МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

кафедра «Вычислительная техника»

**Лабораторная работа №1**

по дисциплине: «Человек машина интерфейс».

Выполнил:

студент гр. ИВТАПбд-21

Вершинин Д. В.

Проверила:

Валюх В. В.

г. Ульяновск, 2018

**Программный продукт Работа с коллекциями с использованием WPF**

**1. Цели создания продукта:**

Приложение, позволяющее работать с коллекцией, в котором реализовано визуальное окно для работы с коллекцией. Данное приложение реализует такие функции как удаление, добавление, поиск и редактирование элемента коллекции, а также вывод коллекций на экран.

**2. Предметная область и сфера применения программного продукта:**

**2.1. Классический вариант в форме спецификаций**

Основным объектом предметной области являются язык программирования c# (реализация самой коллекции и работы с ней), подсистема WPF (реализация графического пользовательского интерфейса), класс базовой анимации – DoubleAnimation (анимация для модальных окон).

Предполагается, что с данным приложением будут работать студенты ФИСТ и преподаватели, и они должны иметь возможность решать с её помощью следующие задачи:

1. получение базовых знаний языка с#

2. изучение библиотеки DoubleAnimation

3.изучение встроенных структур данных и создание пользовательских коллекций

**2.2 В форме пользовательских историй**

Пожелание пользователя приложения:

1. Как преподавателю, использующему данное приложения, мне хотелось бы видеть правильное использование различных структур данных и коллекций для демонстрации студентам.

2. Как пользователю приложения мне необходимо иметь возможность видеть состояние коллекции в любой момент времени, а также добавлять и удалять элементы коллекций.

3. Как студенту, использующему данное приложение, мне хотелось бы научиться правильному использованию анимации в WPF.

**3. Целевая аудитория**

1. Преподаватели ФИСТ

Преподаватели имеют возможность использовать данное приложения в целях демонстрации студентам темы: «основы работы с WPF».

2. Студенты ФИСТ

Студенты имеют возможность использовать данное приложения в целях самостоятельного изучения темы: «основы WFP».

**4. Описательная модель пользователя (профиль).**

Пользователь ***студент***:

- Возраст 19-21 г.

- Род занятий: студент

- Уровень компьютерной грамотности: знание основ программирования, уверенная работа в среде программирования visual studio C#

- Задачи, решаемые пользователем: работа с коллекциями, анимация.

Пользователь ***преподаватель***:

- Возраст 25-50 г.

- Род занятий: преподаватель

- Уровень компьютерной грамотности: знание основ программирования, уверенная работа в среде программирования

- Продвинутое знание языка С#.

**5. Сценарии поведения пользователей на основании составленной модели.**

1. Производим запуск проекта в среде visual studio
2. Появляется главное окно (выбор действия для коллекции)
3. При нажатии на кнопку добавления появляется новое анимированное окно, в котором вводятся данные для нового элемента коллекции.
4. При завершении работы с окном и нажатии кнопки Ок для подтверждения окно плавно сворачивается.
5. При нажатии одну из кнопок с названиями коллекций в текстовом окне выводится список всех элементов соответствующей коллекции.

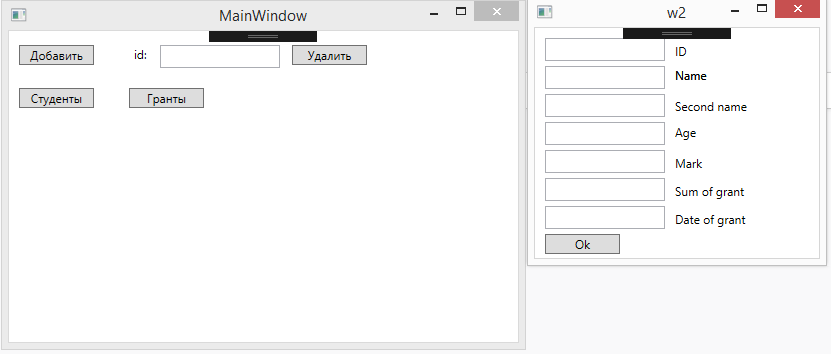


Рис. 1 – Внешний вид пользовательского интерфейса

**6. Функциональные блоки приложения и схема навигации между ними (структура диалога)**

В приложении можно выделить следующие функциональные блоки:

- Режим работы;

- Выбор действия над коллекцией;

- добавление или удаление элемента коллекции;

- Вывод результата на экран

Режим работы

1. выбор действия над коллекцией

3. добавление, удаление, изменение, поиск

**Пользователь**

Вывод результатов

Рис. 2 – Схема навигации

**7. Конечная концептуальная модель интерфейса, с применением UML нотации.**

Пользователь

Режим работы

Студент

Преподаватель

выбор действия над коллекцией

Выбор действия

добавление, удаление, изменение, поиск

**Программный продукт Визуализация работы двусвязных списков**

**1. Цели создания продукта.**

Приложение реализует работу с двусвязным списком в языке программирования Java. В данном приложении реализуются такие действия как добавление элемента в список, удаление элемента списка, поиск элемента, а также просмотр списка в прямом и обратном направлении.

