Отчёт

Разработка автоматизации контроля сроков годности блюд на C++

Техническое задание

Листов 7

Выполнил работу:

студент гр. ИВТ-24-2б

Шитенков Дмитрий Олегович

Проверил:

Доцент кафедры ИТАС

Полякова Ольга Андреевна

Пермь 2025

**Оглавление**

[1. Введение 3](file:///C:\Users\dimas\OneDrive\Рабочий%20стол\2_5350710466840638107%20(1).docx#_heading=h.1fob9te)

[2. Основания для разработки АРМ 3](file:///C:\Users\dimas\OneDrive\Рабочий%20стол\2_5350710466840638107%20(1).docx#_heading=h.3znysh7)

[3. Назначение АРМ 3](file:///C:\Users\dimas\OneDrive\Рабочий%20стол\2_5350710466840638107%20(1).docx#_heading=h.2et92p0)

4. [Требования к](file:///C:\Users\dimas\OneDrive\Рабочий%20стол\2_5350710466840638107%20(1).docx#_heading=h.3dy6vkm) АРМ 4

5. [UML Диаграммы](file:///C:\Users\dimas\OneDrive\Рабочий%20стол\2_5350710466840638107%20(1).docx#_heading=h.1t3h5sf) 6

6. [Описание используемых инструментов 7](file:///C:\Users\dimas\OneDrive\Рабочий%20стол\2_5350710466840638107%20(1).docx#_heading=h.2s8eyo1)

7. [Источники 8](file:///C:\Users\dimas\OneDrive\Рабочий%20стол\2_5350710466840638107%20(1).docx#_heading=h.17dp8vu)

**1. Введение**

**1.1. Наименование программы**

Наименование программы: “Автоматизация контроля сроков годности блюд на C++”.

**1.2. Краткая характеристика области применения**

Программа предназначена для автоматизированного контроля сроков годности блюд на предприятиях общественного питания (рестораны, кафе, столовые, пищевые производства). Данное решение позволяет:

* Отслеживать сроки годности блюд и ингредиентов в режиме реального времени.
* Автоматически классифицировать продукцию по статусам: «свежее», «застоявшееся», «на грани просрочки», «просроченное».
* Уведомлять сотрудников о приближающемся истечении срока.

**2. Основания для разработки**

**2.1. Основания для проведения разработки**

“Автоматизация контроля сроков годности блюд на C++” разрабатывается в соответствии с творческим заданием по разработке автоматизированного рабочего места специалиста.

**3. Назначение**

Специализированная программа для отслеживания сроков годности блюд в заведениях общественного питания. Программа предназначена для автоматизации процесса контроля за сроками годности продукции, что помогает предотвратить использование просроченных блюд и минимизировать потери.

Основная функциональность программы включает:

* Учет активных блюд с их сроками годности;
* Автоматическое уведомление о приближающихся сроках;
* Управление базой данных блюд;
* Загрузку и обработку партий блюд через CSV файлы;
* Визуальную индикацию состояния блюд через цветовую систему;
* Работу в фоновом режиме с уведомлениями в системном трее.

**Программа особенно полезна для:**

* Ресторанов
* Кафе
* Столовых
* Пищевых производств
* Заведений общественного питания

**4. Требования к АРМ**

**4.1. Функциональные требования**

**4.1.1. Управление каталогом блюд**

* Добавление блюд в каталог:
* Название блюда (обязательно)
* Срок годности (в часах) (обязательно)
* Изображение (опционально)
* Редактирование и удаление блюд из каталога.

**4.1.2. Загрузка и обработка данных**

* Импорт CSV-файла поставки:
* Автоматическое заполнение таблицы блюдами (название, количество, дата поступления).
* Начало отсчета срока годности для каждой партии.
* Загрузка отчета о продажах (CSV):
* Автоматическое списание проданных блюд (по принципу FIFO — первым пришел, первым ушел).
* Обновление остатков в таблице.

**4.1.3. Визуализация и контроль сроков**

* Цветовая индикация статусов в таблице:
* "В норме" (цвет системы)
* "Застоявшееся" (по умолчанию желтый) — приближается срок годности
* "Риск просрочки" (по умолчанию розовый) — близко к истечению
* "Просрочено" (по умолчанию красный)
* Автоматическое изменение цвета строки при смене статуса.

**4.1.4. Уведомления**

* Оповещения на компьютере при смене статуса блюда (например, из "В норме" в "Застоявшееся").
* Настройка времени уведомлений:

1. За сколько часов/дней до смены статуса присылать оповещение.
2. Включение/отключение уведомлений.

**4.1.5. Настройки программы**

* Выбор темы (светлая/темная).
* Настройка цветов для каждого статуса блюд.
* Включение/отключение работы в фоне.

**4.1.6. Дополнительные функции**

* Ручное удаление блюд из таблицы (например, при списании)

**4.2. Требования к интерфейсу**

**Главный экран:**

* Таблица с блюдами (название, количество, дата истечения срока годности, статус).
* Кнопки:
* "Загрузить партию"
* "Каталог"
* "Настройки"
* "Удалить выбранное"
* "Загрузить отчет"

Вкладки настроек:

* "Тема оформления"
* "Уведомления"
* "Цвета состояний"
* "Фоновый режим"

**4.3. Требования к безопасности**

* Валидация CSV-файлов (защита от некорректных данных).
* Автоматическое сохранение данных (резервная копия при закрытии).

**5. UML Диаграммы**



**Центральный класс системы – MainWindow, который:**

1. Является основным управляющим компонентом приложения
2. Координирует работу всех модулей
3. Содержит и управляет данными о блюдах

**Основные взаимосвязи:**

Композиция (целое-часть):

* MainWindow ↔ DishEntry
* MainWindow ↔ DishInfo  
  Объекты DishEntry и DishInfo не могут существовать независимо от MainWindow

Ассоциация (временное взаимодействие):

* MainWindow ↔ DatabaseDialog
* MainWindow ↔ DishDialog  
  Диалоговые окна создаются по требованию и существуют независимо

Агрегация (управление коллекциями):

* DatabaseDialog → DishInfo (управление каталогом)
* DishDialog → DishInfo (редактирование отдельной записи)  
  Объекты DishInfo могут существовать вне контекста диалогов

**6. Описание используемых инструментов**

**Среда разработки:**

* Qt Creator 6.9.0
* Компилятор: MinGW 64-bit
* Язык программирования: C++17

**Программа записи видео:**  
OBS Studio  
**Дополнительные инструменты:**

* Git для контроля версий
* Qt Designer для создания интерфейса

**Зависимости:**

* Qt Core
* Qt GUI
* Qt Widgets
* Qt Network

**6. Используемые библиотеки**  
**Qt Core:**

* QMainWindow
* QDialog
* QDateTime
* QMap
* QVector
* QTimer
* QSettings
* QString

**Qt Widgets:**

* QTableWidget
* QPushButton
* QLabel
* QLineEdit
* QSpinBox
* QGroupBox
* QListWidget
* QMenu
* QSystemTrayIcon

**Qt GUI:** QColor

**7. Источники**  
Qt Widgets: <https://doc.qt.io/qt-6/widgets.html>  
Qt Core: <https://doc.qt.io/qt-6/core.html>  
Qt Forum: <https://forum.qt.io/>