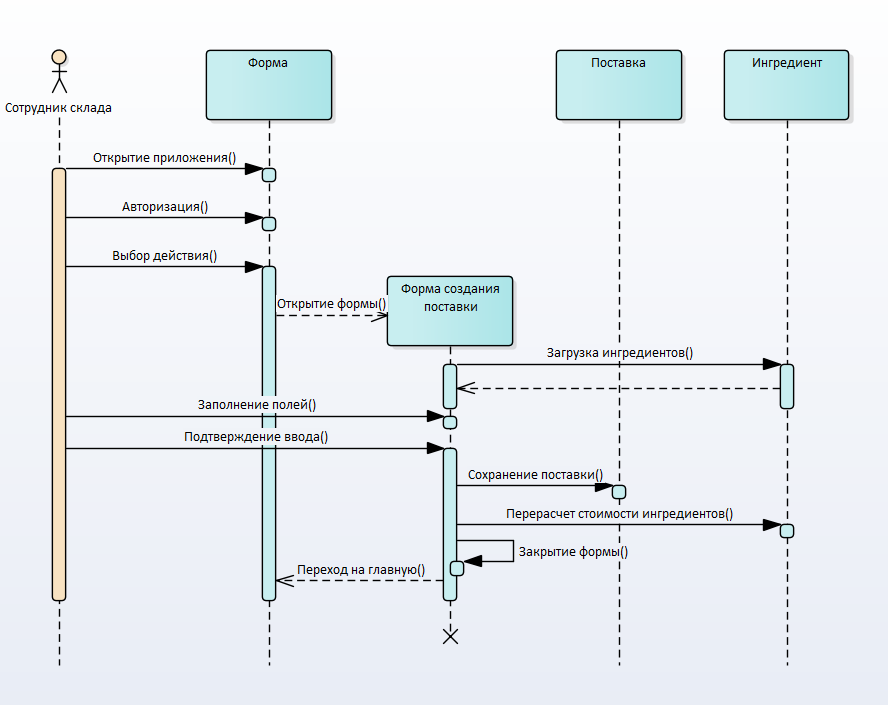


Рисунок 2.1. Создание поставки

В ходе построения объекты диаграммы выявлены следующие объекты:

1. Сотрудник склада
2. Форма
3. Форма создания поставки
4. Поставка
5. Ингредиент

Выявлены следующие сообщения:

Табл., 2.1.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Вызывающий объект | Сообщение | Получатель сообщения | Примечание |
| 1 | Сотрудник склада | Открыть приложение | Форма |  |
| 2 | Сотрудник склада | Авторизация | Форма |  |
| 3 | Сотрудник склада | Выбор действия | Форма |  |
| 4 | Форма | Открыть форму создания поставки | Форма создания поставки |  |
| 5 | Форма создания поставки | Загрузить ингредиенты | Ингредиент |  |
| 6 | Ингредиент | Вернуть ингредиенты | Форма создания заявки |  |
| 7 | Сотрудник склада | Заполнение полей | Форма создания поставки |  |
| 8 | Сотрудник склада | Подтверждение ввода | Форма создания поставки |  |
| 9 | Форма создания поставки | Сохранение поставки | Поставка |  |
| 10 | Форма создания поставки | Перерасчет стоимости ингредиентов | Ингредиент |  |
| 11 | Форма создания поставки | Закрытие формы | Форма создания поставки |  |
| 12 | Форма создания заявки | Перейти на главную | Форма |  |

Табл., 2.2

|  |  |
| --- | --- |
| Операторы | Список сообщений |
| Форма | Открыть форму создания заявки. |
| Форма создания поставки | Загрузка ингредиентов;  Сохранение поставки;  Перерасчет стоимости ингредиентов;  Вернуться на главную. |

Диаграмма последовательности сценария «Создание меню» приведена на рисунках 2.2.

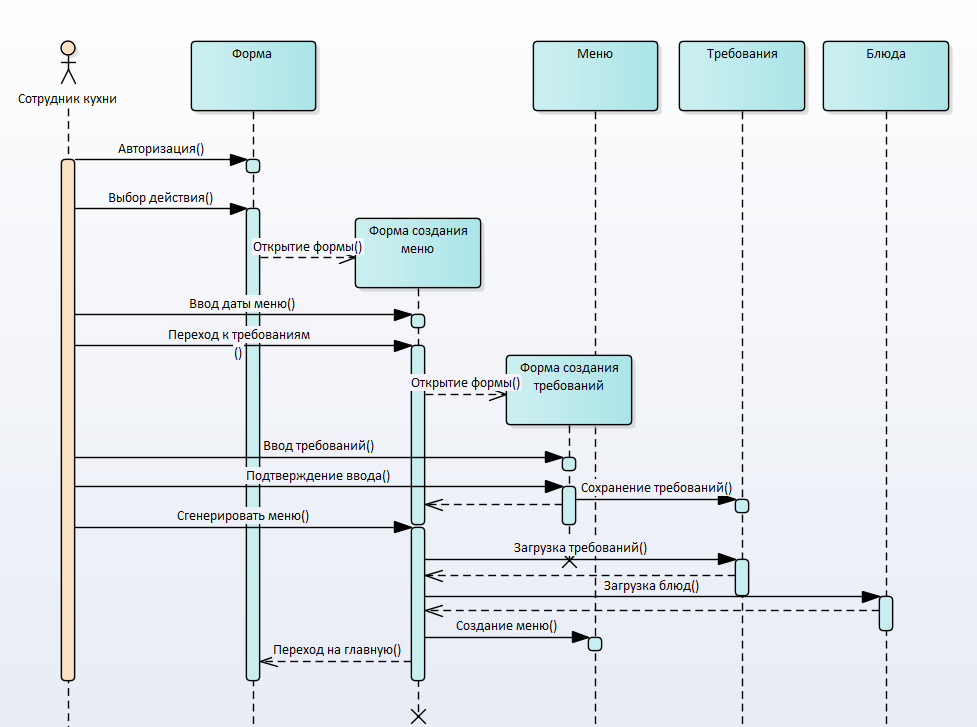


Рисунок 2.2 Создание поставки

В ходе построения объекты диаграммы выявлены следующие объекты:

1. Сотрудник кухни
2. Форма
3. Форма создания меню
4. Форма создания требований
5. Меню
6. Требования
7. Блюда

Выявлены следующие сообщения:

Таблц., 2.3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Вызывающий объект | Сообщение | Получатель сообщения | Примечание |
| 1 | Сотрудник кухни | Авторизация | Форма |  |
| 2 | Сотрудник кухни | Выбор действия | Форма |  |
| 3 | Форма | Открытие формы | Форма создания меню |  |
| 4 | Сотрудник кухни | Ввод даты меню | Форма создания меню |  |
| 5 | Сотрудник кухни | Переход к требованиям | Форма создания меню |  |
| 6 | Форма создания меню | Открытие формы | Форма создания требований |  |
| 7 | Сотрудник кухни | Ввод требований | Форма создания требований |  |
| 8 | Сотрудник кухни | Подтверждение ввода | Форма создания требований |  |
| 9 | Форма создания требований | Сохранение требований | Требования |  |
| 10 | Форма создания требований | Возврат к меню | Форма создания меню |  |
| 11 | Сотрудник кухни | Сгенерировать меню | Форма создания меню |  |
| 12 | Форма создания меню | Загрузка требований | Требования |  |
| 13 | Требования | Загрузка требований | Форма создания меню |  |
| 14 | Форма создания меню | Загрузка блюд | Блюда |  |
| 15 | Блюда | Загрузка блюд | Форма создания меню |  |
| 16 | Форма создания меню | Создание меню | Меню |  |
| 17 | Форма создания меню | Переход на главную | Форма |  |

Табл., 2.4.

|  |  |
| --- | --- |
| Операторы | Список сообщений |
| Форма | Открытие формы. |
| Форма создания меню | Открытие формы;  Загрузка требований;  Загрузка блюд;  Создание меню  Переход на главную. |
| Форма создания требований | Сохранение требований;  Вернуться на форму создания меню. |

Диаграмма последовательности сценария «Создание блюда» приведена на рисунке 2.3.

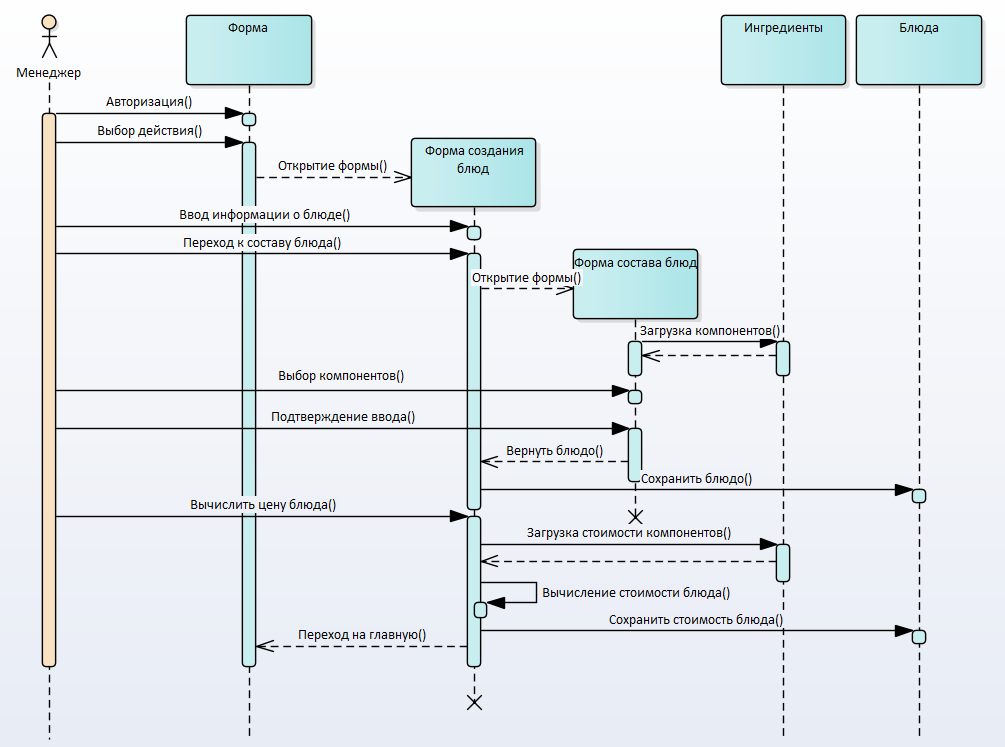


Рисунок 2.3. Создание блюда

В ходе построения объекты диаграммы выявлены следующие объекты:

1. Менеджер
2. Форма
3. Форма создания блюд
4. Форма состава блюда
5. Ингредиенты
6. Блюда

Выявлены следующие сообщения:

Табл., 2.5.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Вызывающий объект | Сообщение | Получатель сообщения | Примечание |
| 1 | Менеджер | Авторизация | Форма |  |
| 2 | Менеджер | Выбор действия | Форма |  |
| 3 | Форма | Открытие формы | Форма создания блюд |  |
| 4 | Менеджер | Ввод информации о блюде | Форма создания блюд |  |
| 5 | Менеджер | Переход к составу блюда | Форма создания блюд |  |
| 6 | Форма создания блюд | Открытие формы | Форма состава блюд |  |
| 7 | Форма состава блюд | Загрузка компонентов | Ингредиенты |  |
| 8 | Ингредиенты | Загрузка компонентов | Форма состава блюд |  |
| 9 | Менеджер | Выбор компонентов | Форма состава блюд |  |
| 10 | Менеджер | Подтверждение ввода | Форма состава блюд |  |
| 11 | Форма состава блюд | Вернуть блюдо | Форма создания блюд |  |
| 12 | Форма создания блюд | Сохранить блюдо | Блюдо |  |
| 13 | Менеджер | Вычислить цену блюда | Форма создания блюд |  |
| 14 | Форма создания блюд | Загрузка стоимости компонентов | Ингредиенты |  |
| 15 | Ингредиенты | Загрузка стоимости компонентов | Форма создания блюд |  |
| 16 | Форма создания блюд | Вычисление стоимости блюда | Форма создания блюд |  |
| 17 | Форма создания блюд | Сохранить стоимость блюда | Блюда |  |
| 18 | Форма создания блюд | Переход на главную | Форма |  |

Табл., 2.6.

|  |  |
| --- | --- |
| Операторы | Список сообщений |
| Форма | Открытие формы. |
| Форма создания блюд | Открытие формы;  Сохранить блюдо;  Загрузка стоимости компонентов;  Вычисление стоимости блюда;  Сохранить стоимость блюда;  Переход на главную. |
| Форма состава блюд | Загрузка компонентов;  Вернуть блюдо. |

Диаграмма последовательности сценария «Авторизация» приведена на рисунке 2.5.

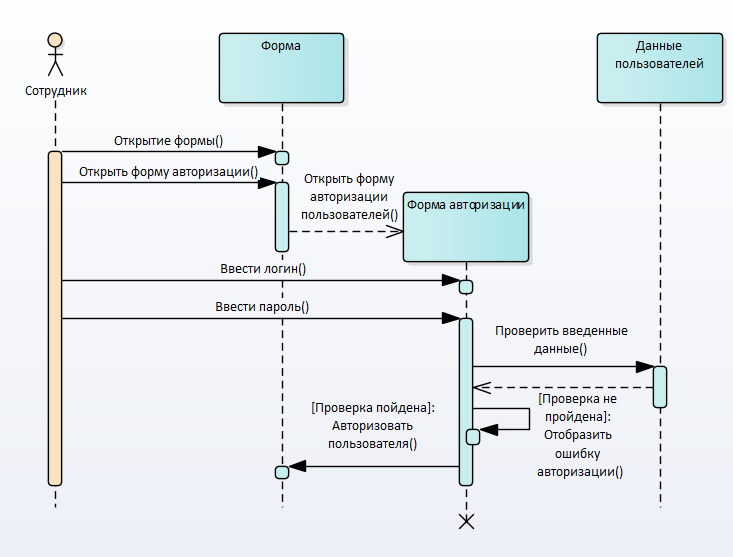


Рисунок 2.5. Авторизация

В ходе построения объекты диаграммы выявлены следующие объекты:

1. Сотрудник
2. Форма
3. Форма авторизации
4. Данные пользователя

Выявлены следующие сообщения:

Табл 2.7.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Вызывающий объект | Сообщение | Получатель сообщения | Примечание |
| 1 | Сотрудник | Открытие приложения | Форма |  |
| 2 | Сотрудник | Открыть форму авторизации | Форма |  |
| 3 | Форма | Открыть форму авторизации пользователя | Форма авторизации |  |
| 4 | Сотрудник | Ввести логин | Форма авторизации |  |
| 5 | Сотрудник | Ввести пароль | Форма авторизации |  |
| 6 | Форма авторизации | Проверить введенные данные | Данные пользователей |  |
| 7 | Данные пользователей | Вернуть результаты проверки | Форма авторизации |  |
| 8 | Форма авторизации | Отобразить ошибку авторизации | Форма авторизации | Проверка не пройдена |
| 9 | Форма авторизации | Авторизовать пользователя | Форма | Проверка пройдена |

Табл 2.8.

|  |  |
| --- | --- |
| Операторы | Список сообщений |
| Форма | Открыть форму авторизации. |
| Форма авторизации | Проверить введенные данные;  Отобразить ошибку авторизации;  Авторизовать пользователя. |

Выявленные методы классов анализа:

* Класс "Форма"
  + Открыть форму создания поставки, тип возврата void - создает граничный объект класса «Форма создания поставки»;
  + Открыть форму создания меню, тип возврата void - создает граничный объект класса «Форма создания меню»;
  + Открыть форму создания блюда, тип возврата void - создает граничный объект класса «Форма создания блюда»;
  + Открыть форму авторизации, тип возврата void - создает граничный объект класса «Форма авторизации».
* Класс "Форма создания поставки"
  + Создать поставку, тип возврата void – создает поставку на основе введенных данных;
  + Пересчет стоимости ингредиентов, тип возврата void – изменяет стоимость ингредиентов;
  + Перейти на главную, тип возврата void - осуществляет переход на главную страницу приложения.
* Класс "Форма создания меню"
  + Открыть форму создания требований, тип возврата void – создает граничный объект класса «Форма создания требований»;
  + Загрузка требований и блюд, тип возврата void – осуществляет загрузку требований и блюд из БД;
  + Создать меню, тип возврата void – сохраняет меню в БД;
  + Перейти на главную, тип возврата void - осуществляет переход на главную страницу приложения.
* Класс "Форма создания требований"
  + Сохранить требования, тип возврата Требования– создает требования на основе введенных данных;
  + Вернуться на форму создания меню, тип возврата void - осуществляет переход на страницу создания меню.
* Класс "Форма создания блюда"
  + Открыть форму состава блюда, тип возврата void – создает граничный объект класса «Форма состава блюда»;
  + Загрузка стоимости компонентов, тип возврата void – получает стоимость компонентов блюда из БД;
  + Вычисление стоимости блюд, тип возврата float – рассчитывает стоимость блюда;
  + Сохранить стоимость блюда, тип возврата void – сохраняет изменения, сделанные в блюде;
  + Перейти на главную, тип возврата void - осуществляет переход на главную страницу приложения.
* Класс "Форма состава блюда"
  + Загрузить компоненты, тип возврата void – загружает компоненты на форму;
  + Вернуть блюдо, тип возврата Блюдо - осуществляет возврат блюда на страницу создания блюда.
* Класс "Форма авторизации"
  + Проверить введенные данные, тип возврата void - запрашивает проверку логина и пароля пользователя;
  + Авторизовать пользователя, тип возврата void - авторизует пользователя в системе;
  + Отобразить ошибку авторизации, тип возврата void - отображает ошибку авторизации.