**2. Предметная область и сфера применения программного продукта**

**2.1. Классический вариант в форме спецификаций**

Основным объектом предметной области является структура данных list, а также графический пользовательский интерфейс реализующий работу с данной коллекцией.

**2.2 В форме пользовательских историй**

Пожелание пользователя приложения:

1. Как преподавателю, использующему данное приложения, мне хотелось бы видеть наглядное представление списков языке Java для демонстрации обучающимся.

2. Как пользователю приложения мне необходимо иметь возможность добавления, удаления и поиска элемента в списке.

3. Как студенту, использующему данное приложение мне хотелось бы ознакомится со структурой списков в языке Java.

**3. Целевая аудитория**

1. Преподаватели ФИСТ

Преподаватели имеют возможность использовать данное приложение для демонстрации студентам правильного написания двусвязного списка в Java с применением указателей.

2. Студенты 1 курса.

Студенты имеют возможность использовать данное приложения в целях самостоятельного изучения темы: «Списки в Java».

**4. Описательная модель пользователя (профиль).**

Пользователь ***студент***:

- Возраст 19-21 год.

- Уровень компьютерной грамотности: знание основ программирования, уверенная работа в среде программирования Eclipse

Пользователь ***преподаватель***:

- Возраст 25-50 г.

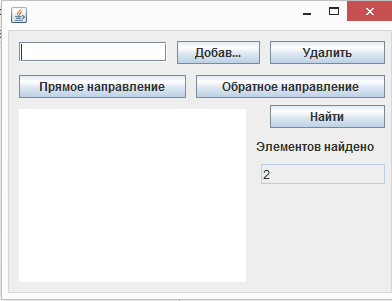
- Род занятий: преподаватель

- Уровень компьютерной грамотности: знание основ программирования, уверенная работа в среде программирования Eclipse

- Задачи, решаемые пользователем: демонстрация на лекциях и практических занятиях выполнения основных операций над структурой данных list демонстрация работы программного продукта, работа со списки на java.

**5. Сценарии поведения пользователей на основании составленной модели.**

1. Запустить проект в IDE Eclipse
2. Появится главное окно приложения



1. Выбор одного из доступных действий со списком
2. Просмотр результата при нажатии на кнопку «Прямое направление» или «Обратное направление»

**6. Функциональные блоки приложения и схема навигации между ними (структура диалога)**

В приложении можно выделить следующие функциональные блоки:

- Режим работы;

- Демонстрация интерфейса

- Выбор действия над коллекцией;

- добавление, удаление или поиск элемента коллекции;

- Вывод результата на экран

Режим работы

1. выбор действия над коллекцией

2. добавление, удаление, изменение, поиск

**Пользователь**

Вывод результатов

Рис. 3. Схема навигации

**7. Конечная концептуальная модель интерфейса, с применением UML нотации.**

Пользователь

Режим работы

Студент

Преподаватель

выбор действия над коллекцией

Выбор действия

добавление, удаление, изменение, поиск

**Программный продукт Игра "Тетрейка"**

**Цели создания продукта.**

Создание приложения-игры, совмещающей в себе игровые элементы тетриса и змейки, написанное на языке JavaScript. При открытии веб страницы с игрой перед игроком появляется игровое поле, по которому передвигается змейка. Сверху падают фигурки из классического тетриса. При поедании генерирующейся на игровом поле еды или поедания кусочка падающей фигуры счет увеличивается на 1 очко, а длина змейки возрастает. При постройке фигурок тетриса в ряд счет возрастает на 100 очков. Игра заканчивается если змейка врежется в границы экрана или уже построенную стену из фигурок тетриса или если башня из тетриса превысит высоту игрового поля.

**Предметная область и сфера применения программного продукта**

**2.1. Классический вариант в форме спецификаций**

Основным объектом предметной области является двумерный массив в языке JavaScript и пользовательский интерфейс представляющий игровое поле.

**2.2 В форме пользовательских историй**

Пожелание пользователя приложения:

1. Как преподавателю, использующему данное приложения, мне хотелось бы наглядно продемонстрировать студентам возможности языка JavaScript, а также увидеть визуализацию двумерных массивов для демонстрации студентам младших курсов.

2. Как пользователю приложения мне интересно поиграть в необычную игру

3. Как студент, использующий данное приложение я хочу лучше познаомится с языком JavaScript, для дальнейшего углубленного изучения в курсе «Web программирование»

**3. Целевая аудитория**

1. Преподаватели ФИСТ

Преподаватели имеют возможность использовать данное приложение для демонстрации студентам визуализации работы языка JavaScript.

