**ЛАБОРАТОРНЫЙ ПРАКТИКУМ   
МОДУЛЯ 1 «ВВЕДЕНИЕ В ПРОБЛЕМУ ЧЕЛОВЕКО-МАШИННОГО ВЗАИМОДЕЙСТВИЯ»**

**по дисциплине «Проектирование человеко-машинных интерфейсов» для подготовки бакалавров по ООП НИУ «Программное обеспечение автоматизированных систем и вычислительных комплексов» – профиль «Программные технологии распределенной обработки информации**»

**Лабораторная работа №2**

Тема: **«Реализация диалогов в графическом пользовательском интерфейсе по технологии WIMP».**

**Цель:** закрепление теоретических знаний и формирование практических умений и навыков использования инструментальных сред программирования для реализации диалогов в графическом пользовательском интерфейсе по технологии WIMP

**Постановка задачи**

1 На основе абстрактного графа диалога системы разработать меню MDI-интерфейса.

2 Использовать меню различных видов: основе, панели инструментов, контекстное и кнопочное. Как альтернативу использовать директивную форму диалога, поставив в соответствие основным командам комбинации клавиш.

3 Предусмотреть возможность управления меню клавиатурой.

4 В соответствии с эргономическими принципами проектирования программных средств определить состав, внешний вид и взаимное расположение визуальных интерфейсных компонентов.

5 Используя инструментальные среды программирования графического диалога, разработать программное средство, содержащее элементы управления в многооконных интерфейсах.

6 Распечатать макеты экранов с описанием функциональности каждого интерфейсного элемента.

7 Запрограммировать различные реакции программного средства на действия пользователя.

**1 Меню интерфейса**

# Для обеспечения удобства использования программы, меню должно иметь интуитивно-понятный интерфейс, который не будет вызывать особых трудностей у неподготовленного пользователя. Меню программы должно соответствовать правилу шести, т.е. в одну линейку меню не должно включаться более шести пунктов, каждый из которых не должен содержать более шести опций.

Также, в программе должны быть предусмотрены различные подсказки или пояснения, облегчающие использование программы. В программе должна быть предусмотрена справка, которая подробно описывает принципы работы с программой.

На рисунке 1 представлена структурная схема программы, согласно этой схеме, и абстрактным графам диалога, представленных на рисунках 2 – 5, разработано меню интерфейса программы, показанное на рисунке 6

Меню

## Файл

## Правка

## Отменить

## Вырезать

## Удалить

## Вставить

## Копировать

## Найти

## Выход

## Справка

## Файл

## Отступы

## Перенос по словам

## Шрифт

## Создать

## Открыть

## Сохранить

## Сохранить как

## Печать

## Выделить все

Рис. 1 – Структурная схема программы

**2 Меню различных видов**

1. Основное меню:



Рисунок 1 – Основное меню

1. Контекстное меню

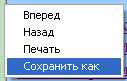


Рисунок 2 – Контекстное меню

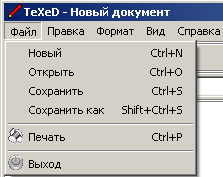
3 Кнопочное меню:



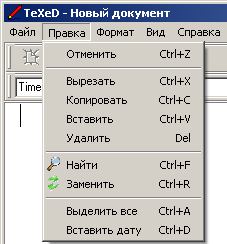
Рис. 3 – Кнопочное меню

**3. Предусмотреть возможность управления меню клавиатурой**

Предусмотрена возможность управления меню клавиатурой, а также возможность использовать сочетания клавиш для быстрого доступа к функциям. Во вкладках основного меню устанавливается соответствие между функциями и сочетаниями клавиш, рисунках 4-5.



**Рис. 4 – Функциональные клавиши меню «Файл»**



**Рис. 5 - Функциональные клавиши меню «Правка»**

1. **В соответствии с эргономическими принципами проектирования определить состав, внешний вид и расположение визуальных интерфейсных компонентов**

Внешний вид и взаимное расположение визуальных интерфейсных компонентов, показанный на эскизах интерфейса пользователя показан на рис. 6 – 7

|  |
| --- |
| Главное окно |
| Меню |
| Кнопочное меню |
| Рабочая область |
| Строка состояния |

**Рис. 6 – Главная форма создания задач**

|  |  |  |
| --- | --- | --- |
| Поиск | | |
|  | | |
| Текст для поиска |  |  |
|  |  |
|  | | |
| ОК | Отмена | |

**Рис. 7 - Форма поиска**

1. **Используя инструментальные среды программирования графического диалога, разработать программное средство, содержащее различные элементы управления**

Инструментальные средства современных IDE, предлагают программисту или человеку, занимающемуся оформлением интерфейсов пользователя довольно много видов различных элементов управления, позволяющих пользователю данного программного средства осуществлять навигацию по графическому интерфейсу программы различными способами, наиболее удобными тому или иному пользователю.