Диаграмма классов-анализа приведена на рисунке 2.7:

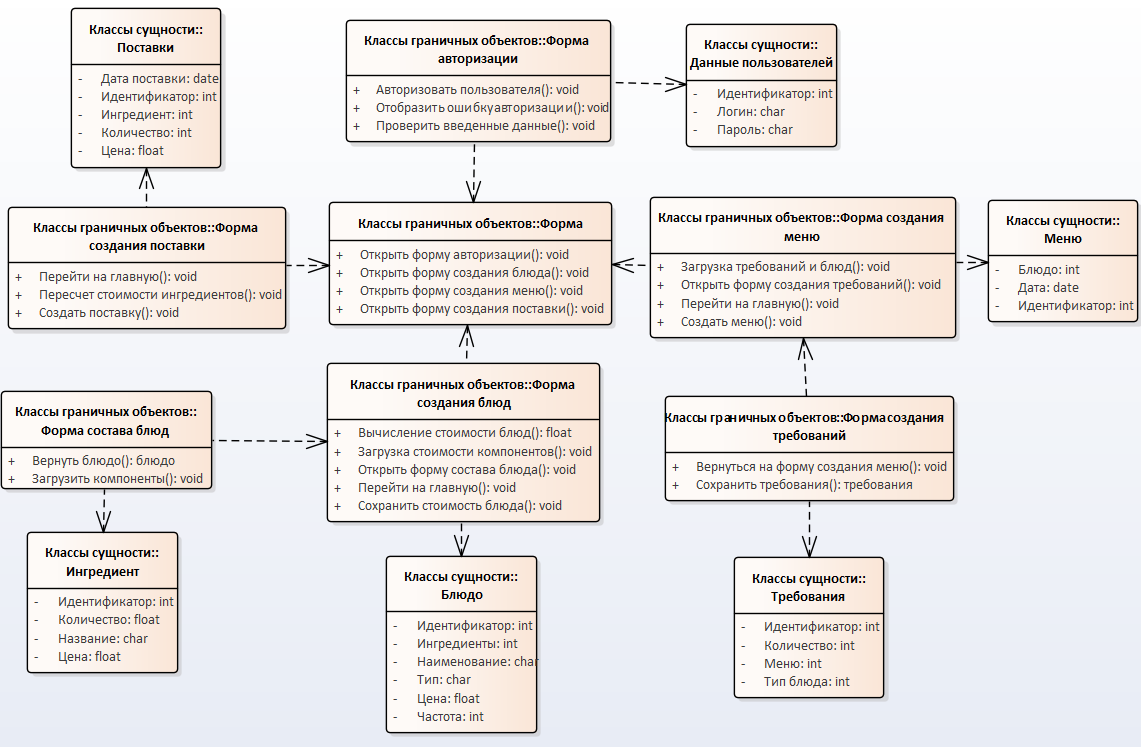


Рисунок 2.7. Диаграмма классов-анализа.

# Разработка системы централизованного хранения и обработки данных

## Формирование требований к БД

На основе требований заказчика система должна быть построена на основе СУБД поддерживающей архитектуру клиент-сервер. Приложение разрабатывается с использованием технологий ASP.NET MVC и Entity Framework.

База данных должна хранить информацию о пользователях системы, блюдах, ингредиентах, меню, поставках, а также их составах, типах блюд и требованиях к меню. База данных должна быть универсальной, т.е. оставаться работоспособной на любой машине и подстраиваться под видоизменение пользовательского интерфейса, быть способной к реорганизации и расширения при изменении предметной области.

В качестве СУБД для разработки системы выбрана Microsoft SQL Server 2012. Microsoft SQL Server — система управления реляционными базами данных (СУБД), разработанная корпорацией Microsoft. Используется для работы с базами данных размером от персональных до крупных баз данных масштаба предприятия. Широкий спектр использования Microsoft SQL Server обосновывает целесообразность выбора.

## Формирование отношений БД

На основе классов, выявленных в разделе анализа и проектирования ИС, были определены следующие сущности базы данных:



Рисунок 3.0. ER-диаграмма

На основе классов, выявленных в разделе анализа и проектирования ИС, были определены следующие сущности базы данных:

* + Сущность «Entrance» - определена на основе класса проектирования «Поставка», отражает поставки ингредиентов на склад;
  + Сущность «EntranceComposition» - связующая таблица, используется для организации связи «многие ко многим» между сущностями «Entrance» и «Ingredient».
  + Сущность «Ingredient» - справочная сущность, определена на основе класса проектирования «Ингредиент», ингредиенты, используемые в блюдах;
  + Сущность «DishComposition» - связующая таблица, используется для организации связи «многие ко многим» между сущностями «Dish» и «Ingredient»;
  + Сущность «Dish» - справочная сущность, определена на основе класса проектирования «Блюдо», хранит в себе все возможные блюда;
  + Сущность «MenuComposition» - связующая таблица, используется для организации связи «многие ко многим» между сущностями «Dish» и «Menu»;
  + Сущность «Menu» - определена на основе класса проектирования «Меню», хранит информацию о меню;
  + Сущность «MenuRequirement» - определена на основе класса проектирования «Требования к меню», хранит информацию о требованиях к меню;
  + Сущность «DishTypes» - справочная сущность, хранит в себе все возможные типы блюд.
  + Сущность «AspNetUsers» - определена на основе класса проектирования «Данные пользователей», хранит в себе данные пользователей, применяемые при авторизации;
  + Сущность «AspNetRoles» - справочная сущность, хранит в себе все возможные уровни доступа пользователей к системе.
  + Сущность «AspNetUserRoles» - связующая таблица, используется для организации связи «многие ко многим» между сущностями «AspNetUsers» и «AspNetRoles».

## Даталогическое проектирование

В связи с тем, что для разработки будет использована технология MVC, таблицы базы данных будут сформированы автоматически, на основе определенных классов анализа.

На основе состава сущностей, созданы таблицы БД:

* " Entrance " – хранит в себе поставки. Атрибуты:
  + Id, тип int – искусственный атрибут, хранит в себе идентификатор поставки;
  + Date, тип datetime – содержит дату поставки;
  + Recalculation, тип bit – показывает пересчитывались ли цены ингредиентов из этой поставки.

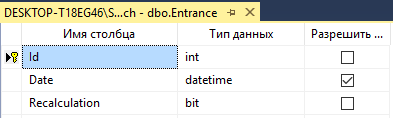


Рисунок 3.1. Таблица Entrance

* " EntranceComposition" – хранит в себе состав поставки. Атрибуты:
  + Id, тип int – искусственный атрибут, хранит в себе идентификатор состава поставки;
  + Entrance, тип int – хранит в себе идентификатор поставки, к которой относится состав;
  + Ingredient, тип int - хранит в себе идентификатор ингредиента, включенного в поставку;
  + Count, тип float – хранит в себе количество пришедшего ингредиента;
  + Cost, тип money – хранит в себе стоимость закупленного ингредиента.

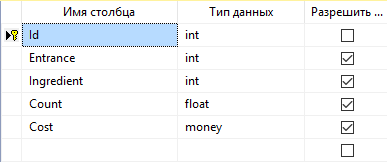


Рисунок 3.2. Таблица EntranceComposition

* " Ingredient " – хранит в себе возможные ингредиенты. Атрибуты:
  + Id, тип int – искусственный атрибут, хранит в себе идентификатор ингредиента;
  + Name, тип char - содержит наименование ингредиента;
  + Cost, тип money – хранит в себе стоимость ингредиента;
  + Count, тип float – хранит в себе количество ингредиента на складе.

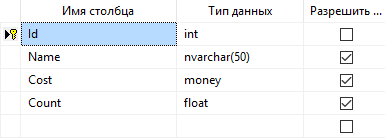


Рисунок 3.3. Таблица Ingredient

* "DishComposition" - хранит в себе состав блюд.
  + Id, тип int – искусственный атрибут, хранит в себе идентификатор состава блюд;
  + Dish, тип int – хранит в себе идентификатор блюда, к которому относится состав;
  + Ingridient, тип int - хранит в себе идентификатор ингредиента, используемого в блюде;
  + Count, тип float - хранит в себе необходимое количество ингредиента;

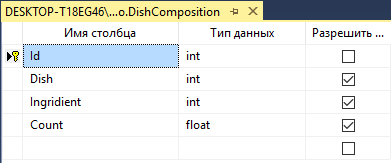


Рисунок 3.4. Таблица DishComposition

* " Dish " - хранит возможные блюда. Атрибуты:
  + Id, тип int - искусственный атрибут, хранит в себе идентификатор блюда;
  + Name, тип char – хранит в себе название блюда;
  + Type, тип int - хранит в себе идентификатор типа блюд, к которому относится блюдо;
  + Cost, тип money – хранит в себе стоимость блюда;
  + Frequency, тип int – хранит в себе частоту попадания блюда в меню;

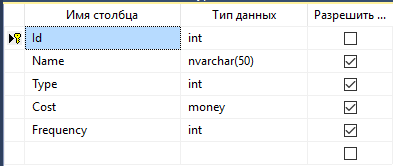


Рисунок 3.5. Таблица Dish

* MenuComposition - хранит в себе состав меню. Атрибуты:
  + Id, тип int - искусственный атрибут, хранит в себе идентификатор состава меню;
  + Menu, тип int – хранит в себе идентификатор меню, к которому относится состав;
  + DishComposition, тип int – хранит в себе идентификатор блюда, включаемого в меню;



Рисунок 3.6. Таблица MenuComposition

* AspNetUsers - автосгенерированная таблица, хранит в себе информацию о персонах, работающих в КОМК. Атрибуты:
  + Id, тип int - искусственный атрибут, хранит в себе идентификатор персоны;
  + UserName, тип nvarchar - содержит логин сотрудника (по факту Email);
  + Email, тип char - содержит Email сотрудника;
  + PasswordHash, тип nvarchar - содержит зашифрованный пароль сотрудника.
  + SecurityStamp, тип nvarchar - содержит “Соль” для пароля сотрудника.

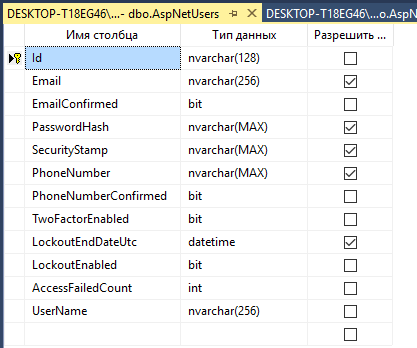


Рисунок 3.7. Таблица AspNetUsers

* AspNetRoles - хранит в себе информацию о ролях сотрудников, работающих в КОМК. Атрибуты:
  + Id, тип nvarchar
  + RoleId, тип nvarchar - искусственный атрибут, хранит в себе идентификатор роли;
  + Description, тип nvarchar - содержит наименование роли.

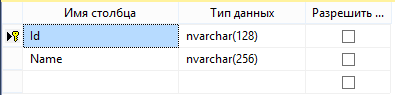


Рисунок 3.8. Таблица AspNetRoles

* AspNetUserRoles- таблица, обеспечивающая связь многие к многим между ролями и пользователями. Атрибуты:
  + RoleId, тип nvarchar- искусственный атрибут, хранит в себе идентификатор роли;
  + UserId, тип nvarchar- искусственный атрибут, хранит в себе идентификатор персоны;
  + UserRoleId, тип int - искусственный атрибут, хранит в себе идентификатор записи.

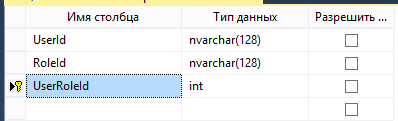


Рисунок 3.9. Таблица AspNetUserRoles

* Menu - хранит в себе меню. Атрибуты:
  + Id, тип int - искусственный атрибут, хранит в себе идентификатор меню;
  + Date, тип date – содержит дату, на которую составлено меню;

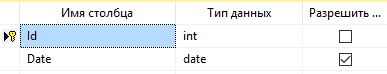


Рисунок 3.10. Таблица Menu

* "MenuRequirement" – хранит в себе требования к составу меню. Атрибуты:
  + Id, тип int - искусственный атрибут, хранит в себе идентификатор требований
  + Menu, тип int - содержит идедификатор меню, к которому относятся требования;
  + DishType, тип int - содержит идентификатор типа блюда, которое должно быть в меню;
  + Count, тип int – содержит количество блюд данного типа.

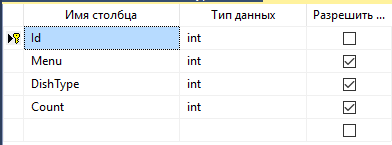


Рисунок 3.11. Таблица MenuRequirement

* "DishType" – хранит в себе типы блюд. Атрибуты:
  + Id, тип int – искусственный атрибут, хранит в себе идентификатор типа блюд;
  + Name, тип char - содержит наименование типа блюд.

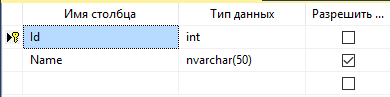


Рисунок 3.12. Таблица DishType

## 3.4 Разработка сценариев работы с данными

* Добавление данных:
  + Добавление блюд. При создании нового блюда в таблицу Dish, добавляется новая запись, автоматически генерируется ее идентификатор Id, в поле Name заносится название блюда, в поле Type заносится идентификатор типа блюда, в поля Frequency и Cost автоматически добавляется значение «0».
  + Добавление ингредиентов. При создании ингредиента в таблицу Ingredient добавляется новая запись, автоматически генерируется ее идентификатор Id, в поле Name заносится название ингредиента, в поле Cost заносится предполагаемая цена ингредиента, в поле Count автоматически добавляется значение «0».
  + Добавление меню. При создании меню в таблицу Menu добавляется новая запись, автоматически генерируется ее идентификатор Id, в поле Date заносится дата, на которую создано меню.
  + Добавление требований к меню. При добавлении требований в таблицу MenuRequirement добавляется новая запись, автоматически генерируется ее идентификатор Id, в поле Menu заносится идентификатор меню, в поле DishType заносится идентификатор типа блюд, в поле Count заносится необходимое количество блюд выбранного типа.
  + Добавление состава меню. При добавлении состава меню в таблице MenuComposition добавляется новая запись, автоматически генерируется ее идентификатор Id, в поле Menu заносится идентификатор меню, в поле DishComposition добавляется блюдо, тип которого указан в требовании к меню с минимальным значением поля Frequency.
  + Добавление поставки. При создании поставки в таблицу Entrance добавляется новая запись, автоматически генерируется ее идентификатор Id, в поле Date заносится дата прихода поставки, в поле Recalculation автоматически добавляется значение «false».
  + Добавление состава поставки. При добавлении состава поставки в таблицу EntranceComposition добавляется новая запись, автоматически генерируется ее идентификатор Id, в поля Entrance и Ingredient добавляются идентификаторы поставки и ингредиента соответственно, в поля Count и Cost добавляются цена ингредиента и его количество в поставке.
  + Добавление типа блюд. При создании типа блюд в таблицу DishTypes добавляется новая запись, автоматически генерируется ее идентификатор Id, в поле Name заносится название типа.
  + Добавление состава блюд. При создании состава блюд в таблицу DishComposition добавляется новая запись, автоматически генерируется ее идентификатор Id, в поля Ingridient и Dish заносятся идентификаторы ингредиента и блюда, в поле Count заносится количество необходимого ингредиента.
  + Добавление пользователей. При создании нового пользователя в таблицу AspNetUsers добавляется новая запись, автоматически генерируется ее идентификатор Id, в поле Email и UserName записывается email пользователя (они будут использоваться для авторизации), в поле PasswordHash записывается пароль введенный пользователем в хешированном виде, в поле SecurityStamp записывается автоматически сгенерированная “соль” к паролю.
* Изменение данных:
  + Пересчет стоимости блюда. В выбранной записи из таблицы Dish в поле Cost меняется значение на сумму стоимостей связанных ингредиентов, умноженных на количество, необходимое для приготовления блюда.
  + Пересчет стоимости ингредиентов. Выбирается запись из таблицы Entrance, находятся связанные с ней записи в таблице EntranceComposition, затем находятся связанные с последней записи в таблице Ingredient, в двух последних таблицах поля Cost и Count перемножаются и суммируются, затем делятся на сумму полей Count. Полученное значение заносится в поле Cost таблицы Ingredient, а в поле Count инкрементируется полем Count таблицы EntranceComposition.
* Удаление данных:
  + Удаление состава поставки. Происходит поиск составляющей поставки из таблицы по идентификатору и удаление всей записи.
  + Удаление состава меню и требований к меню происходит аналогичным образом.

## 3.5 Реализация сценариев работы с данными

* Работа с данными на уровне БД:

Работы с базой данных происходит только на стороне клиента.

* Работа с данными на уровне клиента:

В любом приложении, работающим с БД через Entity Framework, нам нужен будет контекст (класс производный от DbContext) и набор данных DbSet, через который мы сможем взаимодействовать с таблицами из БД. В данном случае таким контекстом является класс MenuUnitEntities.

DbContext: определяет контекст данных, используемый для взаимодействия с базой данных.

DbSet<TEntity>: представляет набор сущностей, хранящихся в базе данных.

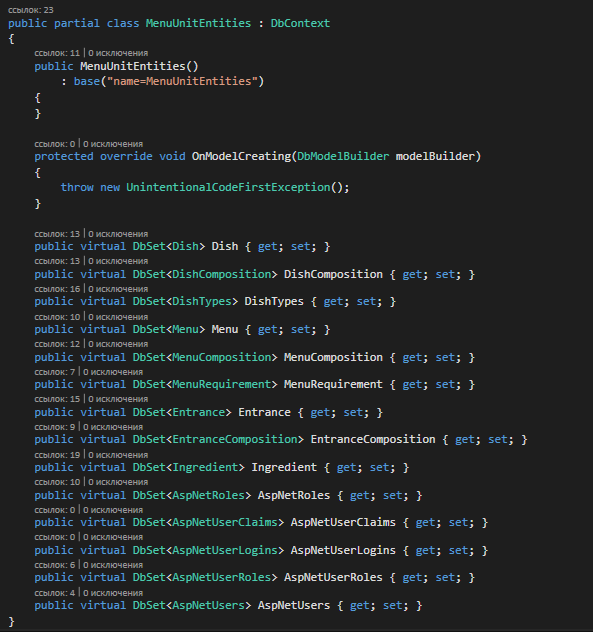


Рисунок 3.13. Контекст БД

И теперь нам надо установить подключение к базе данных. Для установки подключения обычно используется файл конфигурации приложения. В проектах для десктопных приложений файл конфигурации называется *App.config* (как в нашем случае), в проектах веб-приложений - *web.config*. В нашем случае это файл *Web.config*. После добавления Entity Framework он выглядит примерно следующим образом:

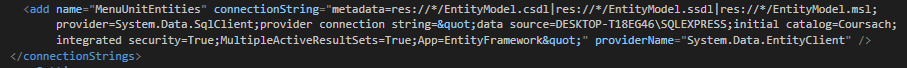


Рисунок 3.14.Конфигурационный файл.

Теперь чтобы работать с данными в любой части программы просто создаем элемент контекста и работаем с ним.



Рисунок 3.15. Контекст.

* **Сценарии добавления данных**

Рассмотрим добавление новой записи в таблицу БД (например, в таблицу Entrance).

Получаем из представления значения полей сущности entrance, проверяем валидность введенных пользователем данных, добавляем сущность в таблицу Entrance и сохраняем изменения в БД.

