Министерство высшего образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Пермский национальный исследовательский политехнический университет» (ПНИПУ)

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

ОТЧЁТ

по работе

Тема: «ARM Учителя»

Выполнил

Студент группы РИС-24-1б

Конькова С. С.

Проверил доц. Кафедры ИТАС

Полякова О. А.

Пермь 2025

**Оглавление**

[1. Введение 3](#_heading=h.1fob9te)

[2. Основания для разработки АРМ 3](#_heading=h.3znysh7)

[3. Назначение АРМ 4](#_heading=h.2et92p0)

4. [Требования к](#_heading=h.3dy6vkm) АРМ 6

5. [Реализация проекта](#_heading=h.1t3h5sf) 8

6. [Стадии и этапы разработки 9](#_heading=h.2s8eyo1)

7. [UML-диаграмма 10](#_heading=h.17dp8vu)

8. Результаты реализации 11

**1. Введение**

**1.1. Наименование программы**

Полное наименование проекта: «Разработка автоматизированного рабочего места учителя на C++». Далее по тексту – АРМ.

**1.2. Краткая характеристика области применения**  
АРМ предназначено для образовательных учреждений и индивидуального использования преподавателями. Основная цель — автоматизация ключевых процессов учебной деятельности, которая позволит систематизировать и структурировать информацию по учебным темам, учет оценок учащихся, анализ прогресса и выявление проблемных зон в освоении материала, формирование статистических отчетов в удобной форме. Программа позволяет исключить ручную обработку информации, минимизировать ошибки и обеспечить прозрачность процесса обучения, а также быстро адаптироваться к изменениям в расписании и структуре учебного плана.

**2. Основания для разработки АРМ**

**2.1. Основания для проведения разработки**

Проект был инициирован в рамках учебного проектирования с целью продемонстрировать практическое применение теоретических знаний в области программирования и проектирования пользовательских интерфейсов. Задание утверждено доцентом кафедры ИТАС Поляковой О. А.

Согласно требованиям Поляковой О. А., именуемой заказчиком, исполнитель, Конькова С. С., обязана разработать систему не позднее 20.05.2025.

Ключевые причины разработки:

* Потребность в универсальном и простом инструменте для организации учебного процесса на уровне одного преподавателя.
* Отсутствие в рамках школьной ИТ-инфраструктуры решений, адаптированных под локальное, офлайн-использование, не требующее подключения к интернету.
* Стремление к созданию наглядной и интуитивной системы, которую можно использовать без длительного обучения.

**3. Назначение АРМ**

**3.1. Функциональное назначение**

Автоматизированное рабочее место (АРМ) учителя выполняет роль централизованной системы управления учебным процессом на уровне одного преподавателя. Оно предоставляет структурированный, интегрированный набор функций, направленных на обеспечение контроля, анализа и систематизации учебной информации.

В частности, АРМ реализует:

1. Управление содержанием учебного процесса:
   * Возможность добавления новых тем в рамках предмета, назначение им идентификаторов и описаний.
   * Хранение и редактирование информации о дате проведения занятий, пояснительных лекций, тестов и контрольных работ.
   * Отслеживание истории изменений по каждой теме, позволяющее вести ретроспективный анализ преподавания.
2. Работа с индивидуальной и групповой информацией об учениках:
   * Формирование базы данных учащихся с возможностью быстрого поиска, сортировки и редактирования.
   * Назначение индивидуальных оценок за конкретные темы, тесты и занятия.
   * Ведение истории успеваемости учащегося, включая анализ по временным отрезкам и предметам.
3. Аналитика и визуализация успеваемости:
   * Расчёт ключевых метрик, включая средний балл, медиану, количество неудовлетворительных оценок.
   * Групповой анализ: формирование отчётов по всем учащимся, по отдельным темам, а также по периоду (четверть, полугодие).
   * Отображение данных в виде таблиц с цветовой кодировкой, где темы с низким средним баллом (< 3.0) автоматически подсвечиваются, что помогает преподавателю своевременно скорректировать подход к обучению.
4. Упрощение документационного сопровождения:
   * Возможность печати таблиц и экспортирования в сторонние форматы (через перспективную доработку).
   * Создание отчётов по успеваемости в разбивке по ученикам, предметам, датам.
   * Использование автодополнения и шаблонов для ускорения ввода повторяющейся информации.

**3.2. Эксплуатационное назначение**

АРМ является прикладным инструментом, предназначенным для ежедневного использования в учебной практике. Его внедрение приносит пользу как отдельным преподавателям, так и администраторам учебных заведений.

* Для преподавателей:
  + Существенное сокращение времени на обработку и систематизацию учебных данных.
  + Устранение необходимости ведения бумажного журнала.
  + Получение мгновенного доступа к аналитике и прогрессу по каждому ученику и теме.
  + Повышение качества подготовки за счёт своевременной корректировки подходов к подаче материала.
* Для администрации школ и колледжей:
  + Формирование базы для сравнительного анализа классов, преподавателей и предметов.
  + Возможность централизованного хранения данных об успеваемости без зависимости от интернет-соединения.
  + Обеспечение единого стандарта хранения и представления учебной информации.
* Для учащихся (опосредованно):
  + Повышение прозрачности системы оценивания.
  + Более быстрая обратная связь по результатам тестов и занятий.
  + Косвенное стимулирование за счёт отслеживаемости прогресса.

