**3. ОПИСАНИЕ ПРОГРАММЫ**

*(Сведения о логической структуре и функционировании программы)*

**3.1. Общее описание**

Приложение «Будильник» реализовано с использованием Windows Forms на языке C# и применяет библиотеку NAudio для воспроизведения аудиофайлов формата MP3. Основная задача программы – обеспечить автоматическое воспроизведение выбранного аудиофайла в указанное время, заданное пользователем через текстовые поля (часы, минуты, секунды). Программа использует таймер, который с периодичностью в 500 мс проверяет системное время и сравнивает его с введёнными значениями. При совпадении запускается процесс воспроизведения выбранного аудиофайла.

**3.2. Логическая структура**

Приложение условно разбито на следующие основные модули:

* **Пользовательский интерфейс (UI):**
  + Отвечает за ввод данных пользователем (три текстовых поля для ввода времени, кнопки для выбора песни и запуска/остановки будильника).
  + Отображает текущую информацию, включая автоматически заполненное текущее время при запуске.
* **Модуль управления временем и таймер:**
  + Использует компонент Timer с интервалом 500 мс, чтобы периодически получать системное время.
  + Выполняет сравнение введённого времени и системного времени в форматированном виде (с ведущими нулями).
* **Модуль воспроизведения аудио (на базе NAudio):**
  + Обеспечивает воспроизведение выбранного MP3-файла с помощью классов WaveOutEvent и AudioFileReader.
  + Содержит методы инициализации, остановки и повторного запуска аудиоплеера.
* **Модуль валидации и форматирования данных:**
  + Обрабатывает события ввода (KeyPress) и потери фокуса (Leave) в текстовых полях, чтобы разрешить только числовой ввод и гарантировать формат чисел с двумя цифрами (например, «03» вместо «3»).

**3.3. Функционирование программы**

При запуске приложения:

1. **Инициализация UI:**
   * Форма загружается в центре экрана с заданными размерами (800×600), без возможности изменения размера и разворачивания.
   * В текстовые поля автоматически записывается текущее системное время, отформатированное с ведущими нулями.
2. **Ввод пользователем времени будильника:**
   * Пользователь может изменить значения часов, минут и секунд. Для каждого поля реализована проверка ввода (разрешаются только цифры, максимум — два символа).
3. **Выбор аудиофайла:**
   * По нажатию кнопки «Выбрать песню mp3» открывается диалог выбора файла, после чего путь к файлу сохраняется и отображается на кнопке (с укороченным именем).
4. **Запуск будильника:**
   * При нажатии кнопки «Запустить» происходит валидация заполненности полей и корректности введённых значений (например, часы от 0 до 23, минуты и секунды от 0 до 59).
   * При успешном запуске поля становятся недоступными для редактирования, а запускается таймер, который регулярно сравнивает системное и заданное пользователем время.
5. **Срабатывание будильника:**
   * Когда системное время совпадает с введённым временем (все три компонента: часы, минуты, секунды), вызывается модуль воспроизведения MP3-файла.
   * Запускается аудиопроигрыватель на базе NAudio, после чего пользователю предоставляется возможность остановить воспроизведение.
6. **Управление воспроизведением:**
   * Кнопка, отвечающая за выбор песни, в режиме воспроизведения преобразуется в кнопку «Выключить музыку», позволяющую остановить проигрывание.

**4. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

*(Схема алгоритма, общее описание алгоритма и обоснование принятых решений)*

**4.1. Схема алгоритма**

Ниже представлен псевдокод – блок-схема работы программы:

scss

КопироватьРедактировать

[Старт]

│

▼

[Загрузка формы]

│ — Установка окна (800×600, центр экрана)

│ — Автоматическое заполнение текстовых полей текущим временем

▼

[Ожидание ввода пользователя]

│ — Пользователь меняет время (Ч, М, С)

│ — Пользователь выбирает MP3-файл через FileDialog

▼

[Нажатие кнопки "Запустить"]

│ — Валидация введённых данных

│ — Запись введённых значений в переменные

│ — Запуск таймера (интервал 500 мс)