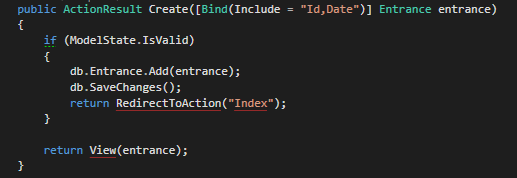


Рисунок 3.16 Добавление данных в таблицу Entrance

Алгоритмы добавления блюда, ингредиента, меню, типа блюд, поставки, состава поставки и требований к меню работают аналогичным образом.

* **Сценарии изменения данных**

Рассмотрим алгоритм пересчета стоимости блюда. Производится поиск блюда в контексте по идентификатору изменяемого блюда и загружаются все его компоненты. Новая стоимость блюда складывается из стоимости компонент умноженной на их количество. Состояние блюда помечается измененным, затем изменения сохраняются в БД.

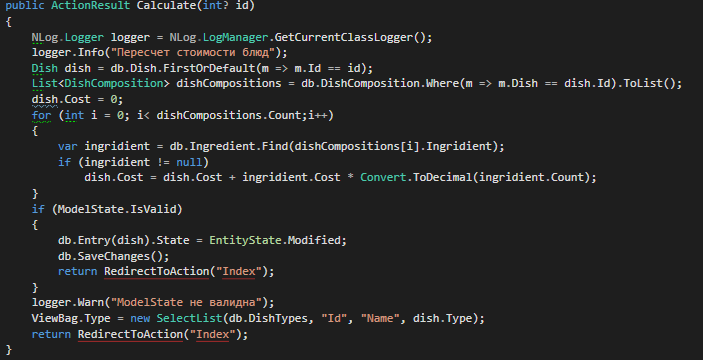


Рисунок 3.17 Изменение данных в таблице Dish

Алгоритм изменения ингредиента работает аналогичным образом.

* **Сценарии удаления данных**

Все сценарии удаления реализованы идентично:

С помощью объекта контекста производим поиск записи из таблицы по параметру id. Найденную запись удаляем из таблицы и сохраняем изменения в базе данных.

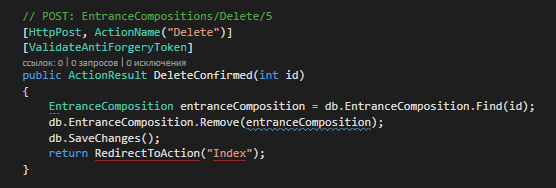


Рисунок 3.18 Удаление данных из БД

# Использование библиотеки Nlog для учёта активности пользователей web-сайта.

## Описание технологии.

NLog – бесплатная платформа логгирования для .NET с богатыми возможностями маршрутизации и управления журналами. Он позволяет легко создавать и управлять высококачественными журналами для вашего приложения независимо от его размера или сложности.

Он может обрабатывать диагностические сообщения, испускаемые с любого языка .NET, дополнять их контекстной информацией, форматировать их в соответствии с вашими предпочтениями и отправлять их на одну или несколько целей, таких как файл или база данных.

NLog - это программное обеспечение с открытым исходным кодом, лицензированное в соответствии с лицензией BSD.

Основная идея логирования в том, чтобы в необходимых методах, фиксировать сообщение, набор параметров, который передавался, чтобы в дальнейшем отслеживать и отлаживать корректность работы.

Преимущества:

* Простота настройки;
* Множество готовых шаблонов сообщений;
* Возможность написать свой шаблон;
* Открытый исходный код библиотеки;
* Поддержка устаревших модулей;

## Пример использования.

Для реализации логирования в среде разработки Visual Studio нужно добавить в ссылку на библиотеку NLog в текущем проекте. Свойства библиотеки представлены на рисунке 4.1.

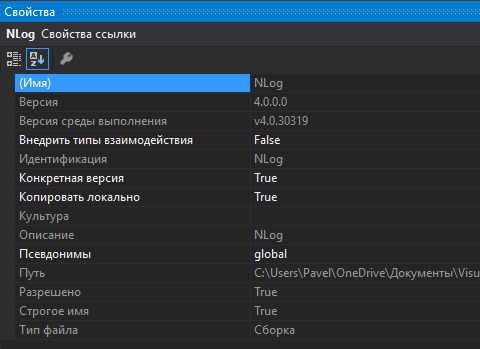


Рисунок 4.1. Ссылка на библиотеку NLog

Для того, чтобы сконфигурировать работу библиотеки в проекте, нужно добавить в конфигурационный файл проекта соответствующую секцию с необходимыми параметрами.

Конфигурационный файл представлен на рисунке 4.2.

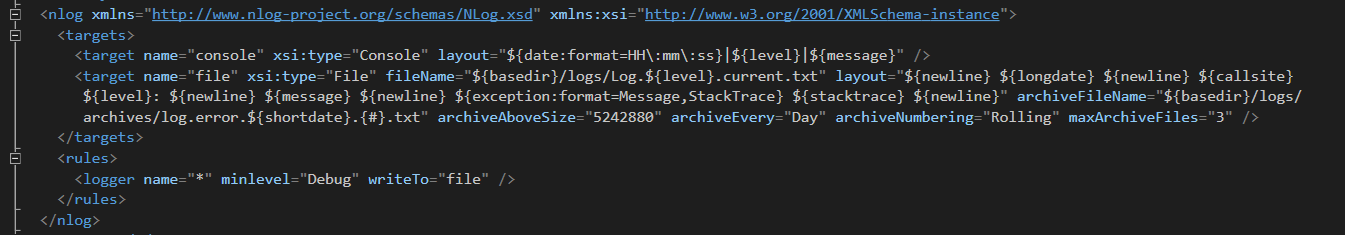


Рисунок 4.2. Добавление в Web.config проекта секции targets для конфигурирования работы NLog.

Также необходимо подключить в проект пространство имён: using NLog, чтобы пользоваться объектом класса: Logger. Пример использования логгера в контроллере представлено на рисунке 4.3.

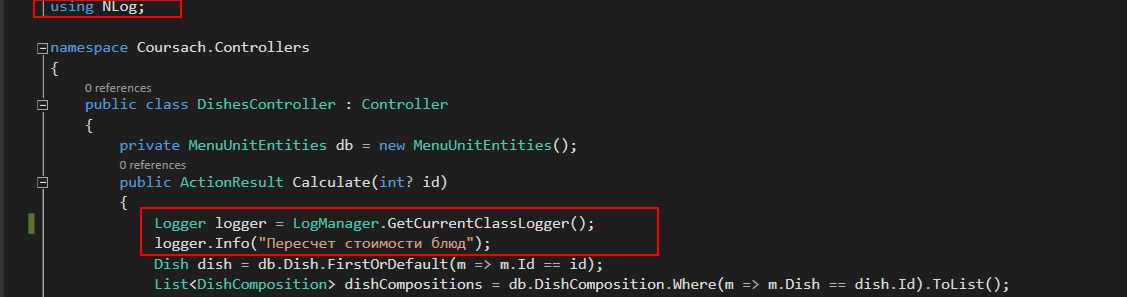


Рисунок 4.3. Контроллер, использующий библиотеку NLog.

Логгер позволяет записывать события разных типов. В зависимости от использованного типа сообщения создастся файл лога (Error – критическая ошибка, Info – информация о выполняемом действии, Warn – предупреждение)

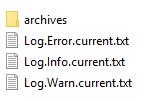


Рисунок 4.4 Разновидности txt-файлов логгера.

Пример информационного лога представлен на рисунке 4.5.

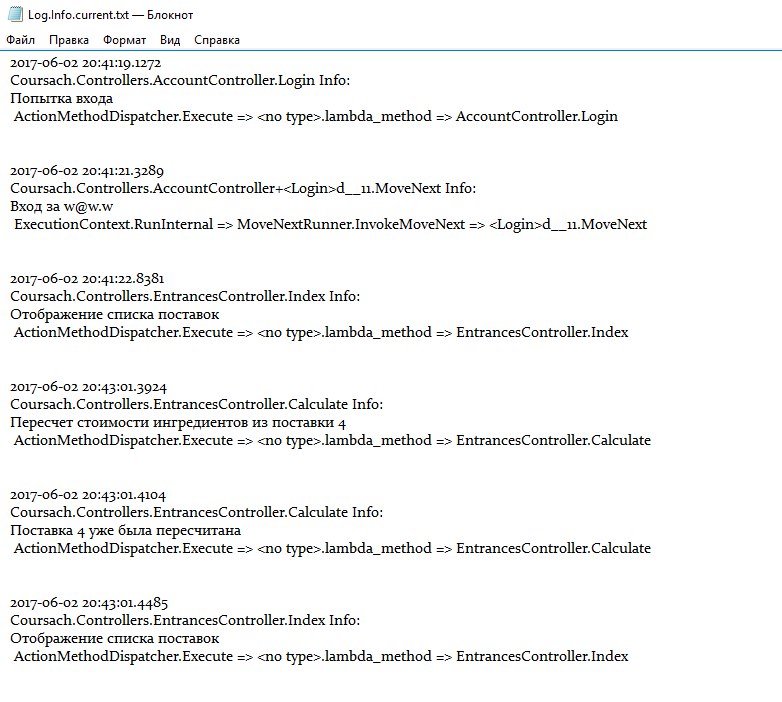


Рисунок 4.5. Пример лога на вывод сообщений по типу: Info.

# Использование библиотеки MVC Site Map Provider для разграничения прав доступа пользователей к ресурсам web-сайта.

## Описание технологии.

MvcSiteMapProvider - это инструмент, ориентированный на ASP.NET MVC, который предоставляет меню, карты сайта, функциональность маршрута карты сайта и многое другое. Он обеспечивает возможность настройки иерархической структуры навигации с использованием подключаемой архитектуры, которая может быть построена на базе XML, базы данных или кода. Обеспечивает поддержку многопользовательских приложений, гибкое кэширование, инъекцию зависимостей и несколько точек расширяемости на основе интерфейса, где практически любая часть поставщика может быть заменена пользовательской реализацией.

Основываясь на областях, именах контроллеров и методах действий, а не жестко заданных URL-адресах, узлы Sitemap полностью динамически основаны на механизме маршрутизации, используемом в приложении. Поддержка поисковой оптимизации также предоставляется в виде динамических XML-файлов Sitemaps, канонических тегов URL-адресов и тегов мета-роботов, чтобы обеспечить отправку согласованной, а не конфликтной информации о ваших URL-адресах.

Основная идея формирования карты сайта в том, чтобы на определённое представление, из всех реализованных в проекте, фиксировать определённую роль доступа.

Преимущества:

* Открытый исходный код;
* Динамичность карты сайта;
* Возможность хранения в базе данных;

## Пример использования.

Воспользуемся Nuget для того, чтобы загрузить в проект необходимые зависимости для реализации карты сайта в нашем проекте.

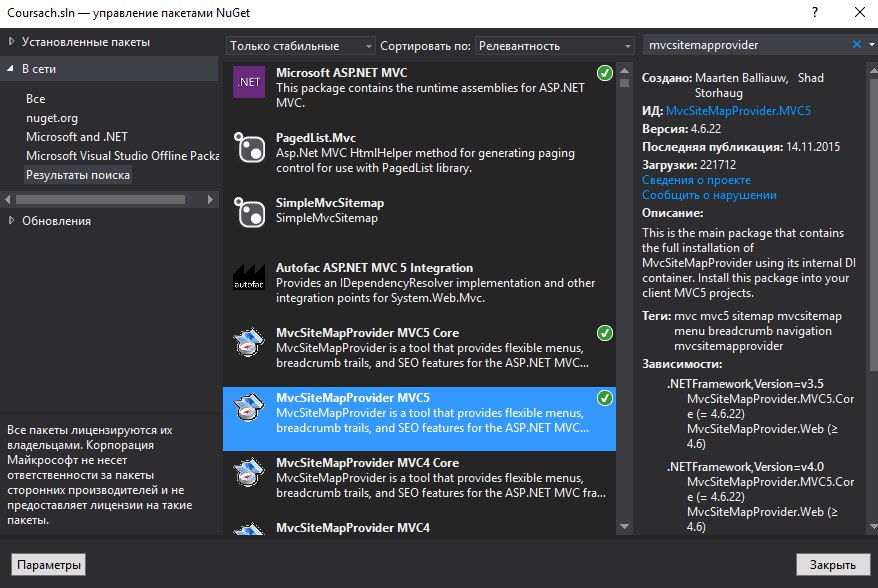


Рисунок 5.1. Загрузка необходимого пакета из Nuget.

Также предварительно необходимо добавить ссылки в текущем проекте на необходимую библиотеку.

C:\Users\Pavel\AppData\Local\Temp\snap_screen_20170603140829.png

Рисунок 5.2. Добавление ссылок на MvcSiteMap MVC5 для интеграции в проекте.

Затем добавим в проект, в файл Web.config необходимые xml-тэги в секции appSettings и namespaces.

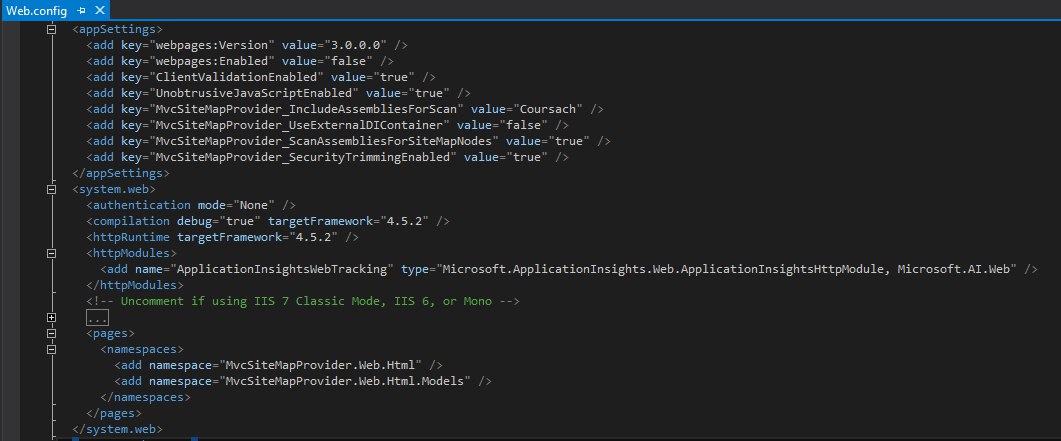


Рисунок 5.3. Добавление необходимых тегов в конфиг проекта.

Далее сконфигурируем работы карты сайта в проекте при помощи добавление файла Mvc.sitemap.

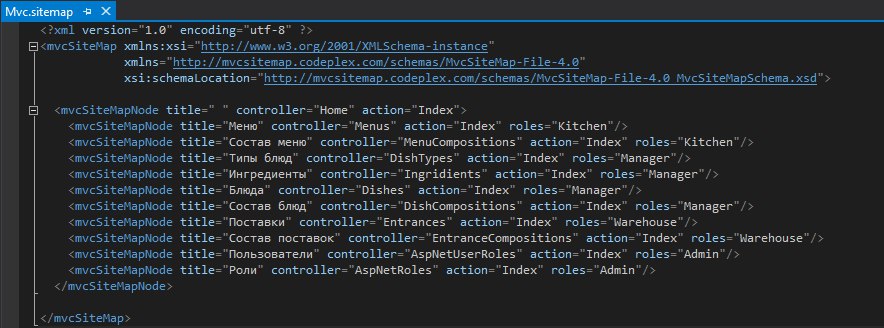


Рисунок 5.4. Файл конфигурации Mvc.sitemap.

Теперь необходимо добавить sitemap-ноду на представление.

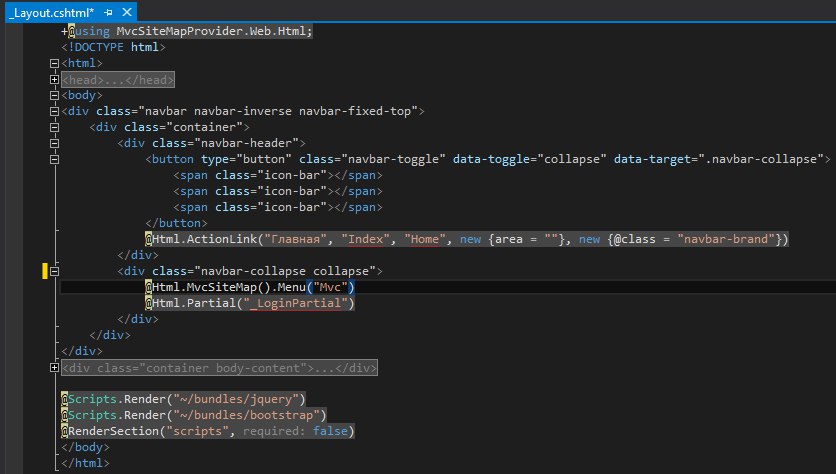


Рисунок 5.5. Добавление sitemap-ноды на представление.

# Технологии разработки и программная реализация

## Выбор операционной системы

В качестве операционной системы для развертывания информационной системы выбираем Windows 7.

Выбор обусловлен следующими факторами:

* На машинах предприятия, для которого предназначена разрабатываемая информационная система, установлен Windows 7 и комплект необходимых для работы программ, смена операционной системы будет нецелесообразной;
* Система обладает достаточно высоким уровнем надежности;
* Система обладает достаточно высоким уровнем безопасности;
* Система является удобной и простой в использовании.

## Выбор типа взаимодействия пользователя с системой, технологии пользовательского интерфейса

Для взаимодействия пользователя с системой создаем интерфейс в виде web приложения на основе технологии ASP.NET, размещаемого на сервере.

Данный выбор обусловлен низкой производительностью рабочих станций пользователей КОМК, ввиду этого было решено положить основную нагрузку на сервер, реализовав «Тонкого клиента». Благодаря тому, что приложение не требуется запускать с рабочей станции, а лишь получать содержимое страниц с сервера, скорость выполнения базового процесса будет выше.

## Выбор технологии взаимодействия пользовательских компонент с данными

Взаимодействие приложения с данными будет построено с использованием технологии ADO.NET Entity Framework.

Данный выбор обусловлен тем, что разработка приложения ведется с использованием технологии из стека Microsoft ASP.NET. Выбор технологии из

того же стека сократит время на разработку приложения, ввиду снижения временных затрат на налаживание взаимодействия между технологиями.

## Выбор языка и среды программирования

В качестве среды программирования для системы выбираем Microsoft Visual Studio, а в качестве языка C#. Данный язык и среда являются универсальными инструментами программирования, поэтому они подходят для решения поставленной задачи по созданию ИС.

Данный выбор обусловлен проектными требованиями к приложению. Приложение должно быть разработано на языке C#.

