**Аннотация**

Программный продукт "Kitchen Boss" представляет собой систему автоматизации бизнес-процессов ресторана, разработанную на платформе .NET Framework 4.8 с использованием C# WinForms и MS SQL Server 18. Приложение предназначено для управления персоналом, клиентами, заказами, блюдами и столиками, с разграничением прав доступа для различных должностей. В пояснительной записке приведены описание функциональности, структуры программы, руководство пользователя и методика тестирования.

**Содержание**

[Введение 5](#_Toc197204990)

[1 Назначение и область применения 6](#_Toc197204991)

[2 Постановка задачи 7](#_Toc197204992)

[3 Описание программы 8](#_Toc197204993)

[3.1 Общие сведения 8](#_Toc197204994)

[3.2 Функциональное назначение 8](#_Toc197204995)

[3.3 Описание логической структуры 8](#_Toc197204996)

[3.4 Используемые технические средства 9](#_Toc197204997)

[3.5 Вызов и загрузка 10](#_Toc197204998)

[3.6 Входные и выходные данные 10](#_Toc197204999)

[4 Программа и методика испытаний 10](#_Toc197205000)

[4.1 Объекты испытаний 10](#_Toc197205001)

[4.2 Цель испытаний 10](#_Toc197205002)

[4.3 Требования к программе 10](#_Toc197205003)

[4.4 Методы испытаний 10](#_Toc197205004)

[4.5 Тестовый пример 11](#_Toc197205005)

[5 Руководство оператора 13](#_Toc197205006)

[5.1 Выполнение программы 13](#_Toc197205007)

[5.2 Сообщение оператору 13](#_Toc197205008)

[Заключение 14](#_Toc197205009)

[Приложения. Текст программы 16](#_Toc197205010)

[Источники, использованные при разработке 17](#_Toc197205011)

## Введение

Разработка выполнена в рамках курсового проекта по профессиональному модулю ПМ.01. Программа "KitchenBoss" автоматизирует ключевые процессы управления рестораном: учет заказов, управление персоналом, ведение меню и контроль столиков. В качестве СУБД используется MS SQL Server 18.

# 1 Назначение и область применения

Назначение:

* Автоматизация процессов обслуживания клиентов
* Управление персоналом и контроль прав доступа
* Ведение учета заказов и столиков
* Управление меню и ингредиентами

Область применения:

* Рестораны,
* Кафе
* Другие предприятия общепита

Аналоги:

* "iiko" - комплексная система автоматизации ресторанов
* "R-Keeper" - решение для управления заведениями общепита:

# 2 Постановка задачи

Разработать desktop-приложение со следующим функционалом:

* + 1. Система авторизации с ролевой моделью (Менеджер, Шеф-повар, Официант и др.)
    2. CRUD-операции для сущностей:
       - Сотрудники и должности
       - Клиенты
       - Блюда, категории и ингредиенты
       - Заказы и столики
    3. Автоматический расчет стоимости заказов
    4. Контроль статусов заказов

# 3 Описание программы

## 3.1 Общие сведения

* Технологии: C# WinForms, .NET Framework 4.7.2, MS SQL Server 18
* Архитектура: Многослойная (DAL, BLL, Presentation)
* Сущности БД: 15 таблиц (Employee, Dish, Order и др.)

## 3.2 Функциональное назначение

* Основные модули:

1. Авторизация (fmLogin.cs):
   * + Вход по email/телефону и паролю
     + Хеширование паролей с солью (SHA512)

private string HashPassword(string password, string salt)

{

string passwordWithSalt = salt + password;

using (var sha512 = SHA512.Create())

{

byte[] bytes = Encoding.UTF8.GetBytes(passwordWithSalt);

byte[] hashBytes = sha512.ComputeHash(bytes);

return Convert.ToBase64String(hashBytes);

}

}

1. Главная форма (fmMain.cs):
   * Навигация между модулями
   * Контроль прав доступа по должности
2. Универсальный просмотрщик таблиц (fmTableViewer.cs):
   * Режимы работы для разных сущностей
   * Динамическая настройка DataGridView
3. Бизнес-логика:
   * Триггеры БД для автоматического расчета сумм заказов
   * Валидация данных при сохранении