▼

[Таймер Tick (каждые 500 мс)]

│ — Получение системного времени

│ — Форматирование времени (добавление ведущих нулей)

│ — Сравнение текущего и заданного времени

▼

[Время совпало?] --- Нет ---> [Возврат к таймеру]

│ Да

▼

[Воспроизведение MP3-файла через NAudio]

│ — Остановка таймера (или блок повторного срабатывания)

│ — Изменение надписей кнопок (например, "Выключить музыку")

▼

[Ожидание дальнейших действий оператора]

│

▼

[Остановка или перезапуск по команде оператора]

│

▼

[Конец]

**4.2. Общее описание алгоритма**

Программа реализует идею простого будильника с использованием циклической проверки системного времени. При запуске приложения инициализируются базовые элементы: окно с предопределёнными размерами, текстовые поля для ввода времени и кнопки для управления.  
Пользователь имеет возможность настроить время срабатывания и выбрать аудиофайл для воспроизведения. После ввода и нажатия кнопки «Запустить» происходит переход в режим контроля времени с использованием таймера. Таймер каждые 500 мс сравнивает введённое время с системным. При совпадении запуск производится через модуль NAudio, что позволяет воспроизвести выбранный MP3-файл.  
Логика работы предусматривает возможность останова музыки, а также блокировку редактирования полей после запуска будильника, чтобы избежать ошибок при сравнении.

**4.3. Обоснование принятых решений**

* **Использование Windows Forms:**  
  Выбор Windows Forms обусловлен простой реализацией графического интерфейса, богатым набором встроенных компонентов (Label, TextBox, Button, Timer) и быстрой сборкой приложений для ОС Windows.
* **Применение библиотеки NAudio:**  
  В случае отсутствия компонента Windows Media Player (WMPLib) библиотека NAudio позволяет воспроизводить аудиофайлы формата MP3, имеет открытый исходный код и широко используется в сообществе разработчиков .NET.
* **Периодическая проверка времени:**  
  Применение таймера с интервалом в 500 мс обусловлено желанием максимально точно уловить момент, когда введённое время совпадает с системным, без значительной нагрузки на процессор.
* **Валидация ввода:**  
  Ограничение ввода с помощью событий KeyPress и Leave гарантирует корректное форматирование данных и исключает возможность ввода некорректных значений (например, однозначных чисел, которые должны отображаться с ведущим нулем).
* **Модульная архитектура:**  
  Разделение функциональности по модулям (интерфейс, таймер, аудиовоспроизведение) облегчает сопровождение, тестирование и последующее расширение функционала приложения.

**5. ЭКСПЛУАТАЦИОННЫЕ ДОКУМЕНТЫ**

Программная документация включает в себя документы для обеспечения правильного применения, настройки, обслуживания и эксплуатации продукта.

**5.1. Описание применения**

**5.1.1. Назначение программы**

Приложение «Будильник» предназначено для автоматического воспроизведения аудиофайлов (MP3) в заданное время, что позволяет пользователю организовать свой день и вовремя получать напоминания о важных событиях. Ключевые возможности:

* Настройка времени срабатывания (часы, минуты, секунды).
* Выбор MP3-файла для воспроизведения.
* Автоматическая проверка системного времени и воспроизведение звука при совпадении.

**5.1.2. Область применения**

Приложение может использоваться:

* В бытовых условиях для установления напоминаний о встречах, будильниках и расписаниях.
* В корпоративных системах, где требуется автоматическое оповещение персонала.
* В учебных заведениях для организации расписания занятий или перерывов.

**5.1.3. Условия применения**

* **Технические требования:**  
  — Компьютер с ОС Windows (рекомендуется Windows 10 или выше).  
  — Минимальные системные характеристики: процессор 1 ГГц, 512 МБ ОЗУ, 100 МБ свободного места.  
  — Установленная среда выполнения .NET Framework (например, 4.7.2 или 4.8).
* **Программные требования:**  
  — Библиотека NAudio, установленная через NuGet.
* **Ограничения:**  
  — Приложение предназначено для работы в оффлайн режиме.  
  — Интерфейс рассчитан на стандартное разрешение экрана.