## Определение состава компонент

При формировании компонент рекомендует ориентироваться по трем логическим сервисным уровням:

* пользовательскому;
* прикладному;
* уровню данных.

**Пользовательский уровень** — отображает данные и позволяет пользователю редактировать их.

**Прикладной уровень** — здесь реализованы бизнес-правила и ограничения на данные. И хотя его сервисы используются презентационным уровнем (пользовательским), он не привязан к какому-либо клиенту — сервисы прикладного уровня доступны любому клиенту. Бизнес-правила выражаются в форме прикладных алгоритмов, корпоративных правил и т. д. Бизнес правила обычно реализуются отдельным модулем на централизованном сервере, что дает возможность доступа к нему сразу нескольким клиентам.

**Уровень данных** — прикладной уровень не знает, как и где хранится обрабатываемая им информация. В этом вопросе он полагается на сервисы доступа к данным, выполняющие всю работу по получению и передаче данных. Сервисы доступа к данным также реализуются в виде изолированных модулей, «знающих» о месте хранения информации. Таким образом, если хранилище перемещено или изменен его формат, потребуется обновить только сервисы доступа к данным. Каждый модуль доступа к данным, как правило, отвечает и за целостность хранилища (например, реляционной базы данных). Для многоуровневых приложений в качестве хранилища информации подходят простые системы управления базами данных (СУБД), необходимые для обслуживания данных в таблицах и быстрой выборки информации (например, с помощью индексов).

Разрабатываемая информационная система будет построена по трехуровневой технологии – пользовательский уровень, прикладной уровень и уровень данных.

## Разработка схемы развертывания

Диаграмма среды развертывания представлена на рисунке 6.1

Рисунок 6.1 Среда развертывания

На серверной машине предприятия уже установлен сервер SqlServer. На сервер необходимо развернуть базу данных KOMK, установить IIS сервер, а также загрузить разработанное приложение.

На клиентских машинах необходимо наличие браузера.

После загрузки всех компонентов необходимо провести тестовый запуск программы.

## Требования к системе

При разработке информационной системы были выявлены следующие требования:

* Возможность создания блюд;
* Возможность создания ингредиентов;
* Возможность формирования блюд;
* Возможность вычисления стоимости блюд;
* Возможность создания меню;
* Возможность создания требований к меню;
* Возможность формирования меню;
* Возможность создания поставки;
* Возможность формирования поставки;
* Возможность вычисления стоимости ингредиентов на складе.

## Разработка компонент

* Главная страница приложения представлена на рисунке 6.2. Реализованы возможности создания блюд, меню, ингредиентов и поставок. Любые действия с блюдами, меню, ингредиентами и поставками возможны только после авторизации пользователя.

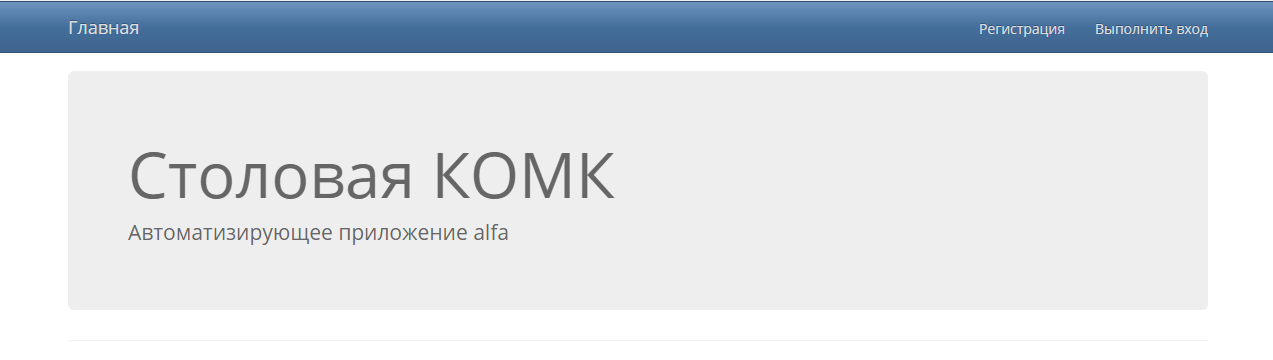


Рисунок 6.2. Главная страница приложения

Блок «Блюда и компоненты» (доступен только менеджеру) содержит следующие страницы:

* Страница создания блюда представлена на рисунке 6.3. Реализована возможность создания блюд.

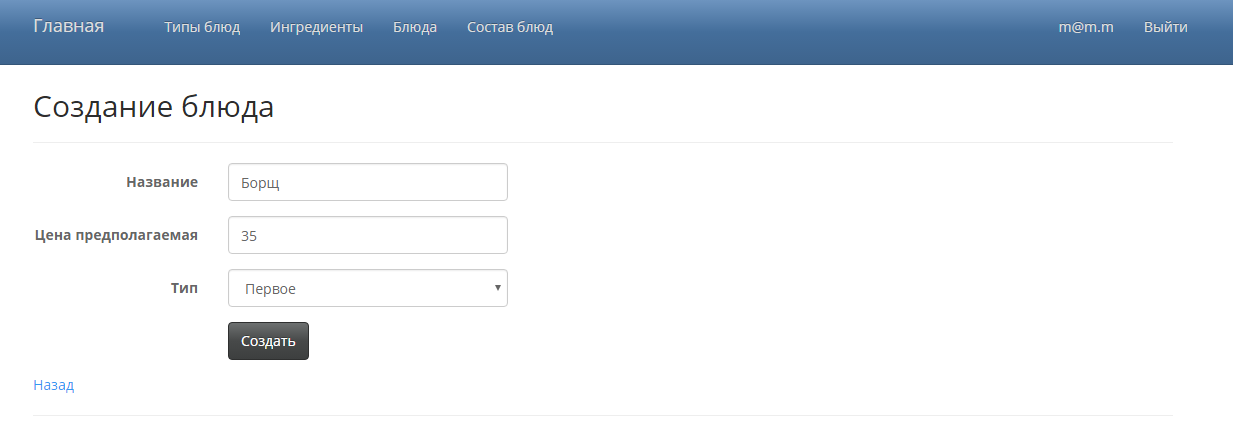


Рисунок 6.3. Страница создания блюда

* Страница создания ингредиента представлена на рисунке 6.4, реализована возможность создания ингредиентов.

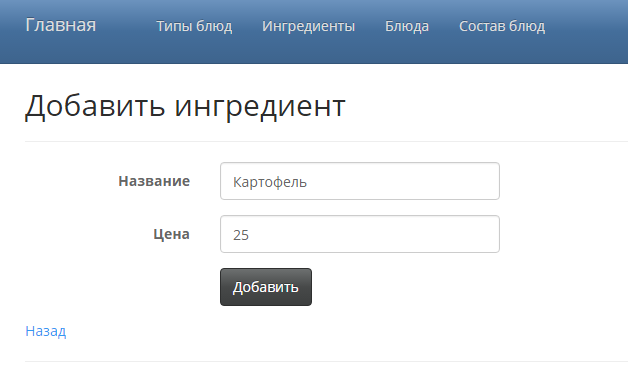


Рисунок 6.4. Страница создания ингредиента

* Страница создания компонентов блюда представлена на рисунке 6.5. Реализована возможность формирования блюда.



Рисунок 6.5. Страница создания компонента блюда

* Страница просмотра блюд представлена на рисунке 6.6. Реализованы возможности создания блюд и вычисления стоимости блюд.

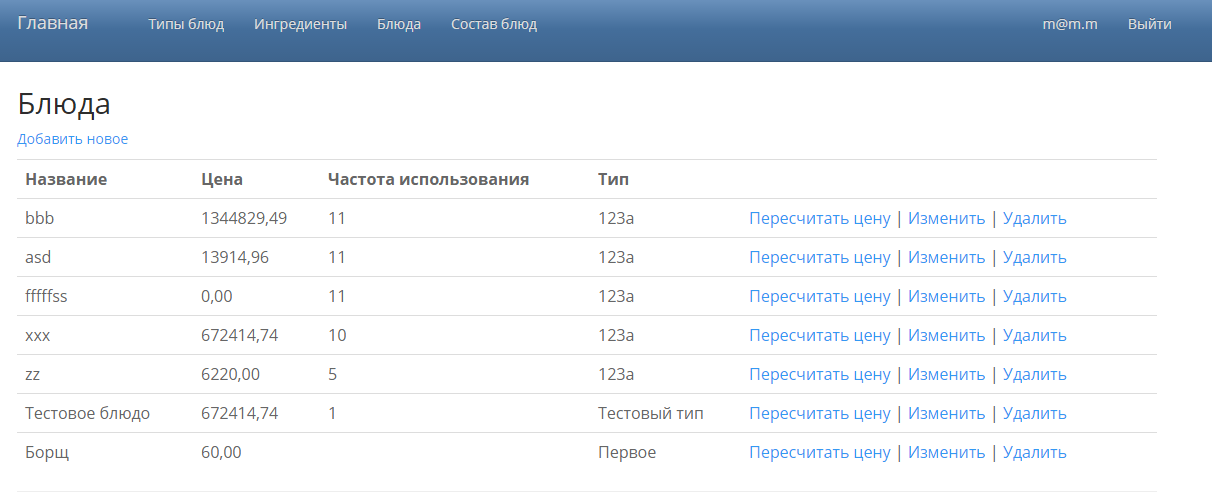


Рисунок 6.6. Страница просмотра блюд

* Страница создания меню представлена на рисунке 6.7. Реализована возможность создания меню.

Блок «Меню» (доступен только сотруднику кухни) содержит следующие страницы:

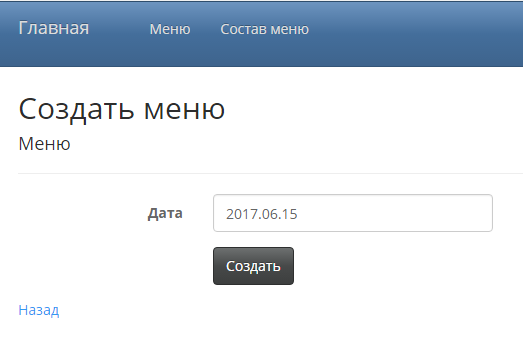


Рисунок 6.7. Страница создания меню.

* Страница просмотра меню представлена на рисунке 6.8. Реализованы возможности создания меню, создания требований к меню, формирования меню.

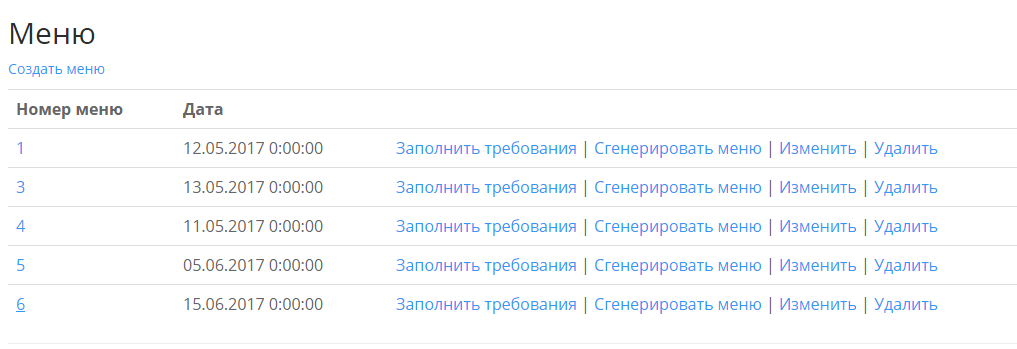


Рисунок 6.8. Страница просмотра меню.

* Страница создания требований к меню представлена на рисунке 6.9. Реализована возможность создания требований к меню.

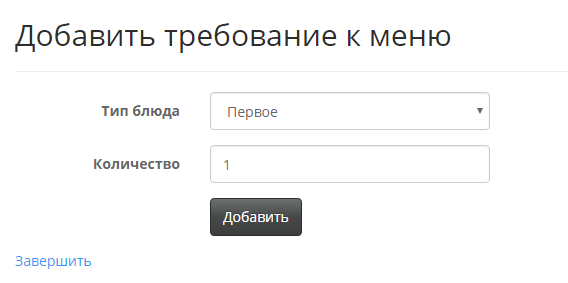


Рисунок 6.9. Страница создания требований к меню.

Блок «Стартовый» (доступен неавторизованному пользователю) содержит следующие страницы:

* Страница авторизации пользователя представлена на рисунке 6.10. Реализована возможность авторизации пользователя.

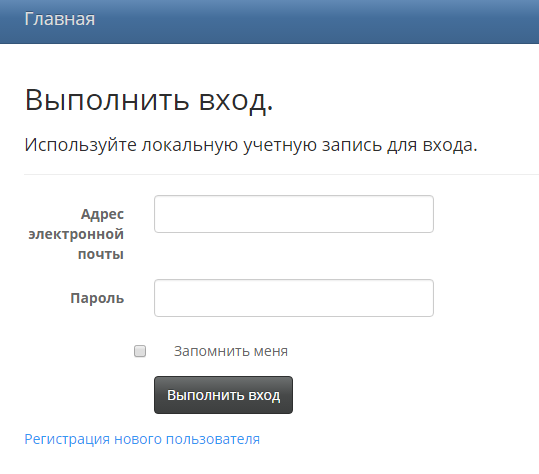


Рисунок 6.10. Страница авторизации пользователя

* Страница регистрации пользователя представлена на рисунке 6.11. Реализована возможность регистрации пользователя.

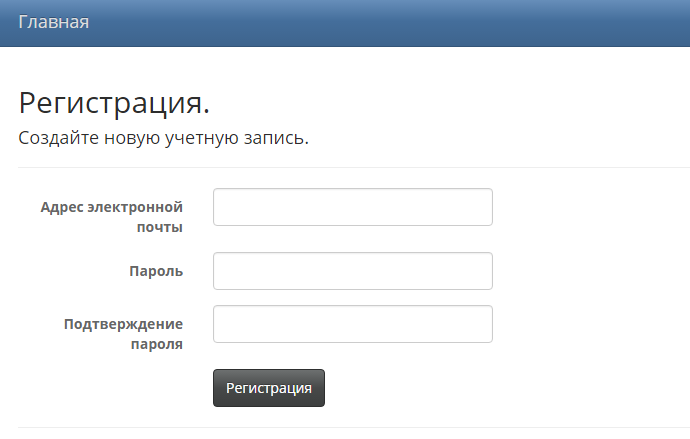


Рисунок 6.11. Страница регистрации пользователя

* Страница создания поставки представлена на рисунке 6.12. Реализована возможность создания поставки.

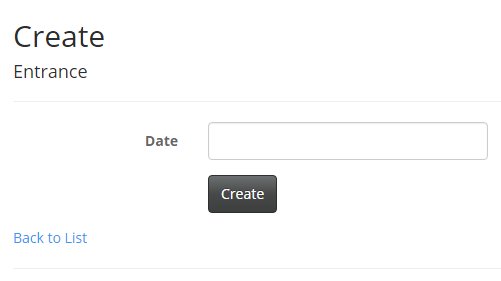


Рисунок 6.12. Страница создание поставки

Блок «Поставки» (доступен только сотруднику склада) содержит следующие страницы:

* Страница просмотр поставок представлена на рисунке 6.13. Реализована возможность создания поставок, вычисление стоимости ингредиентов на складе. Реализована возможность импорта и экспорта информации о поставках в Excel.

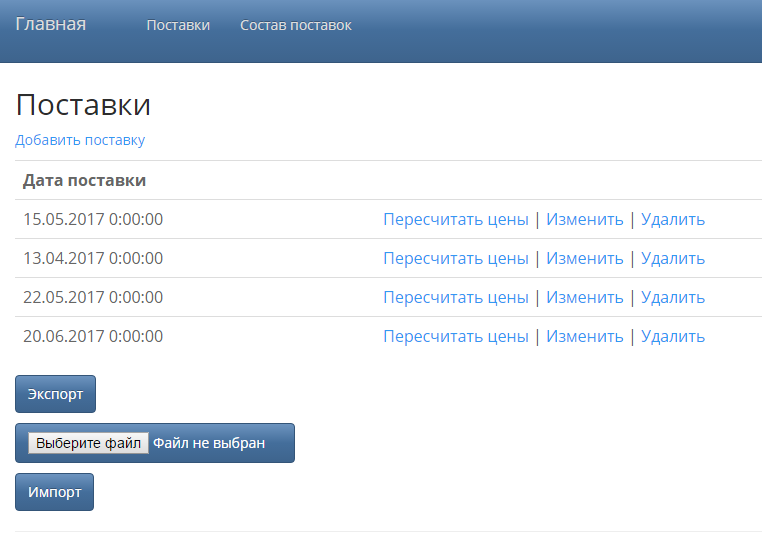


Рисунок 6.13. Страница просмотр поставок

* Страница создания состава поставок представлена на рисунке 6.14. Реализована возможность формирования поставки;

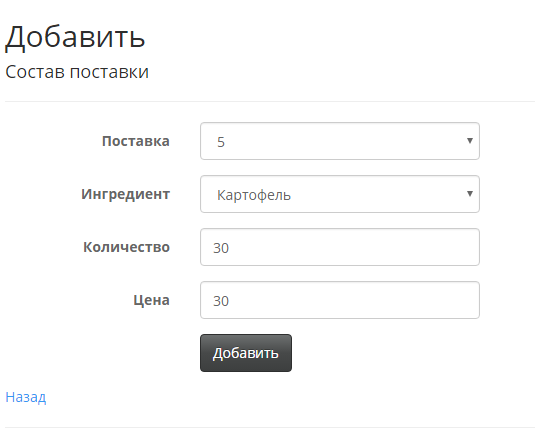


Рисунок 6.14. Страница создания состава поставок

Блок «Администрирование» (доступен только администратору) содержит следующие страницы:

* Страница «Распределение ролей», содержит информацию о ролях пользователей (роль пользователя определяет его уровень доступа к функционалу системы), реализована возможность смены роли пользователя.