**4. Требования к АРМ**

**4.1. Функциональные требования**

АРМ должно реализовывать все ключевые функции, необходимые для работы преподавателя с учебной информацией. К числу обязательных относятся:

Работа с учебными темами:

* + Создание, редактирование и удаление тем.
  + Привязка дат, описаний и типа занятия (тест, объяснение, контрольная).
  + Поддержка хранения нескольких тестов и объяснений на одну тему.

1. Управление данными об учениках:
   * Добавление и редактирование информации о каждом ученике.
   * Хранение всех оценок с привязкой к теме и дате.
   * Поиск ученика по фамилии, автодополнение при вводе.
2. Статистический и визуальный анализ:
   * Расчёт среднего балла по каждой теме и ученику.
   * Отображение информации в табличной форме.
   * Цветовое выделение слабых результатов (условно: красный — слабое понимание, жёлтый — удовлетворительное, зелёный — хорошее и отличное).
3. Хранение и загрузка данных:
   * Работа с внешними XML-файлами (topics.xml, students.xml) с сохранением структуры и читаемости.
   * Поддержка импорта и экспорта данных без потери информации.
   * Возможность загрузки предыдущих сессий и восстановления прогресса после перезапуска программы.
4. Безопасность и стабильность работы:
   * Встроенная проверка корректности вводимых данных (например, проверка диапазона оценок от 0 до 5).
   * Защита от удаления критичных элементов без подтверждения.
   * Обработка ошибок и недопущение краха программы при некорректных действиях пользователя.

**4.2. Требования к надёжности**

Надёжность — критически важный параметр при работе с учебной информацией, поскольку потеря данных может повлечь за собой серьёзные последствия. Поэтому система должна соответствовать следующим надёжностным характеристикам:

* Механизм автосохранения:  
  Данные должны автоматически сохраняться с заданной периодичностью или при выполнении критичных операций (добавление темы, оценка ученика и пр.).
* Обработка сбоев:  
  В случае некорректного завершения работы программа должна при следующем запуске восстанавливать предыдущую сессию с минимальными потерями.
* Устойчивость к пользовательским ошибкам:  
  Все вводимые данные проходят валидацию: оценки — по диапазону, строки — по формату, даты — по корректности. Система не допускает удаления тем с уже внесёнными результатами без двойного подтверждения.
* Логирование действий (перспективно):  
  В будущих версиях возможна реализация журнала действий для отслеживания изменений, внесённых в базу данных.

**4.3. Технические особенности**

Для реализации функционала были использованы технологии, позволяющие достичь хорошего баланса между производительностью, стабильностью и удобством разработки:

* Платформа: .NET Framework с использованием расширений C++/CLI.
* Интерфейс: Windows Forms — простой, проверенный временем инструмент построения настольных графических приложений.
* Файлы хранения:
  + topics.xml — содержит данные по темам, датам, видам занятий и успеваемости.
  + students.xml — база данных учеников и их оценок.

Каждая запись структурирована, что обеспечивает лёгкость ручного редактирования и возможность миграции в другие системы.

**5. Реализация проекта**

**5.1. Структура программы**

* MyForm.h — главное окно программы, отображающее таблицу тем и основные элементы управления (кнопки, меню).
* Topic.h — модуль, содержащий определения классов и методов для управления учебными темами, хранения и обработки оценок.
* StudentsManager.h — модуль для управления списком учеников, поиска и взаимодействия с оценками.

**5.2. Ключевые функции (фрагменты кода)**

// Создание новой темы

Topic^ t = gcnew Topic();

t->Name = "Математика";

topics->Add(t);

// Добавление оценки ученику

topics[idx]->AddStudentScore("Иванов", 4);

// Вычисление среднего балла

double avg = topics[idx]->GetAverageUnderstanding();

**5.3. Пример интерфейса**

* Основная таблица включает следующие колонки:  
  "№", "Тема", "Количество тестов", "Количество объяснений", "Список учеников", "Среднее понимание темы".
* Элементы управления:  
  "Добавить тему", "Добавить тест", "Внести оценку", "Удалить тему".
* Темы с низким средним баллом (< 3.0) подсвечиваются красным цветом, обеспечивая визуальное выделение проблемных зон.