**5.1.4. Описание задачи и методов решения**

* **Задача:** Реализовать программное средство, которое будет воспроизводить MP3-файл в заданное пользователем время.
* **Методы решения:**  
  — Использование Windows Forms для создания интуитивного интерфейса.  
  — Применение таймера для циклической проверки системного времени.  
  — Использование библиотеки NAudio для воспроизведения MP3-файлов.  
  — Реализация событий для проверки ввода и управления состоянием программы.

**5.1.5. Входные и выходные данные**

* **Входные данные:**  
  — Значения времени (часы, минуты, секунды), вводимые в текстовые поля.  
  — Путь к выбранному MP3-файлу, выбранный через диалог выбора файлов.
* **Выходные данные:**  
  — Воспроизведение аудиофайла при совпадении системного времени с заданным.  
  — Изменение состояния элементов управления (блокировка полей, изменение текста кнопок).  
  — Сообщения об ошибках и уведомления оператора через MessageBox.

**5.2. Руководство системного программиста**

**5.2.1. Назначение и условия применения**

Документ рассчитан на системного программиста, ответственного за установку, настройку и обеспечение корректного функционирования приложения в рабочей среде.

**5.2.2. Основные сведения**

* **Техническая конфигурация:**  
  — Проверьте наличие среды выполнения .NET Framework, соответствующей требованиям проекта.  
  — Установите пакет NAudio через NuGet.
* **Настройка системы:**  
  — Расположите исполняемый файл в защищённом каталоге (при необходимости используйте антивирусную и брандмауэрную защиту).  
  — Организуйте резервное копирование исходных кодов и документации с использованием системы контроля версий.

**5.2.3. Рекомендации по диагностике**

* **Логирование ошибок:**  
  — При возникновении ошибок в работе приложения рекомендуется добавить системное логирование (при необходимости – использовать трассировщик).
* **Тестирование:**  
  — Проведите модульное тестирование всех основных функций (проверка таймера, воспроизведение MP3, валидация ввода).  
  — Организуйте автоматические тестовые сценарии для проверки стабильности работы при различных входных данных.

**5.2.4. Обслуживание и обновление**

* **Периодическая проверка работоспособности:**  
  — Убедитесь, что системные настройки и зависимости (например, .NET Framework и NAudio) обновлены до актуальных версий.  
  — При появлении новых требований или обнаружении ошибок – оперативно вносите изменения в исходный код.

**5.3. Руководство программиста**

**5.3.1. Назначение и условия применения**

Данный документ предназначен для программиста, занимающегося поддержкой, доработкой и эксплуатацией приложения.

**5.3.2. Характеристики программы**

* **Основные функции:**  
  — Считывание и форматирование системного времени.  
  — Воспроизведение аудиофайлов через NAudio.  
  — Реализация событий ввода, потери фокуса и нажатия кнопок.
* **Режимы работы:**  
  — Режим ожидания ввода и настройки.  
  — Режим активного контроля времени с циклическим опросом через таймер.  
  — Режим воспроизведения аудио при срабатывании будильника.

**5.3.3. Обращение к программе**

* **Передача параметров:**  
  — Ввод пользователем осуществляется через текстовые поля на форме.  
  — Выбор файла производится через диалоговое окно OpenFileDialog.
* **Входные и выходные данные:**  
  — Формат входных данных – строковое представление числовых значений времени (с форматированием до двух цифр).  
  — Выход – звуковой сигнал, изменение состояния контролов, отображение сообщений об ошибке.

**5.3.4. Сообщения и диагностика**

* **Сообщения об ошибках:**  
  — В случае некорректного ввода или выбора файла, программа уведомляет пользователя через MessageBox.  
  — При ошибках воспроизведения аудио выводятся диагностические сообщения с описанием исключения.

**5.4. Руководство оператора**

**5.4.1. Назначение программы**

Документ содержит сведения, достаточные для понимания функций программы, а также пошаговые инструкции по её эксплуатации.