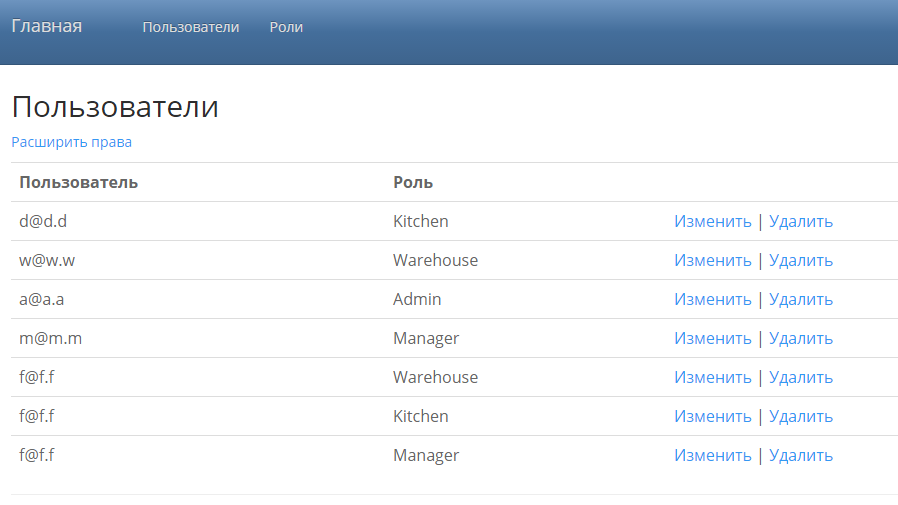


Рисунок 6.15. Страница «Распределение ролей»

Связь между страницами приложения представлена на рисунке 6.16.

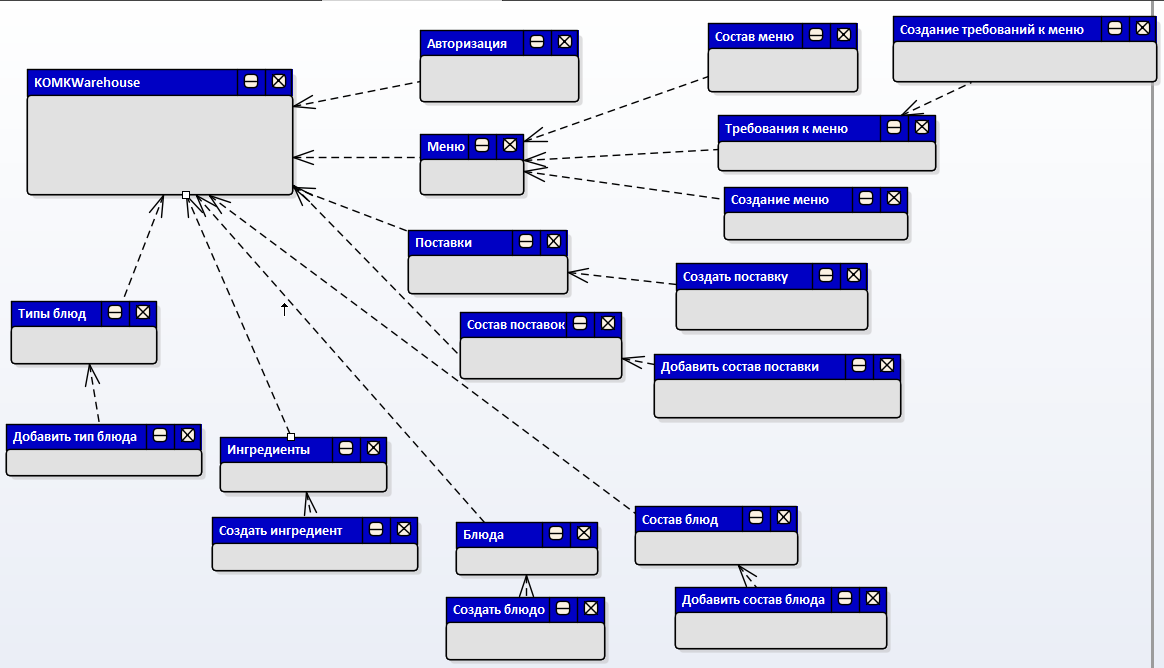


Рисунок 6.16. Связь между страницами приложения.

## Характеристика разработанного решения

Приложение было разработано на основе технологий ADO.NET Entity Framework, с использованием среды разработки Visual Studio 2015 и СУБД MS SQL Server 2012. Была выполнена реализация архитектурных вариантов использования и связанных с ними отношениями включения вариантов использования:

* Работа с блюдом;
  + Создание блюд;
  + Создание ингредиентов;
  + Формирование блюд;
  + Вычисление стоимости блюд;
* Работа с меню;
  + Создание меню;
  + Создание требований к меню;
  + Формирование меню;
* Работа с поставками;
  + Создание поставок;
  + Формирование поставок;
  + Вычисление стоимости ингредиентов;
* Администрирование;
  + Назначение прав пользователей.

Состав компонентов, содержащих программный код представлен на рисунке 6.17.

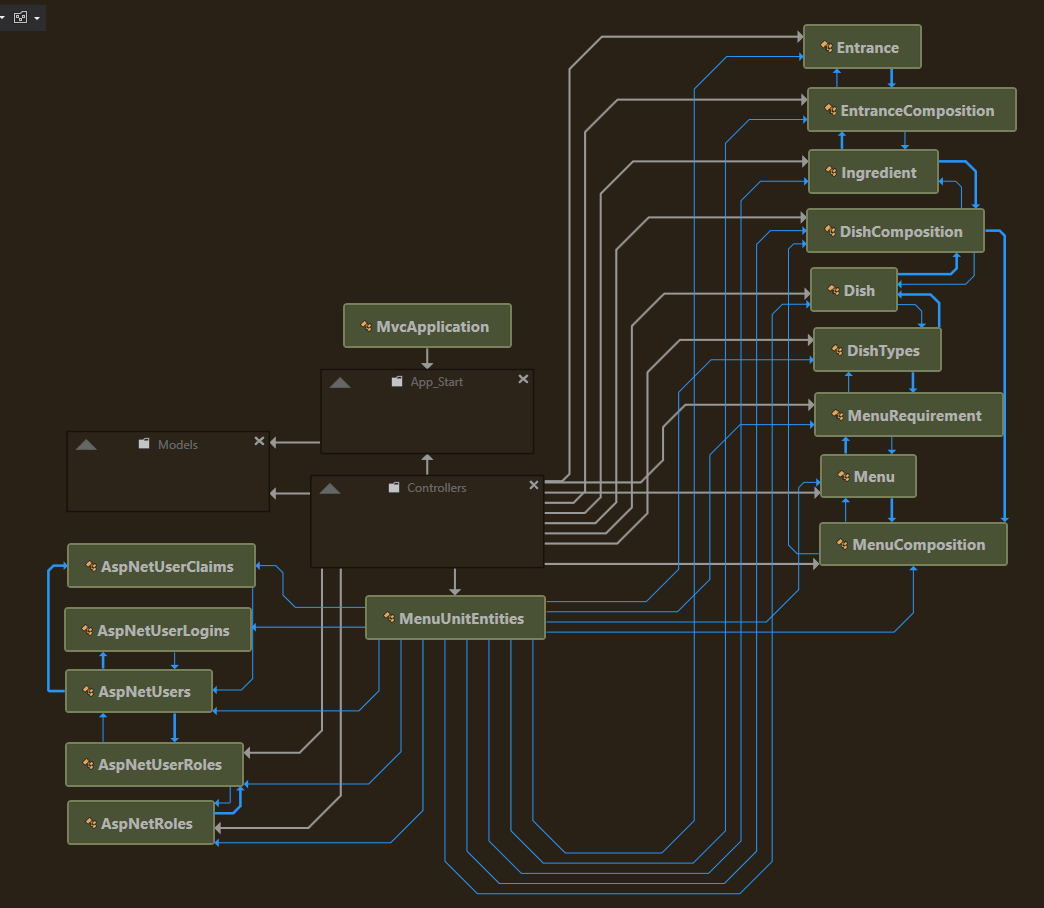


Рисунок 6.17. Состав компонентов программного кода.

## Разработка вопросов тестирования

Для тестирования выбраны следующие сценарии «Создание блюда», «Формирование блюда», «Закрытие заявки», «Смена статуса работы».

Сценарий «Создание блюда»:

1. Менеджер авторизуется.



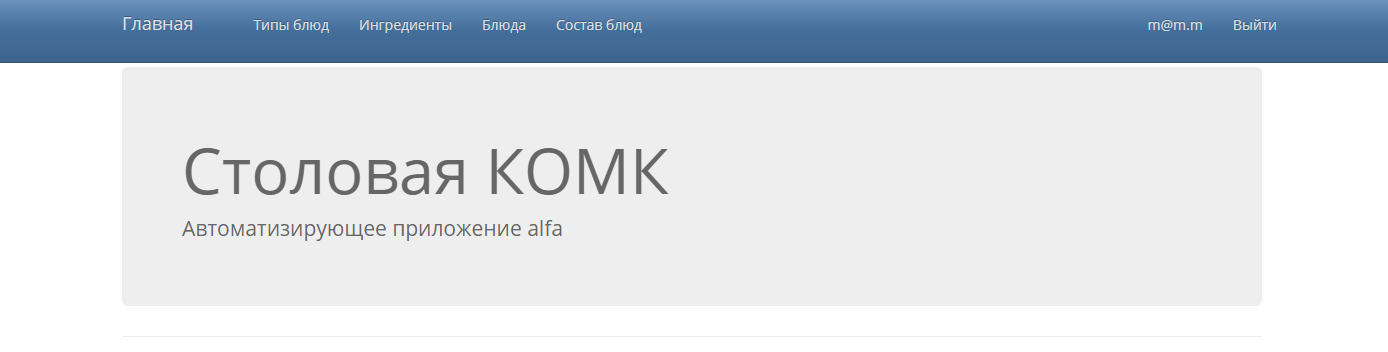
Рисунок 6.18. Вкладка «Авторизация» 

Рисунок 5.24. Вкладка «Главное меню менеджера после авторизации»

2. Открывает вкладку «Блюдо» и переходит по ссылке «Добавить Блюдо».

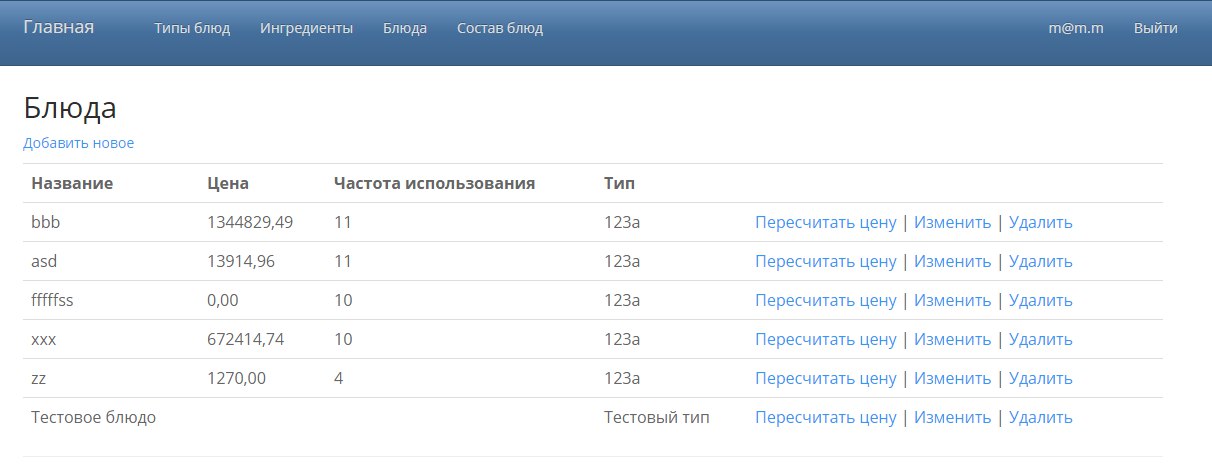


Рисунок 6.19. Вкладка «Блюда»

3. На вкладке «Создание блюда» необходимо заполнить поля данных блюда (название, цена, тип). После нажатия кнопки «Создать» в таблицу базы данных добавляется запись о созданной заявке, а на вкладке «Блюда» обновляется список блюд.



Рисунок 6.20. Вкладка «Создание блюда»

Сценарий «Формирование блюда»:

1.Менеджер авторизуется.



Рисунок 6.21. Вкладка «Авторизация»

2. Открывает вкладку «Состав Блюд» и переходит по ссылке «Создать новый».

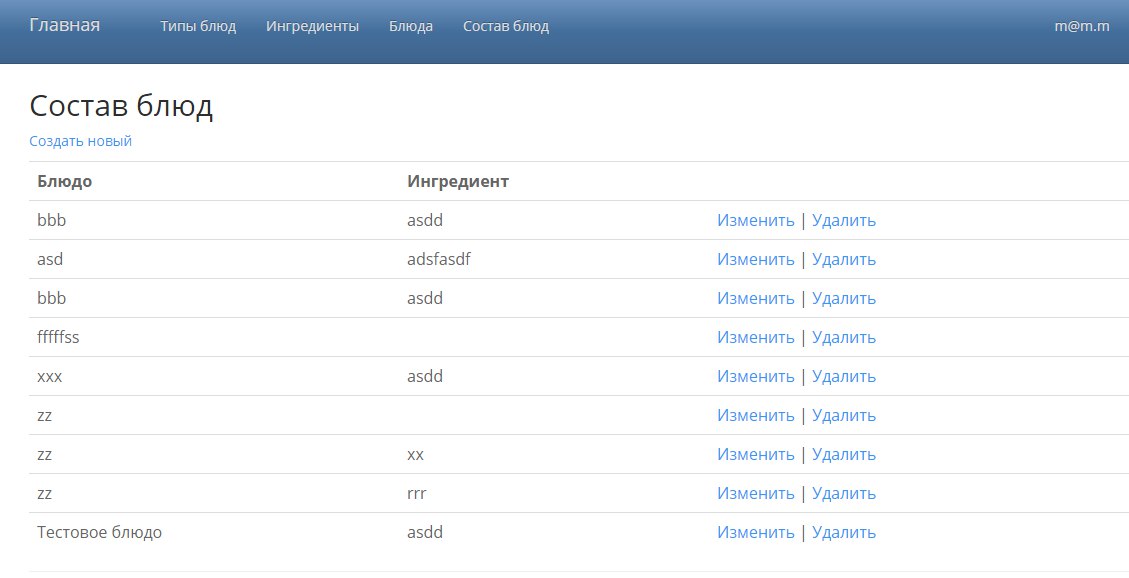


Рисунок 6.22. Вкладка «Состав блюд»

3. На вкладке «Компоненты блюда» необходимо заполнить поля данных заявки (блюдо, ингредиент, количество ингредиента). После нажатия кнопки «Создать» в таблицу базы данных добавляется запись о компонентах блюда, а на вкладке «Состав блюд» обновляется список компонентов.

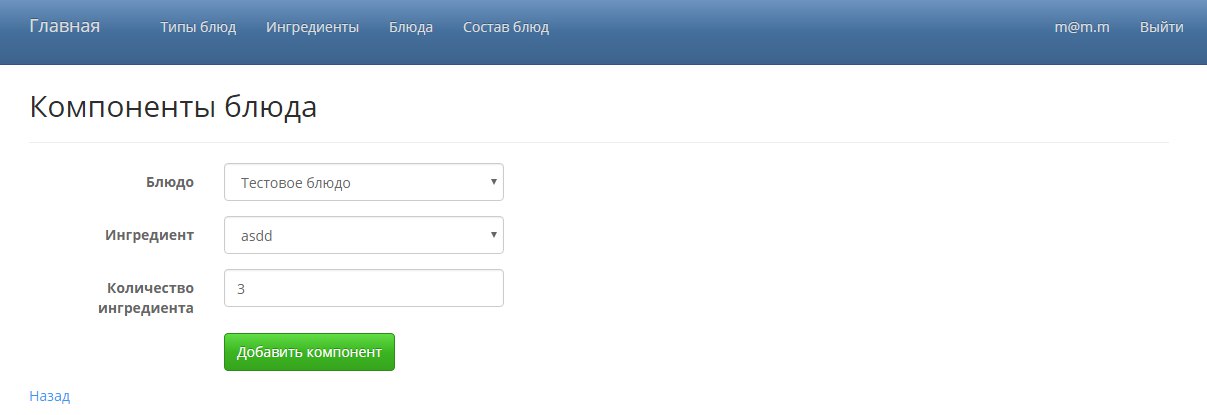


Рисунок 6.23. Вкладка «Компоненты блюда»

Сценарий «Формирование меню»:

* + - 1. Сотрудник кухни авторизуется.



Рисунок 6.24. Вкладка «Авторизация»

* + - 1. Переходит на вкладку «Меню» и нажимает на кнопку «Создать меню».

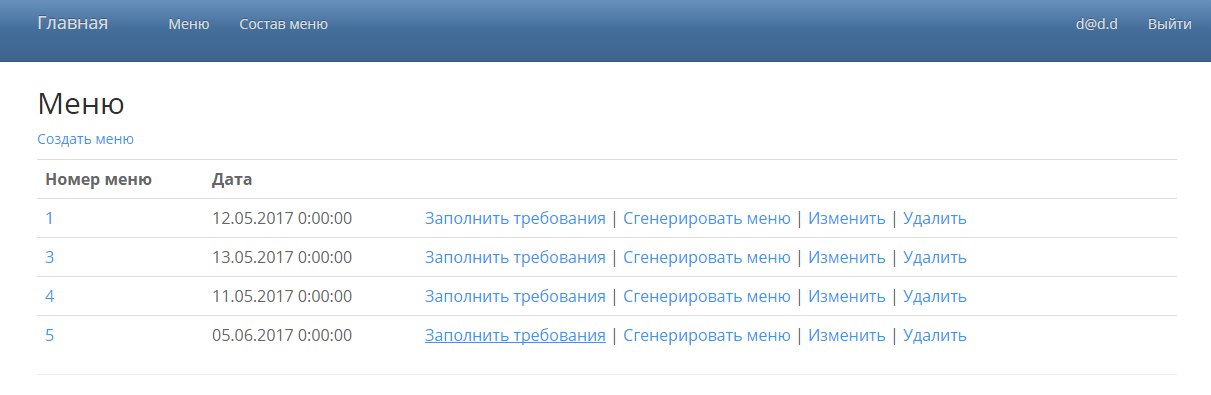


Рисунок 6.25. Вкладка «Меню»

* + - 1. Создаёт меню, для этого переходит по ссылке «Создать меню».

