

МИНИСТЕРСТВО ОБРАЗОВАНИЯ КИРОВСКОЙ ОБЛАСТИ

Кировское областное государственное профессиональное

образовательное бюджетное учреждение

«Слободской колледж педагогики

и социальных отношений»

КУРСОВАЯ РАБОТА

по профессиональному модулю ПМ.01. «Разработка программных модулей»

на тему:

«Разработка программного модуля

Планировщик задач»

Воробьев Сергей
Андреевич

Специальность 09.02.07 -

Информационные системы
и программирование

Курс 21П-1

Форма обучения: очная

Руководитель:

Пентин Николай Сергеевич

Дата защиты курсовой работы:

Оценка за защиту курсовой работы:

Председатель ПЦК:

Слободской

2024

ОГЛАВЛЕНИЕ

	Стр.
Введение	2
1. Анализ предметной области	3
2. Техническое задание на разработку программного модуля	5
3. Алгоритм и схема функционирования программного модуля.....	7
4. Тестирование программного модуля	10
Заключение.....	11
Список литературы	12

ВВЕДЕНИЕ

Планирование - ключевой фактор успеха в современной жизни. Оно позволяет структурировать задачи, расставлять приоритеты, эффективно распределять время и достигать поставленных целей. В связи с этим, разработка программного модуля "Планировщик задач" является актуальной и востребованной.

Данный проект направлен на создание удобного и функционального инструмента, который позволит пользователям:

- **Создавать и редактировать задачи:** определять название, описание, сроки выполнения, приоритет и другие атрибуты.
- **Группировать задачи:** организовывать задачи по проектам, категориям или другим критериям для лучшей навигации.
- **Настраивать напоминания:** получать оповещения о приближающихся сроках выполнения задач.
- **Отслеживать прогресс:** контролировать выполнение задач, отмечая их как завершённые.
- **Синхронизировать данные:** сохранять задачи на разных устройствах.

1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

Существующие аналоги

На рынке представлено множество программных модулей для планирования задач. Некоторые из наиболее популярных:

- **Todoist:** Онлайн-планировщик с богатым функционалом, включая поддержку проектов, категорий, напоминаний, синхронизации данных.
- **Trello:** Инструмент для совместной работы, который позволяет создавать задачи, организовывать их в списки, назначать исполнителей и отслеживать прогресс.
- **Google Calendar:** Онлайн-календарь с возможностью добавления задач, напоминаний, событий и синхронизации с другими сервисами Google.
- **Microsoft To Do:** Простой планировщик с интеграцией в экосистему Microsoft.

Преимущества и недостатки аналогов

Преимущества:

- **Богатый функционал:** включает в себя различные функции для планирования, управления задачами и совместной работы.
- **Синхронизация данных:** позволяет работать с задачами на разных устройствах.
- **Интеграция с другими сервисами:** обеспечивает тесную связь с другими приложениями и сервисами.

Недостатки:

- **Сложность использования:** многие программы обладают сложным интерфейсом, требующим времени на освоение.
- **Ограничения бесплатной версии:** некоторые функции могут быть доступны только в платной версии.
- **Отсутствие гибкости:** не всегда предоставляют возможность настроить систему в соответствии с индивидуальными потребностями пользователя.

Цель и задачи проекта

Цель

проекта:

Разработка простого и интуитивно понятного программного модуля "Планировщик задач", который будет доступен для всех пользователей, независимо от их опыта работы с планированием.

Задачи проекта:

1. **Разработка модели данных:** определить структуру базы данных, включая таблицы для задач, проектов, категорий и напоминаний.
2. **Создание графического интерфейса:** разработать интуитивно понятный и функциональный интерфейс с возможностью создания, редактирования, удаления и группировки задач.
3. **Реализация функционала:** включить в программу функции создания напоминаний, отслеживания прогресса и синхронизации данных.
4. **Тестирование и отладка:** провести тестирование программы на различных устройствах и отладить возможные ошибки.