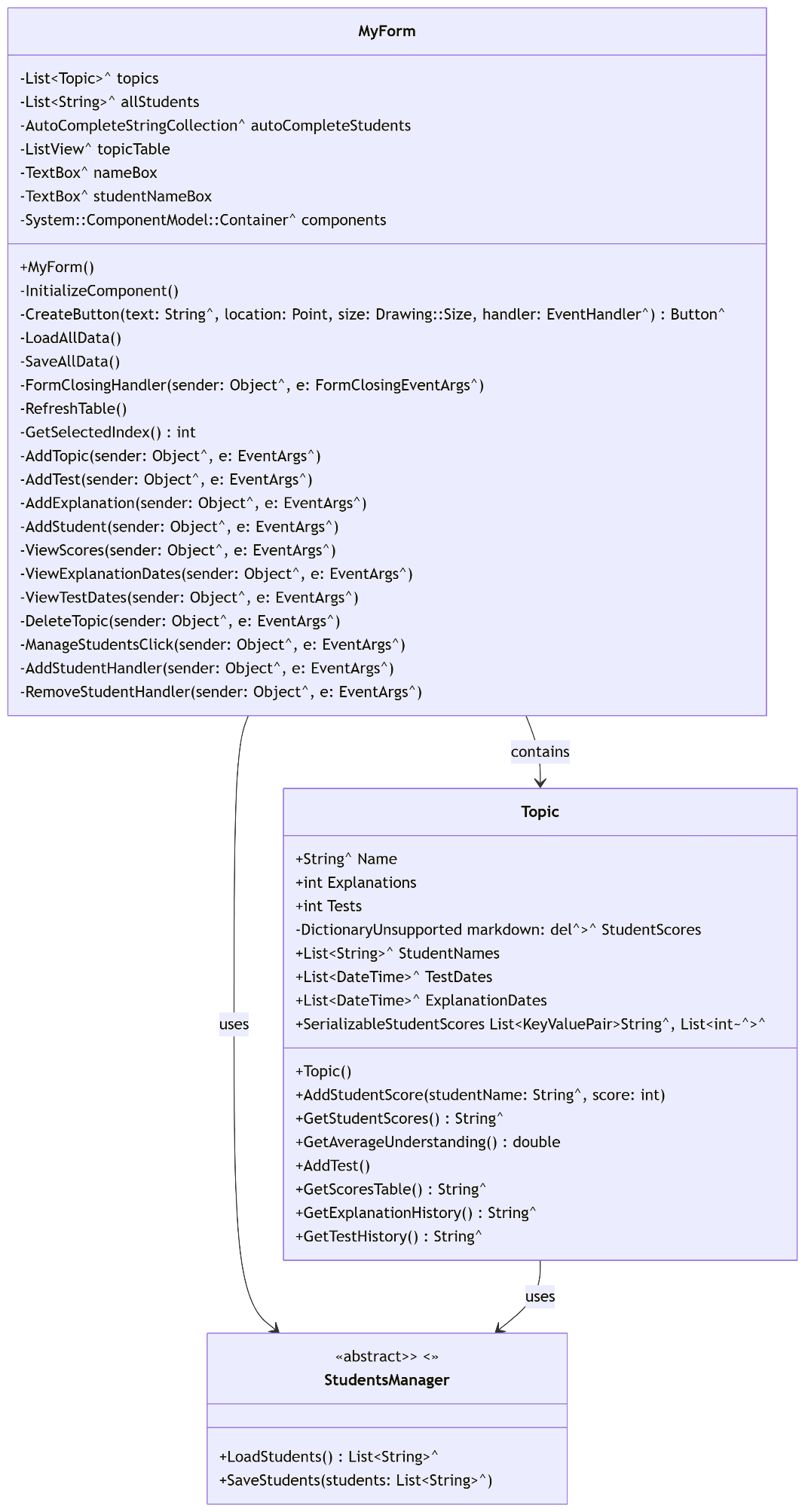
**6. Этапы разработки**

* 1. **Проектирование:**
  + Разработка структуры хранения информации (схемы XML, классы).
  + Определение логики взаимодействия между модулями и пользовательским интерфейсом.
  + Подбор интерфейсных решений, обеспечивающих простоту восприятия.
  1. **. Реализация:**
  + Программирование функций сохранения, загрузки и обработки информации.
  + Имплементация бизнес-логики: вычисления, фильтрации, сортировки.
  + Интеграция с формами Windows Forms, создание откликов на действия пользователя.

**6.3. Тестирование:**

* + Проверка всех функций на устойчивость и корректность при различных сценариях использования.
  + Тестирование граничных условий: пустые списки, максимальные значения, некорректный ввод.
  + Отладка и устранение выявленных логических и синтаксических ошибок.

**7. UML-диаграмма**



**8. Результаты реализации**

В результате была создана работоспособная и стабильная программа, обладающая следующими характеристиками:

* Полный цикл работы с учебными данными: от создания тем и внесения оценок до анализа успеваемости.
* Автоматизация рутинных процессов: вычисление средних значений, генерация таблиц, визуальное представление результатов.
* Интуитивно понятный интерфейс, не требующий обучения для базового использования.
* Надёжность и отказоустойчивость: защита от потери данных, валидация ввода, стабильная работа в офлайн-режиме.

Программа полностью соответствует изначально поставленным задачам и требованиям, успешно прошла тестирование и может быть использована в учебной практике.