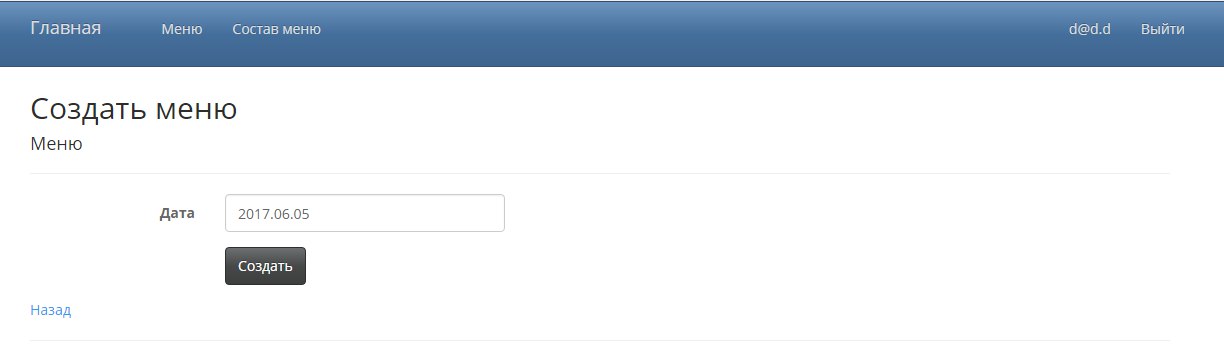


Рисунок 6.26. Вкладка «Создать меню»

* + - 1. Переходит на вкладку: “Заполнить требования”



Рисунок 6.27. Вкладка «Добавить требование к меню»

* + - 1. На вкладке «Требования к меню» можно увидеть меню с количеством блюд и требований, которые можно изменить или удалить.

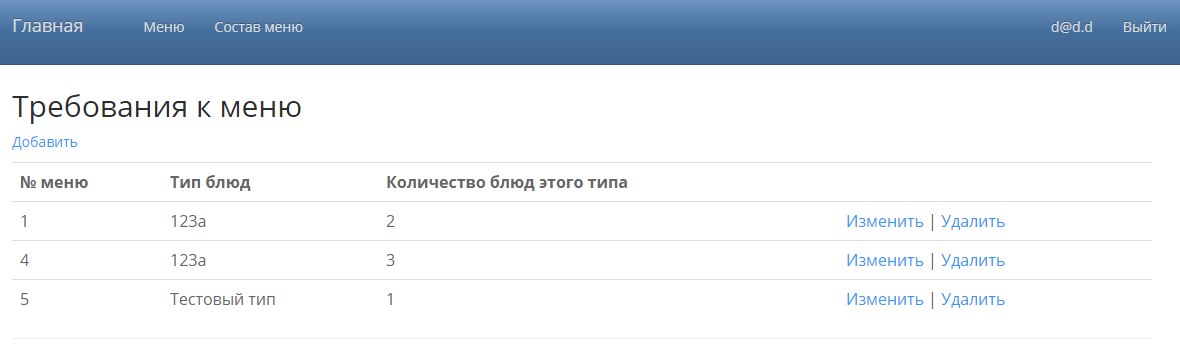


Рисунок 6.28. Вкладка «Требование к меню»

Сценарий «Названия ингредиентов и стоимость»:

1. Менеджер авторизуется.



Рисунок 6.29. Вкладка «Авторизация»

2. Переходит на вкладку «Ингредиенты».

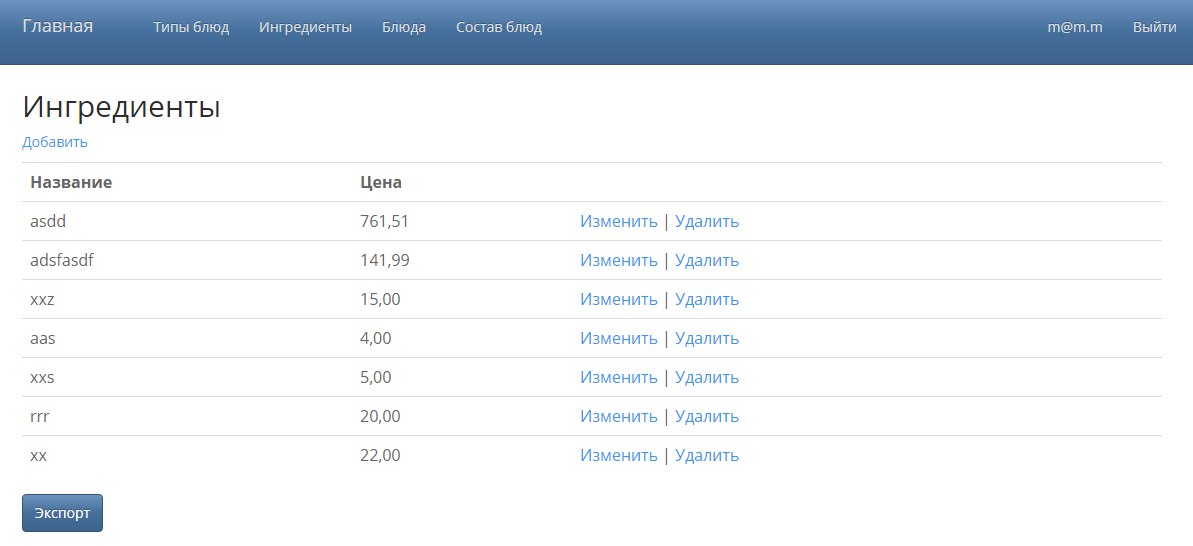


Рисунок 6.30. Вкладка «Ингредиенты»

Сценарий «ревизии прибывших в поставках продуктов питания»:

Сотрудник склада авторизуется.



Рисунок 6.31. Вкладка «Авторизация»

Главное меню, которое видит сотрудник склада после авторизации

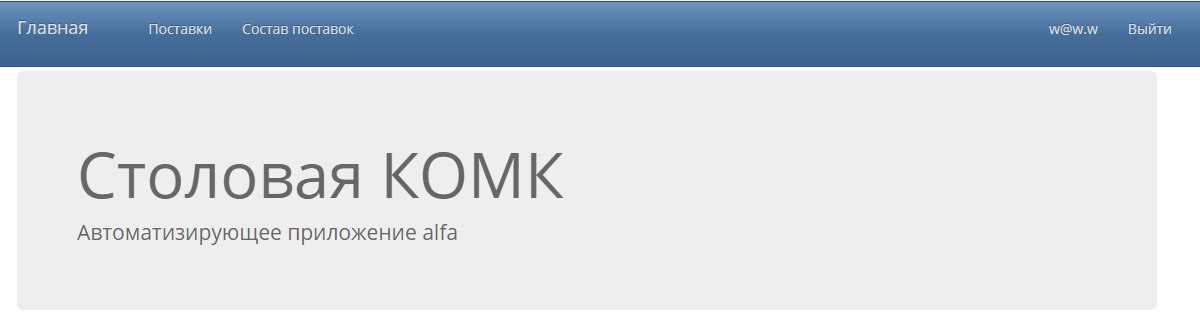


Рисунок 6.32. Главная страница после авторизации сотрудника склада

Во вкладке «Поставки» сотрудник склада видит следующие пункты.



Рисунок 6.33. Поставки

Сотрудник склада переходит по ссылке «Добавить поставку»

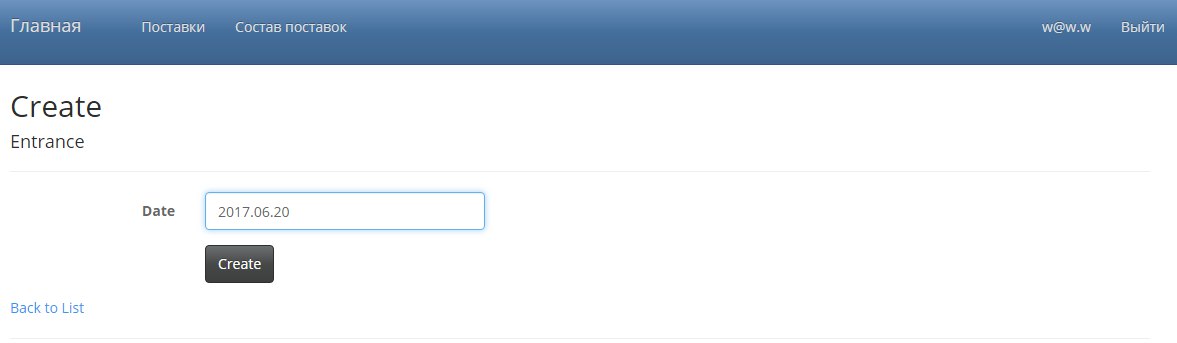


Рисунок 6.34. Добавление поставки

5. Сотрудник склада переходит по вкладке «Состав поставки»

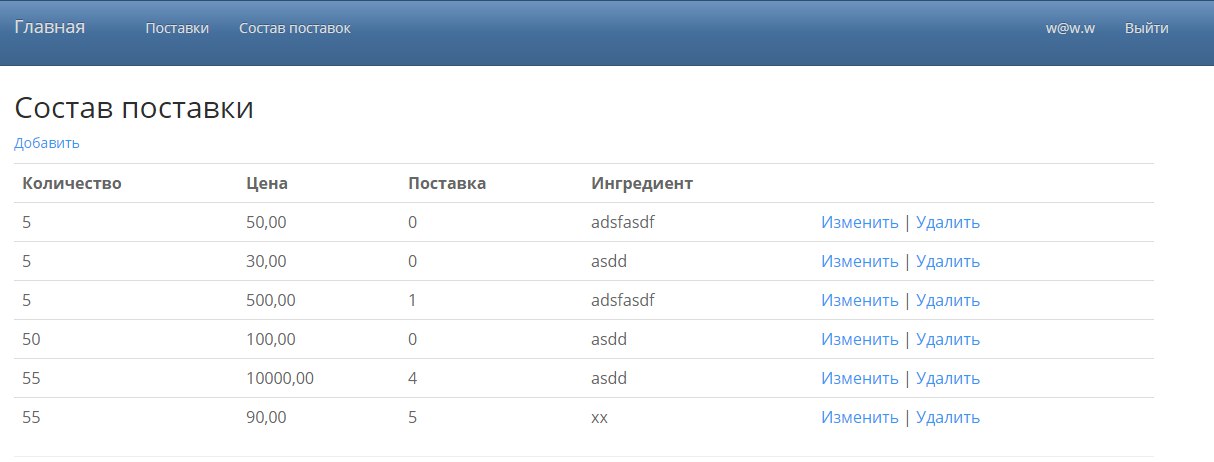


Рисунок 6.35. Состав поставки

1. Сотрудник склада переходит по ссылке «Добавить»

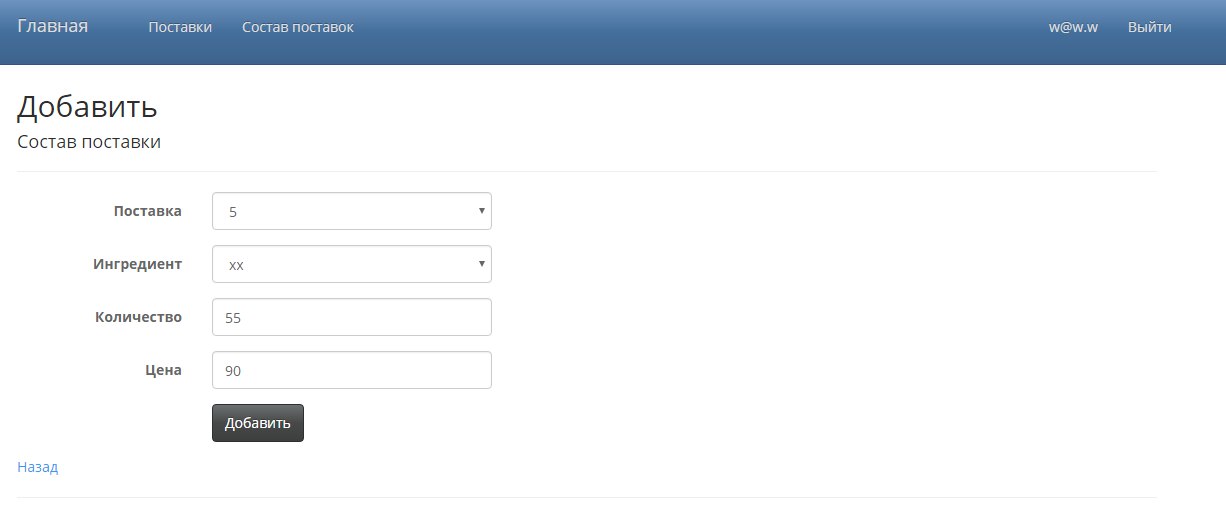


Рисунок 6.36. Добавление состава поставки

# Аппаратная и административная интеграция ИС

## Разработка схемы развертывания

Чтобы развернуть разработанное приложение на предприятии, необходимо установить приложение на сервер, затем развернуть на нем Backup прилагаемой БД “Coursach.bacpak”, требуется наличие IIS сервера. Установка MS SQL Server 2012 и службы IIS на сервер не требуется, т.к. они уже установлены на предприятии. Для работы с приложением на рабочей станции достаточно перейти на определенную веб-страницу с любого браузера.

Схема развертывания представлена на рисунке 6.1.



Рисунок 7.1. Диаграмма развертывания ИС.

## Формулировка требований к физическим устройствам и сетевому оборудованию, состав рабочих мест

Данная система была разработана для последующей загрузки на сервер с установленной ОС Windows Server.

Для функционирования разработанной ИС необходимо следующее оборудование:

* Сервер СУБД - для размещения БД;
* Рабочие станции – будут использоваться для доступа к клиентскому приложению;
* Коммутатор – для соединения нескольких узлов локальной сети;
* Сетевой кабель, для объединения рабочих станций в единую локальную вычислительную сеть (ЛВС).

Для работы с единой базой данных необходимо подключить все рабочие места к сети.

С системой будут работать как минимум 4 рабочие станции, которые должны быть подключены к сети.

Сервер будет располагаться в специально отведенной для него комнате.

Для работоспособности системы будет использоваться сеть с технологией передачи данных Fast Ethernet 100Base-TX с максимальной скоростью передачи данных 100 Мбит\сек.

## Выбор состава аппаратных средств

В качестве сервера рекомендуется использовать компьютер со следующими характеристиками:

* CPU: Intel core i3 3000;
* RAM: 6GB;
* HDD: от 1000 GB;
* ОС Windows Server.

В качестве рабочих станций рекомендуется использовать компьютер со следующими характеристиками:

* CPU: AMD Athlon-5150 1600;
* RAM: 2GB;
* HDD: 250GB;
* OC Windows XP / Windows 7.

## Расчет потребности персонала.

Расчет потребности персонала производится с учетом требований, предъявляемым к системе.

Для обеспечения безопасности и надежности информационной системы необходим следующий персонал:

* **Главный повар.**

Для обеспечения возможности генерирования меню требуется должность повара. Эта должность необходима для использования функций автоматической генерации меню.

* **Заведующий склада.**

Для ввода данных о поставках и перерасчёта стоимости ингредиентов.

* **Менеджер.**

Для добавления новых блюд, ингредиентов. Он изменяет состав блюд, может создавать новые типы блюд и перерасчитывает стоимость блюд.

* **Администратор**

Для обеспечения доступа сотрудников к ресурсам сайта. Он назначает пользователям роли.

На предприятии имеются все необходимые специалисты. Вопрос найма новых специалистов рассматривать не нужно.

## Выбор сетевой архитектуры и технологии

Выбор архитектуры

На предприятии уже настроена ЛВС, со следующими параметрами, представленными в таблице 7.1.

Таблица 7.1 Основные характеристики сети.

|  |  |
| --- | --- |
| Компонент/характеристика | Вариант сети |
| Топология | Звезда |
| Линия связи | Экранированная витая пара cat 5е. |
| Сетевые адаптеры | Fast Ethernet |
| Ретрансляторы (повторители, концентраторы, коммутаторы, мосты, маршрутизаторы, шлюзы) | Коммутаторы (D-Link, Cisco), роутер (Cisco). |
| Управление совместным использование ресурсов | Клиент - серверная архитектура построения. |
| Поддерживаемые приложения | Электронная почта, работа с БД. |

Схема ЛВС представлена в приложении 1.

Выбор технологии и аппаратных средств. Расчет сети

На предприятие установлена стандартная сеть на базе технологии Fast Ethernet.

Поскольку сеть уже создана, а дальнейшее расширение сети не связано напрямую с разрабатываемой системой, то стоимость сети в проект не закладывается.

Таблица 7.2 Расчет сети.

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование | Цена, руб. | Количество | Стоимость, руб |
| Оборудование: | | | |
| Кабель UTP 5e, м | 12 | 80 | 960 |
| Коннектор RJ-45, шт | 4 | 10 | 40 |
| Кабель-канал, м | 10 | 80 | 800 |
| Монтаж: | | | |
| Монтаж кабель-канала, м | 25 | 80 | 2000 |
| Обжим коннектора RJ-45, шт | 23 | 10 | 230 |
| Итого:4030 | | | |

# Общие вопросы администрирования

## Определение стратегии администрирования на уровне руководства.

В ИС реализуются следующие цели администрирования:

* обеспечение безопасности и целостности данных путем управления доступом пользователей (определение каждого пользователя базы данных, назначение паролей):
* обеспечение бесперебойной работы информационной системы колледжа;
* внедрение инструкций для пользователей и администраторов.

## Определение объектов администрирования на уровне предприятия.

Объектами администрирования на уровне предприятия являются:

* рабочие станции;
* сетевые устройства (маршрутизатор);
* база данных и её объекты (таблицы БД, база выполняется на основе MS SQL Server 2012);
* программное обеспечение, в том числе разрабатываемая ИС;

## Политика администрирования на уровне предприятия.

* Для фильтрации трафика поступающего из Интернета на пограничном маршрутизаторе настроен брандмауэр;
* Разграничение по ролям осуществляется администратором с использованием службы Active Directory;
* Ограничение трафика проходящего по сети производится при помощи разделения рабочих станций предприятия по VLAN-ам;
* Конфигурирование новых рабочих станций производится с использованием Active Directory;
* Режим работы сети – круглосуточный. Основным транспортным протоколом для передачи данных является протокол TCP/IP;

## Определение объектов администрирования на уровне разрабатываемой ИС.

Объекты администрирования на уровне разрабатываемой ИС:

В процессе анализа были выделены следующие объекты администрирования на уровне разрабатываемой ИС:

* разработанное приложение (клиентская часть, таблицы БД);
* база данных, включающая таблицы:
  + «AspNetUsers» - содержит данные о пользователях;
  + «Dish» - содержит данные о существующих блюдах;
  + «Ingredient» - содержит данные о существующих ингредиентах;
  + «Menu» - содержит данные о существующих меню;
  + «DishTypes» - содержит данные о существующих типах блюд;
  + «DishComposition» - содержит данные о составе конкретных блюд;
  + «MenuComposition» - содержит данные о составе конкретных меню;
  + «AspNetUserRoles» - содержит данные о ролях конкретных пользователей;
  + «MenuRequirement» - содержит данные о требованиях к конкретным меню;
  + «Entrance» - содержит данные о поставках;
  + «EntranceComposition» - содержит данные о составе конкретных поставок.
* сервер.