2. ТЕХНИЧЕСКОЕ ЗАДАНИЕ НА РАЗРАБОТКУ ПРОГРАММНОГО МОДУЛЯ

Наименование:

"Планировщик задач"

Функциональное назначение:

Предоставление пользователям возможности создавать, редактировать, группировать, удалять задачи, настраивать напоминания и отслеживать прогресс выполнения задач.

Основные функциональные требования:

- **Создание и редактирование задач:**
 - Ввод названия задачи, описания, даты выполнения, приоритета, категории, проекта.
 - Возможность редактирования существующих задач.
- **Группировка задач:**
 - Создание проектов или категорий.
 - Добавление задач в проекты или категории.
 - Фильтрация задач по проектам и категориям.
- **Настройка напоминаний:**
 - Возможность установки времени напоминания.
 - Выбор типа напоминания (звук, уведомление).
- **Отслеживание прогресса:**
 - Пометка задач как завершенных.
 - Визуализация прогресса выполнения задач.
- **Синхронизация данных:**
 - Сохранение данных в облаке или на локальном устройстве.
 - Возможность синхронизации данных на разных устройствах.

Нефункциональные требования:

- **Интерфейс:**
 - Простой и интуитивно понятный интерфейс.

- Графический интерфейс, совместимый с разными операционными системами.
- **Надежность:**
 - Защита данных от потери и несанкционированного доступа.
 - Автосохранение данных для предотвращения потери информации.
- **Эффективность:**
 - Быстрая работа программы, независимо от количества задач.
 - Оптимизация использования ресурсов компьютера.

Условия эксплуатации:

- Программный модуль будет работать на персональных компьютерах и мобильных устройствах с подключением к Интернету или без него.
- Программный модуль должен быть совместим с популярными операционными системами.

Технические требования:

- Язык программирования: C#.
- Фреймворк Visual Studio 2022.
- База данных: SQLite.

Этапы разработки:

1. **Анализ предметной области и проектирование модели данных.**
2. **Разработка графического интерфейса.**
3. **Реализация функциональных требований.**
4. **Тестирование и отладка.**
5. **Документирование.**

3. АЛГОРИТМ И СХЕМА ФУНКЦИОНИРОВАНИЯ ПРОГРАММНОГО МОДУЛЯ

Основные компоненты

- **Модуль ввода данных:** позволяет пользователю вводить информацию о задачах (название, описание, срок выполнения, приоритет, категория, проект).
- **Модуль напоминаний:** отвечает за установку времени напоминаний, их вид (звук, уведомление) и срабатывание в указанное время.
- **Модуль группировки:** предоставляет функции создания проектов и категорий, добавления задач в них и фильтрации по этим параметрам.
- **Модуль отслеживания прогресса:** позволяет отмечать задачи как завершённые и отслеживать прогресс их выполнения.
- **Модуль синхронизации:** обеспечивает сохранение и синхронизацию данных в облаке или на локальном устройстве.

Схема функционирования

1. **Запуск программы:** открывается главное окно с списком задач.

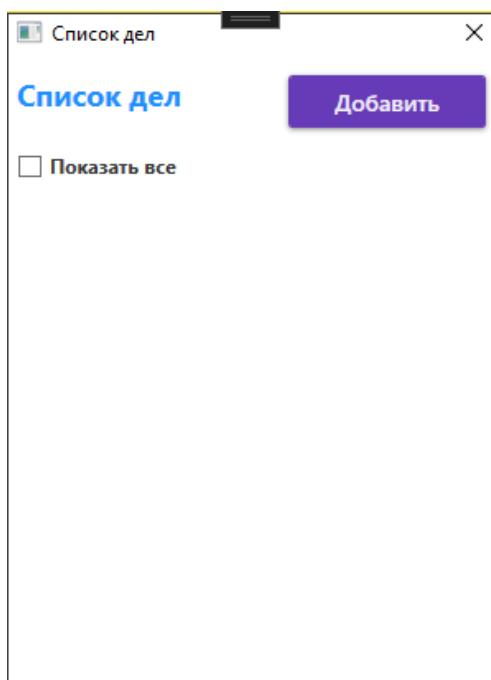


Рисунок 1

2. **Добавление задачи:** пользователь нажимает кнопку "Добавить задачу" и вводит необходимую информацию в специальные поля.