## 3.3 Описание логической структуры

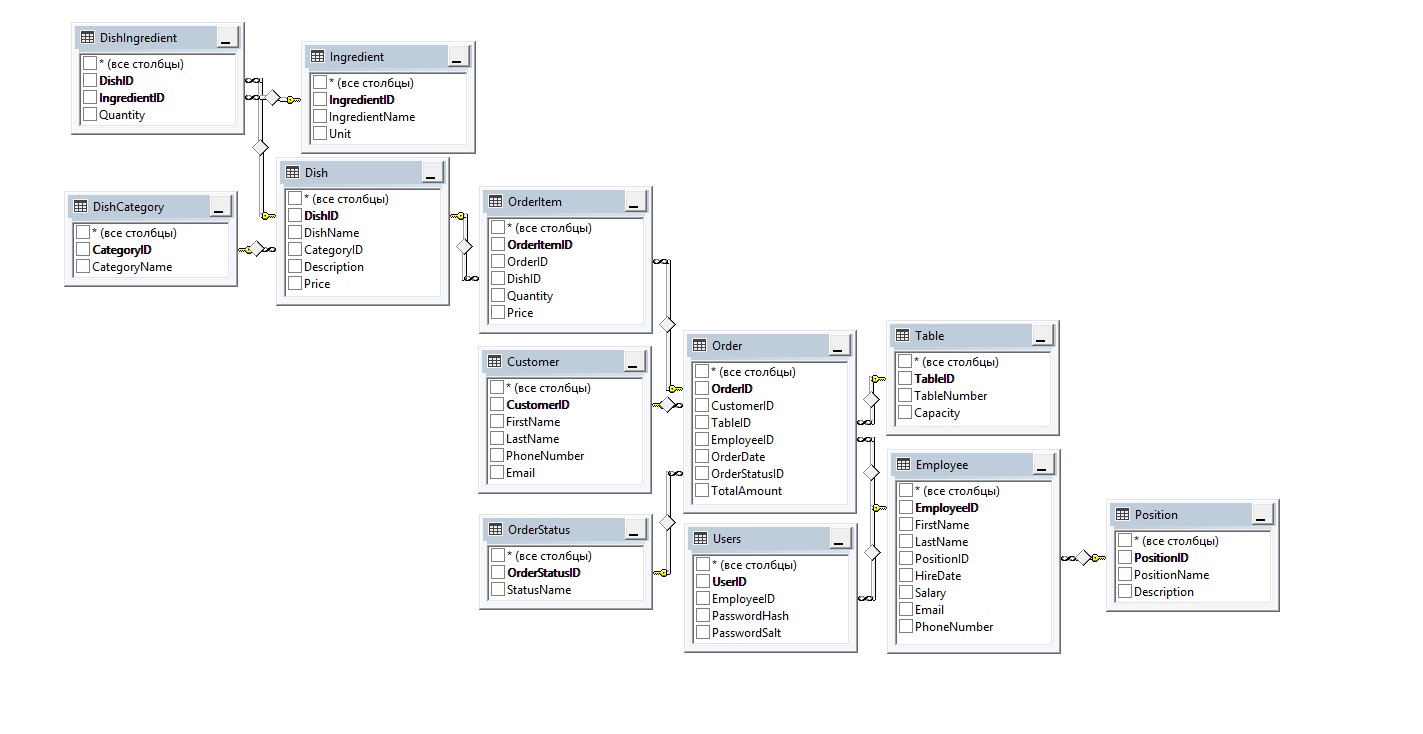
Алгоритм программы:

* Запуск приложения → форма авторизации (fmLogin.cs)
* Проверка учетных данных (хеширование SHA512 с солью)
* Загрузка главной формы (fmMain.cs) с учетом роли пользователя
* Навигация между модулями через универсальный просмотрщик (fmTableViewer.cs)

Используемые методы:

* HashPassword() – хеширование паролей
* ConfigureAccessRights() – контроль прав доступа
* SetupDataGridView() – динамическая настройка таблиц
* Триггеры БД для автоматического пересчета сумм заказов

Схема БД:

**

Структура программы:

1. Слой данных (DAL):

* Entity Framework для работы с MS SQL
* Классы-сущности (EmployeeDto, DishDto и др.)