## Политика администрирования на уровне разрабатываемой ИС.

В разрабатываемой системе присутствует разделение по ролям:

* Сотрудник кухни
  + Имеет полный доступ к таблицам БД «Menu», «MenuComposition», «MenuRequirement», содержащим информацию о меню. Способен создавать (с использованием функционала разработанной ИС) новые меню.
  + Способен создавать новые меню, созданные меню записываются в таблицу БД «Menu». Создание меню производится с использованием функционала приложения.
  + Способен создавать новые требования к меню, созданные требования записываются в таблицу БД «MenuRequirement». Создание требований производится с использованием функционала приложения.
  + Способен создавать новые записи в таблице «MenuComposition», задает только требования в таблице «MenuRequirements» и таблица «MenuComposition» заполняется без его участия функционалом приложения.
* Сотрудник склада
  + Имеет полный доступ к таблицам БД «Entrance» и «EntranceComposition», содержащим информацию о поставках.
  + Имеет частичный доступ к таблице БД «Ingredient», содержащей информацию о ингредиентах.
  + Способен создавать новые поставки, созданные поставки записываются в таблицу БД «Entrance». Создание поставок производится с использованием функционала приложения.
  + Способен создавать состав поставки, созданные составы поставки записываются в таблицу БД «EntranceComposition». Создание поставок производится с использованием функционала приложения
  + Способен менять стоимость ингредиентов на складе, которые были включены в поставку. Изменение ингредиента производится с использованием функционала приложения.
* Менеджер
  + Имеет полный доступ к таблицам БД «Dish», «Ingredient», «DishTypes», «DishComposition».
  + Способен создавать новые типы блюд, созданные типы блюд записываются в таблицу БД «DishTypes». Создание типов блюд производится с использованием функционала приложения.
  + Способен создавать новые блюда, созданные блюда записываются в таблицу БД «Dish». Создание блюд производится с использованием функционала приложения.
  + Способен создавать новые ингредиенты, созданные ингредиенты записываются в таблицу БД «Ingredient». Создание ингредиентов производится с использованием функционала приложения.
  + Способен создавать состав блюд, созданный состав блюда записываются в таблицу БД «DishComposition». Создание состава блюд производится с использованием функционала приложения.
  + Обладает возможностью конфигурирования системы, изменяя содержимое таблицы БД «Dish».
* Администратор
  + Имеет полный доступ к таблице БД «AspNetUserRoles», содержащей информацию о правах пользователя.
  + Способен разрешать и запрещать пользователям доступ к ресурсам сайта, созданное разрешение записывается в таблицу БД «AspNetUserRoles», Создание разрешение производится с использованием функционала приложения.

Для получения доступа к функционалу системы, пользователь обязан авторизоваться в системе, введя соответствующий логин и пароль. Пароли пользователей хранятся в базе данных предприятия в хешированном виде в таблице AspNetUsers.

# Вопросы информационной безопасности

## Анализ угроз

Под угрозами информационной безопасности понимается потенциальная возможность нарушения следующих основных характеристик:

* Конфиденциальности сведений, составляющих государственную, служебную или коммерческую тайну, а также персональных данных. Заключается в том, что информация становится известной тому, кто не располагает полномочиями доступа к ней;
* Целостности и достоверности информационных, программных и других ресурсов, а также фальсификация документов. Включает в себя любое умышленное изменение информации, хранящейся в вычислительной системе или передаваемой из одной системы в другую
* Доступности информации, в условиях случайных и преднамеренных угроз информация предоставляется в том виде и том месте, которые необходимы авторизованному (санкционированному) пользователю, и в то время, когда она ему необходима. Доступность определяется как наличием необходимой информации, так и готовностью системы к обслуживанию.

В отношении информационных систем принято считать, что информационная безопасность информационных систем обеспечена в том случае, если для любых информационных ресурсов в системе поддерживается определенный уровень: конфиденциальности, целостности, доступности.

Проведем анализ угроз и их оценку с точки зрения вероятности реализации и ущерба от реализации угрозы.

Оценка вероятности возникновения угрозы:

1. очень вероятна – 9-10 баллов,
2. вероятна – 5-8 баллов,
3. маловероятна – 3-5 баллов.
4. практически невероятна 1-2 балла.

Оценка степени ущерба возникновения угрозы:

1. полная потеря данных – 9-10 баллов,
2. частичная потеря данных – 3-8 балла,
3. возможная потеря данных – 1-2 балла.

В соответствии с ГОСТ Р 21275-2006 выделены следующие угрозы (таблица 9.1).

Таблица 9.1 – Анализ угроз

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Угрозы | Меры предотвращения | Вероятность возникновения | Ущерб | Коэффициент оценки угрозы |
| Разглашение защищаемой информации лицами, имеющими к ней право доступа. | Составление регламентов, должностных инструкций для пользователей системы. | 8 | 4 | 32 |
| Несанкционированный доступ к защищаемой информации | Контроль доступа в помещения | 7 | 5 | 35 |
| Блокирование доступа к защищаемой системе путем перегрузки тех. средств обработки информации | Особая организация работы систем с поступающей информацией | 4 | 3 | 12 |
| Искажение, уничтожение или блокирование информации с применением тех. средств | Экранирование элементов системы от внешних излучений, физических воздействий | 4 | 8 | 32 |
| Ошибки обслуживающего персонала | Увеличение степени автоматизированности процессов системы; обучение персонала. | 3 | 3 | 9 |
| Явления техногенного характера | Установка систем пожаротушения | 3 | 8 | 24 |
| Природные явления, стихийные бедствия | Заземление и установка громоотводов | 2 | 8 | 16 |

## Информационная безопасность на уровне предприятия.

### Контроль доступа в помещения.

Контроль доступа в помещения регламентируется общими правилами «КОМК». В частности, доступ к помещениям, в которых установлено серверное и коммутационное оборудование, осуществляется обученным и сертифицированным персоналом и ограничивается стандартными офисными средствами безопасности (ключ от помещения выдается под роспись).

### Обеспечение безопасности с помощью аппаратных средств.

* Все аппаратные средства имеют маркировку и занесены в журналы по учету оборудования предприятия;
* Все помещения КОМК оборудованы огнетушителями;
* Сотрудники колледжа ознакомлены с правилами безопасности использования аппаратных средств.

### Обеспечение безопасности с помощью программных средств.

* На рабочие станции запрещено устанавливать нелицензионное ПО;
* Перед началом использования ПО сотрудники КОМК обязаны пройти курс обучения по работе с данными;
* Перед введением ПО в эксплуатацию, оно в обязательном порядке проходит тестирование на наличие уязвимостей и соответствие требованиям безопасности.

### Определение политики управления доступом пользователей.

* Для персонала определены полномочия и права доступа к информации в соответствии с занимаемой должностью;
* Каждому сотруднику выдан личный логин и пароль, с помощью которого происходит аутентификация личности. С периодичностью в два месяца пароль меняется для всех сотрудников колледжа.
* Данные о логинах и паролях хранятся централизованно. Передача логина или пароля другому лицу запрещена.
* Пользователи блокируют свою учетную запись, когда не работают на ней.
* При вводе пароля он не должен отображаться в явном виде.

### Требования к административным обязанностям.

В соответствии «Должностной инструкцией администратора», документом, регламентирующим обязанности администратора, на нем лежат следующие обязанности:

* 1. Устанавливает на серверы и рабочие станции операционные системы и необходимое для работы программное обеспечение.
  2. Осуществляет конфигурацию программного обеспечения на серверах и рабочих станциях.
  3. Поддерживает в работоспособном состоянии программное обеспечение серверов и рабочих станций.
  4. Регистрирует пользователей локальной сети и почтового сервера, назначает идентификаторы и пароли.
  5. Осуществляет техническую и программную поддержку пользователей, консультирует пользователей по вопросам работы локальной сети и программ, составляет инструкции по работе с программным обеспечением и доводит их до сведения пользователей.
  6. Устанавливает права доступа и контролирует использование сетевых ресурсов.
  7. Обеспечивает своевременное копирование, архивирование и резервирование данных.
  8. Принимает меры по восстановлению работоспособности локальной сети при сбоях или выходе из строя сетевого оборудования.
  9. Выявляет ошибки пользователей и программного обеспечения и принимает меры по их исправлению.
  10. Проводит мониторинг сети, разрабатывает предложения по развитию инфраструктуры сети.
  11. Обеспечивает сетевую безопасность (защиту от несанкционированного доступа к информации, просмотра или изменения системных файлов и данных), безопасность межсетевого взаимодействия.
  12. Осуществляет антивирусную защиту локальной вычислительной сети, серверов и рабочих станций.
  13. Готовит предложения по модернизации и приобретению сетевого оборудования.
  14. Осуществляет контроль за монтажом оборудования локальной сети специалистами сторонних организаций.
  15. Сообщает своему непосредственному руководителю о случаях нарушения правил пользования локальной вычислительной сетью и принятых мерах.

Системный администратор несет ответственность за:

* 1. Нарушение функционирования локальной вычислительной сети, серверов и персональных компьютеров вследствие ненадлежащего исполнения своих должностных обязанностей.
  2. Несвоевременную регистрацию пользователей локальной вычислительной сети и почтового сервера.
  3. Несвоевременное уведомление руководства о случаях нарушения правил пользования локальной вычислительной сетью.
  4. Системный администратор привлекается к ответственности:
     1. За ненадлежащее исполнение или неисполнение своих должностных обязанностей, предусмотренных настоящей должностной инструкцией - в пределах, установленных действующим трудовым законодательством Российской Федерации.
     2. За правонарушения, совершенные в процессе своей деятельности - в пределах, установленных действующим административным, уголовным и гражданским законодательством Российской Федерации.
     3. За причинение материального ущерба компании - в пределах, установленных действующим законодательством Российской Федерации.

### Антивирусная защита информации.

* На каждой рабочей станции установлено антивирусное ПО;
* Внешние носители сканируются на наличие вирусного ПО;
* Отключение антивирусного ПО запрещено;
* Раз в неделю проводится сканирование рабочих станций на наличие вирусного ПО.

### Информационная безопасность на уровне АИС.

Так как рабочие станции на предприятии уже защищены от выделенных угроз, разрабатываемая информационная система не нуждается в дополнительной защите.

В разрабатываемой информационной системы будет использоваться механизм идентификации пользователей по логину и паролю, они хранятся в базе данных информационной системы в отдельной таблице БД.

Использование логина и пароля необходимо, так как в разрабатываемой системе будет несколько групп пользователей. В других мерах защиты информации система не нуждается, так как на самом предприятии предусмотрена парольная система аутентификации пользователей за рабочим местом.

### Защита персональных данных.

Защита персональных данных в Кемеровском областном медицинском колледже регламентируется «Положением о защите персональных данных». В соответствии с действующим положением действуют следующие правила:

1. **Получение персональных данных**
   1. При устройстве на работу работник обязан ознакомиться с действующим «Положением о защите персональных данных» и дать согласие на обработку своих персональных данных в письменном виде.
   2. Работник обязан предоставить колледжу достоверные паспортные данные. При смене паспортных данных, работник должен письменно уведомить об этом колледж в срок, не превышающий 14 дней. Колледж имеет право запросить у работника дополнительные сведения и документы, подтверждающие его личность.
2. **Обработка персональных данных**
   1. Допуск к персональным данным работника разрешен должностным лицам, которым персональные данные необходимы для выполнения конкретных трудовых функций, в частности: начальник отдела кадров и делопроизводства, специалисты по кадрам, инспектор по кадрам, главный бухгалтер, заместитель главного бухгалтера, начальник отдела организации и оплаты труда, бухгалтера, начальник отдела сопровождения системной инфраструктуры, сотрудники тех.поддержки и т.д.
   2. При обработке персональных данных, не связанных с исполнением трудового договора, колледж обязан получить согласие работника на обработку его персональных данных в письменном виде.
   3. Колледж обязан обеспечивать защиту персональных данных от несанкционированного доступа и копирования.
3. **Хранение персональных данных**
   1. Отдел кадров и делопроизводства организует хранение и использование персональных данных работников в соответствии с федеральным законом "О персональных данных" от 27.07.2006 N 152-ФЗ.
   2. Хранение персональных данных работников осуществляется на электронных носителях, а также в бумажном варианте.
   3. Доступ к программному обеспечению, а также к персональной информации, хранящейся на электронных носителях, осуществляется при введении личного идентификатора и пароля пользователя.
   4. Документы персонального характера хранятся в сейфе отдела кадров и делопроизводства.
4. **Уничтожение персональных данных**
   1. Документы, содержащие персональные данные, подлежат хранению и уничтожению в порядке, предусмотренном федеральным законом "О персональных данных" от 27.07.2006 N 152-ФЗ.
   2. Персональные данные работников подлежат уничтожению по достижении целей обработки или в случае утраты необходимости в достижении таких целей.

# Экономическое обоснование разработки ИС

## Расчет затрат на разработку системы

Проектные затраты на разработку системы определяются по формуле 9.1.

 (9.1)

где  – затраты на приобретение оборудования, руб.;

 – заработная плата исполнителям и разработчикам проекта, руб.;

– отчисления на страховые взносы, руб.;

 – затраты на электроэнергию, руб.;

 – накладные расходы, руб.;

 – затраты на амортизацию, руб.;

 – прочие затраты, руб.

1. Затраты на приобретение оборудования:

Т.к. требуемое для развертки приложения оборудование уже есть в наличии, то:

*Зоб = 0*

1. Затраты на заработную плату исполнителям и разработчикам проекта:

Разрабатывается программа проведения проектных работ, определяются ориентировочные трудозатраты этапов работ по исполнителям и стоимость этих затрат (человек-часов) по формуле 9.2

 (9.2)

где  – заработная плата *j*-го исполнителя;

 – ориентировочные трудозатраты *i*-го этапа *j*-го исполнителя в часах;

 – стоимость человека-часа *j*-го исполнителя.

Общие затраты на заработную плату определим по формуле 9.3.

, (9.3)

где *j* = 1…*m* – количество исполнителей.

*Сч\ч = Ом / (Крд \* Драб)*

где *Сч\ч* – стоимость человека-часа, руб.,

*Ом* – месячный оклад, руб.,

*Крд* – количество рабочих дней в месяце (равный двадцати одному),

*Драб* – 8-ми часовой рабочий день.

Средняя заработная плата программиста, занимающегося разработкой приложений составляет 30 000 руб. в месяц. Вычисляем стоимость человеко-часа:

*Сч\ч* = 30000/(21\*8) = 178,57 руб.

 (9.4)

где *Тор* – ориентировочные трудозатраты;

*Дi* – дни, затраченные на выполнение i-ого этапа работы;

*Драб* – 8-ми часовой рабочий день.

Работа ответственного за проект включает в себя следующие этапы, представленные в таблице 10.1:

Таблица 10.1 - Расчет ориентировочных трудозатрат

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Этап работы | Затраченные дни | Ориентировочные трудозатраты чел-час, Top |
| 1 | Постановка задачи | 1 | 8 |
| 2 | Определение хода работы | 4 | 32 |
| 3 | Анализ поставленной задачи |
| 4 | Изучение объекта автоматизации |
| 5 | Составление плана работы |
| 6 | Обзор литературы, детальное изучение объекта | 2 | 16 |
| 7 | Написание общей части дипломного проекта и разработка информационной системы автоматизации | 30 | 240 |
| 8 | Расчет экономиче­ской эффективно­сти | 3 | 24 |
| 9 | Оформление проекта | 7 | 56 |
| итого: | | 47 | 376 |

Произведем расчет заработной платы по формуле 9.2:

Зз/п = *Сч\ч* \**Top=*178,57\*376 = 67142,32 руб.

Таким образом, общие затраты на заработную плату составляют 67142,32 рублей.

1. Отчисления в страховые взносы:

Страховы взносы составляют 30 % от заработной платы.

Зсв = Зз/п \* 0,3 = 95713,52 \* 0,3 = 20142,7 руб.

1. Затраты на электроэнергию:

, (9.5)

где – потребляемая мощность *i*-го токоприемника, кВт/ч;

 – фактическое время работы *i*-го токоприемника, час;

 – цена на электроэнергию за 1 кВт/час, руб.;

 – коэффициент использования по времени *i*-го токоприемника, %.

 = 0,25 кВт/ч;

= 376 часа;

 = 3,04 руб;

 = 0,93.

Подставляя значения в формулу 9.5, получим

Зэн = 0,25 \* 376 \* 3,04 \* 0,93 = 265,76 рублей

5. Накладные расходы:

Накладные расходы принимаются в размере 10 % от заработной платы исполнителей

Зн.р = 67142,32 \* 0,1 =6714,23 руб.

6. Расходы на амортизацию:

Расходы на амортизацию составляют 12,5 % от стоимости оборудования:

Система, развернутая на предприятии использует для функционирования:

* Сервер HP ProLiant ML150 Gen9 776274-421 стоимостью 71 944 рублей;
* Рабочая станция сотрудника отдела тех. обслуживания:
  + ПК DEXP Aquilon O100, стоимостью 9 890 рублей;
  + Монитор ViewSonic VA1903a [VS16216], стоимостью 3 990 рублей;
  + Мышка и клавиатура общей стоимостью 500 рублей.

Цоб = 71944 + 9890 + 3990 + 500 = 86324 рублей.

Зам = Цоб \* 0,125 = 86324 \* 0,125 = 10790,5 руб.

7. Прочие затраты:

Прочие затраты составляют 3 % от суммы всех предшествующих затрат

. (9.6)

 = 0,03 \* (0 + 67142,32 + 20142,7 + 265,76 + 6714,23 + 10790,5) = 3151,67руб.

8. Затраты на этапе проектирования.

Смета проектных затрат приведена в таблице 10.2:

Таблица 10.2 – Таблица затрат

|  |  |  |
| --- | --- | --- |
| пп/п | Наименование затрат | Сумма, руб. |
| 1 | Расходы на оборудование (Зоб ) | 0 |
| 2 | Расходы на заработную плату (Зз/п ) | 67142,32 |
| 3 | Отчисления на страховые взносы (ЗСВ) | 20142,7 |
| 4 | Затраты на электроэнергию (Зэн ) | 378,84 |
| 5 | Накладные расходы (Зн.р) | 6714,23 |
| 6 | Затраты на амортизацию (Зам) | 10790,5 |
| 7 | Прочие расходы (Зпр) | 3151,67 |
|  | Итого | 108320,26 |

Итого затраты на проектирование составят 108320,26 рублей.

## Расчёт эксплуатационных затрат

Затраты на внедрение системы вычисляются по формуле 9.7.