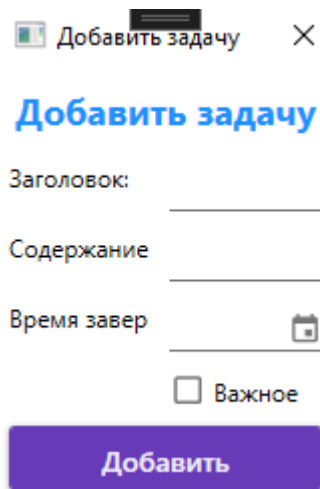


Рисунок 2

3. **Группировка задач:** пользователь может создать новые проекты или категории и добавить в них задачи.



Рисунок 3

4. **Настройка напоминаний:** пользователь устанавливает время и тип напоминания для каждой задачи.

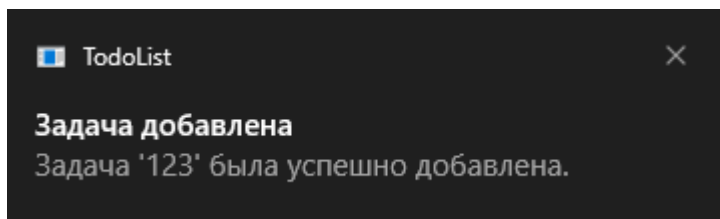


Рисунок 4

5. **Отслеживание прогресса:** пользователь отмечает задачи как завершённые, и программа отображает прогресс выполнения.
6. **Синхронизация данных:** программа синхронизирует данные с облачным хранилищем или локальным устройством.

Алгоритмы

- **Создание задачи:**
 - Получение данных от пользователя (название, описание, срок, приоритет, категория, проект).

- Создание нового объекта задачи с полученной информацией.
- Добавление задачи в базу данных.
- Отображение задачи в списке задач.
- **Группировка задач:**
 - Создание нового объекта проекта или категории с названием.
 - Добавление задач в проект или категорию.
 - Обновление базы данных.
- **Настройка напоминаний:**
 - Получение данных от пользователя (время напоминания, тип напоминания).
 - Создание нового объекта напоминания с полученной информацией.
 - Добавление напоминания в базу данных.
 - Запуск таймера срабатывания напоминания.
- **Отслеживание прогресса:**
 - Получение информации о выполнении задачи от пользователя.
 - Обновление статуса задачи в базе данных.
 - Отображение прогресса выполнения задач.
- **Синхронизация данных:**
 - Сохранение данных в облачное хранилище или на локальном устройстве.
 - Загрузка данных из облачного хранилища или с локального устройства.

4. ТЕСТИРОВАНИЕ ПРОГРАММНОГО МОДУЛЯ

Типы тестов:

- **Функциональные тесты:** проверка корректности выполнения всех функций программы (создание, редактирование, удаление задач, настройка напоминаний, группировка задач, отслеживание прогресса, синхронизация данных).
- **Нагрузочные тесты:** проверка работоспособности программы при большом количестве задач, напоминаний и пользователей.
- **Тесты безопасности:** проверка защищенности данных от несанкционированного доступа и изменений.
- **Тесты интерфейса:** проверка удобства и интуитивности интерфейса программы.

Этапы тестирования:

1. **Модульное тестирование:** проверка корректности работы отдельных компонентов программы.
2. **Интеграционное тестирование:** проверка взаимодействия разных компонентов программы между собой.
3. **Системное тестирование:** проверка работы программы как целого на разных устройствах и в разных условиях.
4. **Приемочное тестирование:** проверка соответствия программы требованиям технического задания.

ЗАКЛЮЧЕНИЕ

В данной курсовой работе представлен проект программного модуля "Планировщик задач", который позволит пользователям эффективно организовывать свои задачи, отслеживать их выполнение и достигать поставленных целей. В проекте учтены основные функциональные и нефункциональные требования к программе, а также представлены алгоритмы работы и схема функционирования. Разработка такого инструмента является актуальной задачей в современном мире, где время становится все более ценным ресурсом.

СПИСОК ЛИТЕРАТУРЫ