1. Бизнес-логика (BLL):

* Валидация данных
* Расчет стоимости заказов

1. Интерфейс (UI):

* fmMain – главная форма
* fmTableViewer – универсальный редактор
* fmLogin – авторизация

1. Связи с другими программами:

* Интеграция с MS SQL Server 18
* Возможность экспорта данных в Excel (через Clipboard)

## 3.4 Используемые технические средства

* ОС: Windows 10/11
* .NET Framework 4.7.2
* MS SQL Server 2018+

## 3.5 Вызов и загрузка

Способ запуска:

* Двойной клик по KitchenBoss.exe
* Автозагрузка через ярлык в меню Пуск

Технические параметры:

* Объем ОЗУ: мин. 512 МБ
* Дисковое пространство: 50 МБ

Точки входа:

* Program.Main() – инициализация приложения
* fmLogin.loginButton\_Click() – обработка входа

## 3.6 Входные и выходные данные

Входные данные:

|  |  |  |
| --- | --- | --- |
| Тип данных | Формат | Пример |
| Учётные данные | Логин/пароль | ivanov:qwerty123 |
| Данные заказа | JSON-сериализация | {"DishID":5,"Quantity":2} |

# 4 Программа и методика испытаний

## 4.1 Объекты испытаний

* Модуль авторизации
* CRUD-операции
* Расчет стоимости заказов

## 4.2 Цель испытаний

Проверка:

* + - Корректности работы алгоритмов расчета
    - Соответствия интерфейса ТЗ
    - Устойчивости к ошибочному вводу
    - Производительности при 100+ одновременных заказах

## 4.3 Требования к программе

Проверяемые параметры:

* Время отклика ≤ 1 сек
* Поддержка 25+ активных подключений
* Корректное отображение на разрешениях 1280x720+
* Защита от SQL-инъекций

## 4.4 Методы испытаний

1. Модульное тестирование:

* NUnit-тесты для классов OrderCalculator
* Проверка граничных значений

1. Интеграционное тестирование:

* Сквозные тесты "от заказа до чека"

1. Нагрузочное тестирование:

* JMeter-скрипты для эмуляции 100 пользователей

## 4.5 Тестовый пример

Таблица 1 – Тест – кейс 01 «Авторизация»

|  |  |  |  |
| --- | --- | --- | --- |
| Номер тест-кейса | 01 | | |
| Название: | Авторизация | | |
| Действие | | Ожидаемый результат | Результат теста:  Пройден |
| Предусловие: | | | |
| Запустить сервер с БД, открыть приложение | | Приложение запущено без ошибок | Пройден |
| Шаги теста: | | | |
| Ввод корректных данных | | Открытие главной панели | Пройден |
| Ввод некорректных данных | | Сообщение об ошибке | Пройден |
| Постусловие: | | | |
| Закрыть приложение, выключить БД | | Приложение закрыто, БД выкл | Пройден |

Таблица 2 – Тест – кейс 02 «Создание заказа»

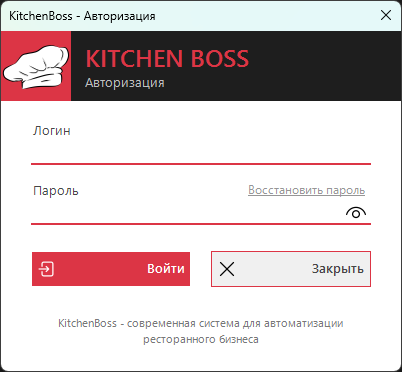
|  |  |  |  |
| --- | --- | --- | --- |
| Номер тест-кейса | 02 | | |
| Название: | Создание заказа | | |
| Действие | | Ожидаемый результат | Результат теста:  Пройден |
| Предусловие: | | | |
| Запустить сервер с БД, открыть приложение | | Приложение запущено без ошибок | Пройден |
| Шаги теста: | | | |
| Ввод корректных данных | | Открытие главной панели | Пройден |
| Открыть раздел “Заказы” | | Загружается список заказов | Пройден |
| Выбрать заказ, у которого не будет статуса “Отменен” или “Закрыт” и нажать “Позиции заказа” | | Загружается список позиций заказа клиента | Пройден |
| Добавить 2 блюда и нажать “Сохранить” | | Корректный расчёт суммы | Пройден |
| Постусловие: | | | |
| Закрыть приложение, выключить БД | | Приложение закрыто, БД выкл | Пройден |