**5.4.2. Условия выполнения программы**

* **Минимальные технические требования:**  
  — Компьютер с ОС Windows, установленным .NET Framework и пакетами, указанными в техническом задании.
* **Подготовка к запуску:**  
  — Запустите приложение через значок ярлыка или меню Пуск.  
  — После загрузки формы убедитесь, что текущие значения времени отображаются в текстовых полях.

**5.4.3. Пошаговое выполнение**

1. **Выбор времени:**  
   — Проверьте, что поля с часами, минутами и секундами заполнены корректно (при необходимости отредактируйте их).
2. **Выбор аудиофайла:**  
   — Нажмите на кнопку «Выбрать песню mp3», выберите нужный MP3-файл через появившийся диалог и дождитесь отображения выбранного файла на кнопке.
3. **Запуск будильника:**  
   — Нажмите кнопку «Запустить». В этот момент поля становятся недоступны для редактирования, и запускается таймер.
4. **Ожидание срабатывания:**  
   — При совпадении заданного и системного времени программа воспроизводит аудио и меняет статус кнопок для возможности остановки.
5. **Остановка:**  
   — Если требуется остановить будильник до завершения воспроизведения, нажмите кнопку «Стоп».

**5.4.4. Сообщения оператора**

* При ошибочном вводе данных или выборе неверного файла оператор получает соответствующие уведомления через стандартное окно сообщений (MessageBox).
* В случае системных сбоев рекомендуется связаться с технической поддержкой.

**5.5. Описание языка**

**5.5.1. Язык программирования**

Приложение реализовано на языке **C#** с использованием технологии **Windows Forms**.

* **Синтаксис:**  
  — Основная структура программы построена на объектах и классах.  
  — Для управления пользовательским интерфейсом используется модель событий и обработчиков событий.
* **Семантика:**  
  — Объектно-ориентированный подход позволяет разделять функциональные блоки (UI, логика, аудио).  
  — Использование пространства имён (например, System, System.Windows.Forms, NAudio.Wave) позволяет организовывать код и облегчает его сопровождение.

**5.5.2. Комментарии и документация**

* Код снабжен XML-комментариями, поясняющими назначение методов и переменных.
* Дополнительное документирование реализовано в виде пояснительной записки и эксплуатационных документов.

**5.6. Руководство по техническому обслуживанию**

**5.6.1. Назначение**

Документ предназначен для специалистов, отвечающих за диагностику, тестирование и периодическое обслуживание как программного обеспечения, так и технических средств.

**5.6.2. Технические меры**

* **Мониторинг работы:**  
  — Регулярное тестирование работоспособности таймера и модуля воспроизведения аудио.  
  — Использование журналов (логов) для записи ошибок (при необходимости, дополнить систему логирования).
* **Технический осмотр:**  
  — Проведение проверки установленных версий .NET Framework и обновлений пакетов NuGet (NAudio).  
  — Убедитесь, что аппаратное обеспечение соответствует минимальным системным требованиям.
* **Использование тестовых программ и диагностических утилит:**  
  — Программист или технический специалист может применять встроенные средства отладки Visual Studio для локализации и устранения ошибок.  
  — Рекомендуется периодически проводить стресс-тестирование таймера и модуля аудиовоспроизведения.

**5.6.3. Процедуры технического обслуживания**

* **Обновление:**  
  — Обеспечьте резервное копирование исходного кода и сопутствующей документации перед внесением любых изменений.  
  — Обновляйте компоненты приложения, если появляются новые версии библиотек или фреймворка.
* **Диагностика и устранение неполадок:**  
  — При возникновении ошибок анализируйте сообщения, выводимые MessageBox, и логи (если реализованы).  
  — При необходимости обращайтесь к документации Microsoft по технологиям Windows Forms и C#.

**5.6.4. Рекомендации по обеспечению надежности**

* Регулярно проводите аудит кода, чтобы выявить потенциальные проблемные места и оптимизировать работу таймера и аудио-модуля.
* Организуйте тестовый цикл для проверки работы в различных конфигурациях аппаратного обеспечения.
* Обеспечьте достаточное резервное копирование как исходных кодов, так и данных о конфигурации системы.