, (9.7)

где  – эксплуатационные затраты, руб.;

 – затраты на оборудование, руб.;

 – затраты на ремонт и содержание оборудования, руб.;

 – затраты на заработную плату обслуживающего персонала, руб.;

 – отчисления на страховые взносы, руб.;

 – расходы на энергию для эксплуатации оборудования, руб.;

 – затраты на амортизацию, руб.;

 – накладные расходы, руб.;

 – прочие расходы, руб.

1. Затраты на оборудование:

, (9.8)

где – цена оборудования, руб.;

 – транспортные расходы (10 % от), руб.;

 – расходы, связанные с монтажом оборудования (8 % от), руб.

 = 0 руб;

 = 0,1\*0= 0 руб;

= 0,08\*0= 0 руб.

Подставляя значения в формулу 9.11, получим:

 = 0 руб.

2. Затраты на ремонт:

Затраты на ремонт и содержание оборудования принимаются в размере 2,5 % от стоимости основных средств

. (9.9)

 = 86324 \* 0,025 = 2158,1 руб.

3. Затраты на заработную плату обслуживающего персонала:

Затраты на заработную плату будут состоять из заработной платы специалиста по ремонту специально предназначенного для обслуживания ЭВМ и оборудования. ЭВМ работает весь рабочий день. Количество выходов в месяц специалиста по ремонту составит: 30 – 9 = 21 день.

Количество дней в году 365, из них 117 – выходные и праздничные дни, 28 – отпуск. Таким образом действительный фонд рабочего времени составляет 220 дней.

Среднюю заработную плату сервисного инженера, обслуживающего оборудование, рассчитаем по формуле 9.10:

 (9.10)

где *D* – годовой должностной оклад работника, руб.;

*К* – коэффициент, учитывающий премиальный (Кпр = 1,4) и районный (Кр = 1,3) коэффициенты.

Годовые затраты на заработную плату составят:

= 23 000 \* 12 \* 1,4 \* 1,3 = 502320 руб.

1. Отчисления в :

Страховы взносы составляют 30 % от заработной платы.

ЗСВ = Зз/п \* 0,3 = 502320 \* 0,3 = 150696 руб. (9.11)

1. Затраты на электроэнергию:

Расходы на электроэнергию рассчитывают по стоимости потребляемой токоприемником электроэнергии

, (9.12)

где *Р* – средняя потребляемая мощность, кВт/час.;

*Т* – время работы токоприемника, час.;

 – коэффициент использования по мощности (0,92–0,95);

 – коэффициент использования по времени (0,92–0,95);

 – цена за электроэнергию за 1 кВт∙час., руб.

*Р =* 0,25 кВт/час;

*Т* = 220\*8 = 1760 часов;

 ≈ 0,93;

 ≈ 0,94;

 = 3.04 руб.

= 0,25\*1760\*0,93\*0,93\*3,04 = 1156,89 руб.

1. Накладные расходы:

Накладные расходы принимаются в размере 10 % от заработной платы обслуживающего персонала.

= 502320 \* 0,1 = 50232 руб.

1. Расходы на амортизацию:

Расходы на амортизацию составляют 12,5 % от стоимости оборудования.

 = Цоб \* 0,125 = 86324 \* 0,125 = 10790,5 руб.

1. Прочие затраты:

Прочие затраты составляют 3 % от суммы всех предшествующих затрат

и рассчитываются по формуле 9.13.

 = 0,03( + + +  +  +  + ) (9.13)

= 0,03 \* (0 + 2158,1 + 502320 + 150696 + 1156,89 + 10790,5 + 50232) = 21520,6 руб.

1. Затраты на этапе эксплуатации.

Смета проектных затрат приведена в таблице 10.3.

Таблица 10.3 – Смета затрат

|  |  |  |
| --- | --- | --- |
| п/п | Наименование затрат | Сумма, руб. |
| 1 | Расходы на оборудование (Зоб) | 0 |
| 2 | Расходы на заработную плату (Зз / п) | 502320 |
| 3 | Затраты на ремонт (Зрем) | 2158,1 |
| 4 | Отчисления на единый социальный налог (ЗСВ) | 150696 |
| 5 | Расход на электроэнергию (Зэн) | 1156,89 |
| 6 | Накладные расходы (Зн.р) | 50232 |
| 7 | Расходы на амортизацию (Зам) | 10790,5 |
| 8 | Прочие расходы (Зпр) | 21520,6 |
|  | Итого | 738874,09 |

Итого затраты на эксплуатацию составят 738874,09 рублей.

## Экономическая эффективность

Экономическая эффективность внедрения системы осуществляется за счет повышения эффективности разработанной ИС: повышения производительности и снижения эксплуатационных затрат по сравнению с существующей системой.

При одинаковой производительности систем ориентировочно годовая экономия составит



где Зэкс.анал.сист. – эксплуатационные затраты аналогичной системы в случае ее приобретения (или существующей); Зэкс.проект.сист. – эксплуатационные затраты проектируемой системы.

Возьмем за аналогичную систему R-Keeper. Стандартная стоимость системы за 6 лицензий за год 120 000 руб. При условии, что затраты на эксплуатацию систем, а именно расходы на заработную плату, единый социальный налог и прочее равны для обеих систем получаем, что годовая экономия равняется стоимости годовой лицензии:

Эгод = Зэкс.анал.сист. + Згод.лиценз. - Зэкс.проект.сист.= 738874,09 + 120 000 - 738874,09 = 120 000 руб.

## Срок окупаемости

Срок окупаемости – период времени, необходимый для того, чтобы доходы покрыли затраты.

Срок окупаемости информационной системы определяется как период времени, в течении которого экономический эффект от эксплуатации системы сравнивается с капитальными затратами на данную систему.

Срок окупаемости определяется по формуле 9.19.

*Т = Зобщ / Эгод* (9.19)

где *Т* – срок окупаемости, лет;

*Зобщ* - суммарные затраты на разработку и внедрение системы, руб.

*Эгод*– годовая экономия, руб

Т = 108320,26*/*120 000 = 0,9 года

Проанализировав экономические показатели, можно сделать вывод о целесообразности разработки информационной системы автоматизации процесса учёта деятельности на комбинате питания и внедрения ее в эксплуатацию, поскольку максимальный срок окупаемости составляет 0,9 года, что не превышаетнормативный срок окупаемости информационных технологий, равный 3 года.

## Технико-экономические показатели проекта

Технико-экономические показатели проекта представлены в таблице 16.

Таблица 10.4. – Технико–экономические показатели проекта

|  |  |  |
| --- | --- | --- |
| Наименование показателя | Единица измерения | Значение показателя |
| Проектные затраты | | |
| 1.1. Затраты на оборудование | руб | 0 |
| 1.2. Заработная плата исполнителям и разработчикам проекта | руб | 67 142,32 |
| 1.3. Отчисления в страховые взносы | руб | 20 142,7 |
| 1.4. Затраты на электроэнергию | руб | 378,84 |
| 1.5. Накладные расходы | руб | 6 714,23 |
| 1.6. Затраты на амортизацию | руб | 10 790,5 |
| 1.7. Прочие затраты | руб | 3 151,67 |
| Эксплуатационные затраты | | |
| 1.8. Затраты на оборудование | руб | 0 |
| 1.9. Затраты на ремонт и содержание оборудования | руб | 2 158,1 |
| 1.10. Затраты на заработную плату обслуживающего персонала | руб | 502 320 |
| 1.11. Отчисления в страховые взносы | руб | 150696 |
| 1.12. Расходы на энергию для эксплуатации оборудования | руб | 1156,89 |
| 1.13. Затраты на амортизацию | руб | 10790,5 |
| 1.14. Накладные расходы | руб | 50232 |
| 1.15. Прочие расходы | руб | 21520,6 |
| 2. Всего затрат | руб | 847 194,35 |
| 3. Экономическая эффективность | руб | 120 000 |
| 4. Срок окупаемости | года | 0,9 |

Таким образом стоимость экономических затрат составила 847194,35 рублей со сроком окупаемости в 0.9 года.

Список используемой литературы

1. Ванеев О.Н. Селезнев В.В. Методические указания по выполнению курсового проекта по дисциплине "Управление данными" для студентов 3 курса специальности 071900 (230201) "Информационные системы и технологии". Кемерово, КузГТУ 2014.
2. Ванеев О.Н. Программа преддипломной практики для студентов специальности 230201 "Информационные системы и технологии". Кемерово, КузГТУ 2015.
3. Ванеев О.Н., Полетаев В.А. Методические указания по дипломному проектированию для студентов специальности 230201 "Информационные системы и технологии". Кемерово, КузГТУ 2015.
4. Ванеев О.Н., Методические указания по выполнению курсового проекта по дисциплине "Проектирование информационных систем" для бакалавров направления 09.03.02 Информационные системы и технологии / О. Н. Ванеев –Кемерово, КузГТУ 2014.
5. Шилдт Г. “C# 4.0 Полный справочник”. Пер. с англ. –М.: ООО “И.Д.Вильямс”, 2011-1056с.
6. Microsoft Developer Network //Microsoft[Электронный ресурс].-URL: <http://msdn.microsoft.com>
7. Хабрахабр [Электронный ресурс].- URL: <http://habrahabr.ru>
8. METANIT.COM [Электронный ресурс].- URL: <https://metanit.com>
9. ProfessorWeb [Электронный ресурс].- URL: https://professorweb.ru
10. ГОСТ Р 51275-2006. «Защита информации. Объект информатизации. Факторы, воздействующие на информацию».

ЗАКЛЮЧЕНИЕ

В процессе выполнения выпускной квалификационной работы решены все поставленные задачи.

Для выявления места разрабатываемой системы в общей деятельности предприятия, уточнения цели разрабатываемой системы был проведен анализ деятельности предприятия, его целей и функций.

Решены вопросы, связанные с:

* Разработкой технического задания;
* Анализом и проектированием;
* Разработкой системы централизованного хранения и обработкой данных;
* Специальной частью;
* Технологии разработки и программная реализация;
* Аппаратная и административная интеграция ИС;
* Общие вопросы администрирования;
* Вопросы информационной безопасности;
* Экономическое обоснование разработки ИС.

Результатом работы является информационная система автоматизации процесса учёта деятельности на комбинате питания ГБПОУ «Кемеровский областной медицинский колледж» г. Кемерово. Система позволяет сократить временные затраты с учетом средств вычислительной техники, а также минимизировать шанс человеческой ошибки.

Разработанная ИС решает задачи конкретного предприятия. В чем и заключается ее основное достоинство по сравнению с альтернативными системами.

ПРИЛОЖЕНИЕ 1. Техническое задание

«ИНФОРМАЦИОННАЯ СИСТЕМА АВТОМАТИЗАЦИИ ДЕЯТЕЛЬНОСТИ НА КОМБИНАТЕ ПИТАНИЯ»

Объект автоматизации – Процесс поставок и составления меню

Краткое название системы - АДНКП

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

На 5 листах

Действует с 16.05.17 г. до 19.06.17 г.

УТВЕРЖДАЮ

Кем утверждается

ФИО

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Личная подпись Расшифровка подписи

16.05.17

УТВЕРЖДАЮ

Студент группы ИТб-132 Тартынский Алексей Валерьевич

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Личная подпись Расшифровка подписи

19.06.17

УТВЕРЖДАЮ

Студент группы ИТб-132 Красильников Павел Витальевич

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Личная подпись Расшифровка подписи

19.06.17

1. **Общие сведения**
2. Полное наименование системы – «Информационная система автоматизации деятельности на комбинате питания для ГБПОУ «Кемеровский областной медицинский колледж» г. Кемерово»

Условное обозначение - «ИСАДНКП»

1. Заказчик системы – «ГБПОУ «Кемеровский областной медицинский колледж» г. Кемерово

Разработчики – студент Тартынский Алексей Валерьевич, студент Красильников Павел Витальевич.

1. Система разрабатывается на основании «Приказа на дипломирование», договора между заказчиком системы и разработчиком.
2. Сроки разработки системы

* 16.05.17 Начало разработки. Разработка технического задания;
* 16.05.17 Утверждение темы диплома;
* 21.05.17 Первый рубежный контроль. Подготовка презентаций по работе;
* 26.05.17 Второй рубежный контроль. Утверждение даты защиты. Предоставление варианта пояснительной записки и версии системы (бета-версии);
* 19.06.17 Окончание разработки. Предоставление подписанных разделов. Готовая версия системы и готовая документация (пояснительная записка, презентации);

1. Финансирование работы по созданию системы не будет осуществляться.
2. Результаты по созданию системы будут предоставляться заказчику в установленные п. 4. сроки. В качестве промежуточных результатов предъявляется:

* Вариант рабочей версии системы с установленной на данной момент функциональностью;
* Рабочая документация, обеспечивающая возможность работы с предоставленным вариантом системы;

Окончательные результаты в виде системы с полным набором функций, утвержденных согласно п. 4, предоставляется 19.06.16

1. **Назначение и цели создания системы**

Назначение системы:

Разрабатываемая система предназначена для автоматизации процессов:

* Создание блюда;
* Создание меню;
* Создание поставок;
* Вычисление стоимости блюд;
* Учет ингредиентов.

Цель разработки системы:

* Автоматизировать процесс создания меню и учета складских поставок;
* Сократить временные затраты на обработку заявок пользователей.

1. **Характеристика объектов автоматизации**

Формирование меню в колледже производится вручную, для этого необходимо постоянное обращение к документам, а также взаимодействие с документоведом и складским персоналом для поддержания актуальной информации.

Из вышесказанного можно сделать вывод, что требуется автоматизация процесса формирования меню и учета складских поставок.

**Требования к системе**

1. Требования системы в целом:
   1. Требования системы к структуре и функционирования.

Компоненты системы должны обеспечивать функционирование системы на существующем аппаратном обеспечении:

* + CPU: Intel® Xeon® E5-2603v3 @ 1.6 Ггц;
  + ОЗУ: 4 ГБ;
  + ОС: Windows Server с установленным MS SQL Server 2012 и IIS;

Функционирование должно обеспечиваться без модернизации.

* 1. Требования к численности и квалификации персонала системы и режиму его работы

Для обслуживания системы не должны привлекаться дополнительные сотрудники.

* 1. Требования к надежности

Максимальное количество сбоев за рабочий день – не более одного раза.

* 1. Требования к безопасности

Требуется разграничение пользователей системы на четыре роли: сотрудник кухни, менеджер, сотрудник склада, администратор. В зависимости от роли, авторизованный пользователь имеет различный доступ к функционалу приложения и данным, фигурирующим в разрабатываемой системе.

* 1. Требования к защите информации от несанкционированного доступа

Учетная запись пользователя на локальной рабочей станции должна быть защищена паролем, удовлетворяющим критерии надежности пароля.

* 1. Требования по сохранности информации при авариях

Системный администратор должен обеспечивать сохранность информации при сбоях подачи электроэнергии, при скачках напряжения, при незапланированных выключениях электроэнергии посредством использования средств резервного копирования и источников бесперебойного питания.

* 1. Требования к патентной чистоте

При разработке системы должны использоваться компоненты, не подпадающие под закон об интеллектуальной собственности. Используемое программное обеспечение должно быть или уже приобретено предприятием – заказчиком или быть свободно распространяемым.

* 1. Требования по стандартизации и унификации

При разработке системы и создании документации должны быть использованы следующие нормативно-технические документы:

1. ГОСТ 34.601-90. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания.

2. ГОСТ 34.602-89 Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы.

3. ГОСТ 34.603-92. Информационная технология. Виды испытаний автоматизированных систем.

4. ГОСТ 19. Единая система программной документации.

5. РД 50-34.698-90. Методические указания. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов.

1. Требования к функциям системы:

Общая характеристика выделяемых функций и их пользователей:

1. В системе выделяются следующие пользователи:

* Сотрудник кухни – доступны функции просмотра и создания меню, требований к меню, формирования меню;
* Сотрудник склада – доступны функции просмотра и создания поставки, формирования поставки, вычисления стоимости ингредиентов;
* Менеджер – доступны функции просмотра, изменения и создания блюд, типов блюд, ингредиентов, вычисления стоимости блюд;
* Администратор – доступны функции назначения прав пользователей.

1. Выделяются следующие функции системы:

* Добавление, изменение, удаление объектов;

1. Требования к информационному обеспечению системы:

Информационное обеспечение системы должно быть создано на основе уже существующих документов с данными, которые необходимы для работы разрабатываемой ИС.

Приложение должно быть разработано с использованием технологии ASP.NET MVC на языке программирования C#.

1. Требование к программному обеспечению:

Система должна работать в среде операционной системы Windows 7 и более поздними версиями;

Система должна иметь установленные компоненты .NET Framework версии 3.5 и выше;

Для разработки системы не должно использоваться дополнительное покупное программное обеспечение.

1. **Состав и содержание работ по созданию системы**

Работы по созданию системы выполняются в три этапа:

* Проектирование. Разработка рабочего проекта. Разработка технического проекта (продолжительность - 20 дней)
* Разработка рабочей приложения (продолжительность - 14 дней).
* Ввод в действие (продолжительность - 1 день).

1. **Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие.**

Для ввода системы в действие, требуется:

* Наличие установленного SQL Server 2012 на сервере предприятия;
* Наличие установленного IIS (Internet Information Server) на сервере предприятия.

1. **Требования к документированию**

* **Проектирование. Разработка эскизного проекта. Разработка технического проекта**.

На этих этапах получаем следующие документы:

* Пояснительная записка к эскизному проекту;
* Пояснительная записка к техническому проекту;
* Схема функциональной структуры.
* **Разработка рабочей документации. Адаптация программ.**

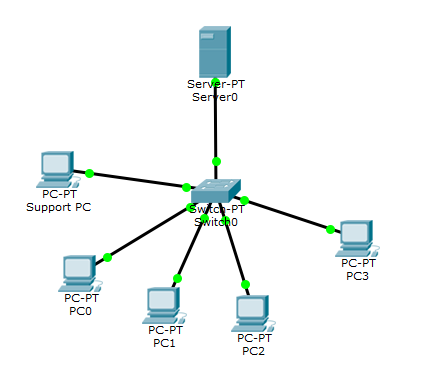
На этих этапах получаем следующие документы:

* Общее описание системы;
* Технологическая инструкция;
* Руководство пользователя;
* Схема базы данных;
* Спецификация;
* Описание конфигурации;
* Текст реализованного программного кода.
* **Ввод в действие.**

На этих этапах получаем следующие документы:

* Акт приемки в опытную эксплуатацию;
* Акт завершения работ.

ПРИЛОЖЕНИЕ 2. Схема ЛВС



ПРИЛОЖЕНИЕ 3. Организационная структура колледжа

ПРИЛОЖЕНИЕ 4. Схема управления организационными структурами КОМК