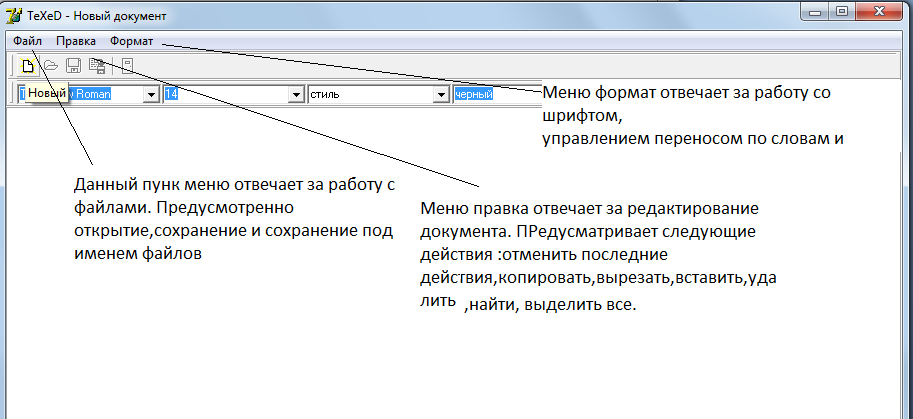
Многообразие элементов управления имеет за собой несколько функций, которые положительно влияют на восприятие графического интерфейса программы пользователем, во-первых, это функциональное разнообразие программы, т.е. множество функций, которое можно реализовать с помощью тех или иных элементов управления, а во-вторых, это возможность создать свой непосредственный стиль приложения на основе различного вида форм элементов управления, которые в избытке существуют во всяческих библиотеках компонент.

Спецификация типовых объектов управления графического интерфейса представлена в таблице 1

Таблица 1 - Спецификация типовых объектов управления графического интерфейса

|  |  |
| --- | --- |
| Название | Описание |
| меню | выбор опций работы программы по пунктам |
| кнопка | выполнение действия по заранее описанному алгоритму |
| флажок | выбор одного или нескольких значений сразу |
| текстовое окно | служит для отображения текстовой информации произвольного вида |
| окно-список | содержит список альтернативных значений |
| метка | постоянный текст, не подлежащий изменению |
| линии горизонтальной и вертикальной прокрутки | для горизонтальной и вертикальной прокрутки документа |
| однострочный текстовый редактор | служит для ввода или отображения строки данных |

1. **Макеты экранов с описанием функциональности интерфейсных элементов**



**Рис. 8 – Элементы главного окна**

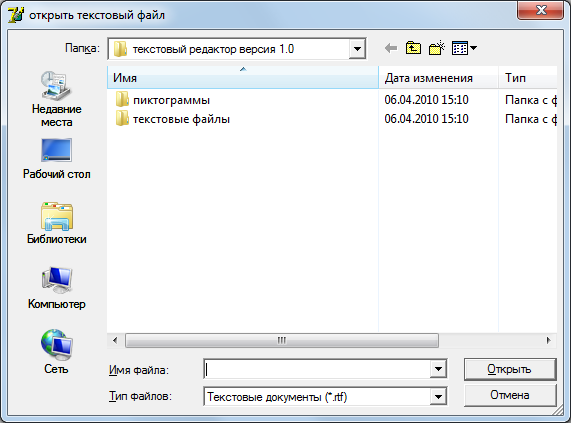


Рис. 9 – Элементы окна открытие файла

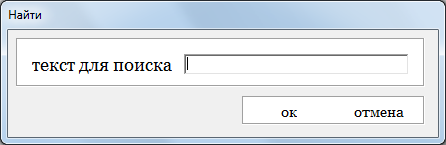


Рис. 10– Элементы окна поиска текста

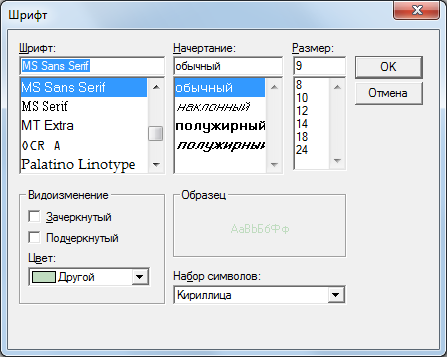
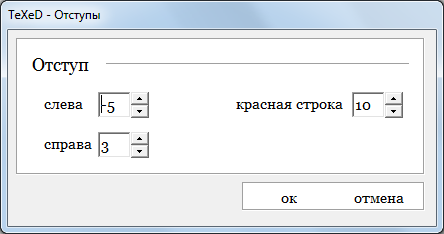


Рис. 11 – Элементы окна работы со шрифтами

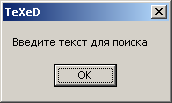


**Рис. 12 – Элементы окна Отступы**

1. **Запрограммировать различные реакции программного средства на действия пользователя**

Реакция программного средства на действия пользователя определяет алгоритм поведение пользователя для достижения им той или иной цели.

Например, после нажатия кнопки найти, с рисунка 10 при не заполненном поле текста, выводится окно-предупреждение, показанное на рисунке 13.



**Рис. 13 - Окно-предупреждение**