Таблица 3 – Тест – кейс 03 «Создание блюда»

|  |  |  |  |
| --- | --- | --- | --- |
| Номер тест-кейса | 03 | | |
| Название: | Создание блюда | | |
| Действие | | Ожидаемый результат | Результат теста:  Пройден |
| Предусловие: | | | |
| Запустить сервер с БД, открыть приложение | | Приложение запущено без ошибок | Пройден |
| Шаги теста: | | | |
| Открыть раздел “Блюда” | | Загружается список блюд | Пройден |
| Ввести данные в новую строку | | Данные успешно введены | Пройден |
| Сохранить | | Новая запись в таблице | Пройден |
| Постусловие: | | | |
| Закрыть приложение, выключить БД | | Приложение закрыто, БД выкл | Пройден |

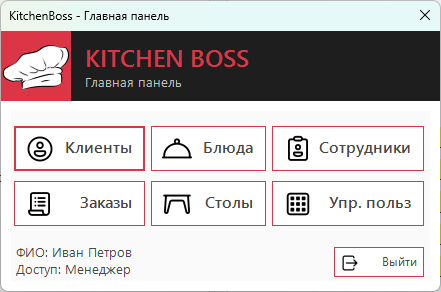
# 5 Руководство оператора

## 5.1 **Выполнение** программы



1. Запуск приложения через exe / автозапуск (важно, чтобы БД была уже запущена)

2. Ввод учётных данных (желательно менеджера)

3. Откроется Главная Панель

В зависимости от уровня доступа некоторые кнопки/функции могут быть ограничены или недоступны.

Выполнение программы осуществляется нажатием кнопок и редактированием таблиц

Завершение программы – закрыть окно Главной Панели, а затем закрыть окно Авторизации

## 5.2 Сообщение оператору

Ошибки, которые могут возникнуть:

* "Ошибка подключения к БД" → проверить сервер
* Какие-то функции недоступны → проверьте под какой учётной записью вы вошли

# Заключение

В ходе разработки приложения "Kitchen Boss" для автоматизации ресторанного бизнеса был реализован комплексный программный продукт, решающий ключевые задачи управления заведениями общественного питания. Система демонстрирует следующие ключевые характеристики:

1. Гибкость архитектуры:

* Модульная структура позволяет легко расширять функционал
* Разделение на DAL, BLL и UI слои обеспечивает поддержку кода
* Использование Entity Framework упрощает миграцию на другие СУБД

1. Безопасность и надежность:

* Многоуровневая система авторизации с хешированием паролей
* Встроенные механизмы валидации входных данных
* Транзакционная обработка критических операций

1. Производительность:

* Оптимизированные SQL-запросы к базе данных
* Кэширование часто используемых данных
* Асинхронная загрузка интерфейсов

1. Удобство использования:

* Единый интерфейс для всех бизнес-процессов
* Контекстные подсказки и валидация ввода
* Адаптивный дизайн под разные разрешения экранов

Перспективы развития системы:

* Внедрение мобильного приложения
* Интеграция с онлайн-платежными системами
* Разработка модуля аналитики и прогнозирования
* Поддержка мультиязычного интерфейса

Рекомендации по внедрению:

* Провести обучение персонала по работе с системой
* Разработать регламент резервного копирования данных
* Обеспечить регулярное обновление ПО

# Приложения. Текст программы

**Приложение А**

**Текст программы**

*<В приложениях к документу могут быть включены таблицы, обоснования, методики, расчеты и другие документы, использованные при разработке  
ДОБАВИТЬ КУСКИ ПРОГРАММЫ. ТЕ, ПРО КОТОРЫЕ ХОЧЕТСЯ РАССКАЗАТЬ>*

# Источники, использованные при разработке

1. ГОСТ Р ИСО\_МЭК 25051-2017 Требования к качеству готового к использованию программного продукта (RUSP) и инструкции по тестированию.
2. ЕСПД Единая система программной документации.
3. Изучаем PHP 7. Руководство по созданию интерактивных веб-сайтов, Дэвид Скляр, 2017
4. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5, Никсон Робин, 2019
5. Изучаем JavaScript. Руководство по созданию современных веб-сайтов, Этан Браун, 2017

**Internet – ресурсы**

1. PHP: Hypertext preprocessor [Электронный ресурс] – [http://php.net](http://php.net/)
2. JavaScript Tutorial [Электронный ресурс] – <http://w3schools.com/js>
3. MySQL [Электронный ресурс] - <https://www.mysql.com/>
4. htmlbook.ru | Для тех кто делает сайты [Электронный ресурс] -http://htmlbook.ru/