2. Студенты 1-2 курсов.

Студенты имеют возможность использовать данное приложения в целях самостоятельного изучения темы: «JavaScript», а также для развлечения.

**4. Описательная модель пользователя (профиль).**

Пользователь:

- Возраст любой.

- ***Примечание для студентов***: Уровень компьютерной грамотности: знание основ программирования, уверенная работа в среде программирования MS Visual Studio

Пользователь ***преподаватель***:

- Возраст 25-50 г.

- Род занятий: преподаватель

- Уровень компьютерной грамотности: знание основ программирования

**5. Сценарии поведения пользователей на основании составленной модели.**

- При открытии веб страницы с игрой перед игроком появляется игровое поле, по которому передвигается змейка. Сверху падают фигурки из классического тетриса.

- При поедании генерирующейся на игровом поле еды или поедания кусочка падающей фигуры счет увеличивается на 1 очко, а длина змейки возрастает.

- При постройке фигурок тетриса в ряд счет возрастает на 100 очков.

- Игра заканчивается если змейка врежется в границы экрана или уже построенную стену из фигурок тетриса или если башня из тетриса превысит высоту игрового поля.

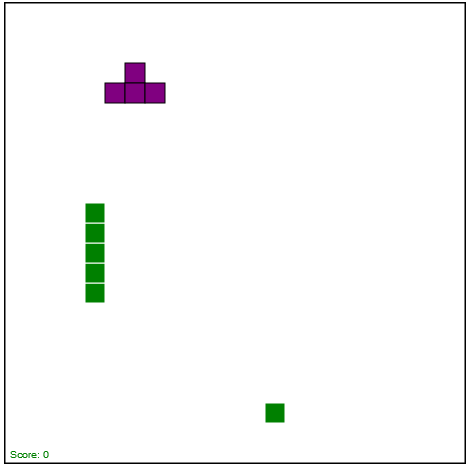


Рис. 4 – Внешний вид игрового поля

**6. Функциональные блоки приложения и схема навигации между ними (структура диалога)**

В приложении можно выделить следующие функциональные блоки:

- Режим работы;

- Демонстрация игрового поля;

- просмотр исходного кода игры;

- Управление змейкой;

- Управление тетрисом;

- Вывод итогового результата

Режим работы

1. Демонстрация игрового поля

2. Просмотр исходного кода

3. Управление змейкой

5. Управление тетрисом

**Пользователь**

Вывод результатов

Рис. 5. Схема навигации

**7. Конечная концептуальная модель интерфейса, с применением UML нотации.**

Управление змейкой

Пользователь

Режим работы

Студент

Преподаватель

Демонстрация графической игры

Просмотр и редактирование исходного кода игры

Игровой процесс

Управление тетрисом

Просмотр исходного кода через консоль браузера

**Программный продукт Визуализация сортировки одномерного массива**

**1. Цели создания продукта.**

Приложение реализует работу с одномерным массивом в языке программирования Java. В данном приложении наглядно демонстрируется пузырьковая сортировка строкового массива по возрастанию в случае чисел и по алфавиту в случае строк.

**2. Предметная область и сфера применения программного продукта**

**2.1. Классический вариант в форме спецификаций**

Основным объектом предметной области является структура данных array, а также графический пользовательский интерфейс реализующий обработку и сортировку массива.

**2.2 В форме пользовательских историй**

Пожелание пользователя приложения:

1. Как преподавателю, использующему данное приложения, мне хотелось бы видеть наглядное представление пузырьковой сортировки для демонстрации обучающимся.

2. Как студенту, использующему данное приложение мне хотелось бы ознакомится с пузырьковой сортировкой, а также с особенностями работы с массивами в языке Java.

**3. Целевая аудитория**

1. Преподаватели ФИСТ

Преподаватели имеют возможность использовать данное приложение для демонстрации студентам, начинающим обучаться программированию одной из самых простых сортировок.

2. Студенты 1 курса.

Студенты имеют возможность использовать данное приложения в целях самостоятельного изучения тем: «Одномерные массивы в Java» и «Сортировки».

**4. Описательная модель пользователя (профиль).**

Пользователь ***студент***:

- Возраст 19-21 год.

- Уровень компьютерной грамотности: знание основ программирования, уверенная работа в среде программирования Eclipse

Пользователь ***преподаватель***:

- Возраст 25-50 г.

- Род занятий: преподаватель

- Уровень компьютерной грамотности: знание основ программирования, уверенная работа в среде программирования Eclipse

- Задачи, решаемые пользователем: демонстрация на лекциях и практических занятиях выполнения основных операций над структурой данных array демонстрация работы программного продукта.

**5. Сценарии поведения пользователей на основании составленной модели.**

1. Запустить проект в IDE Eclipse
2. Появится главное окно приложения

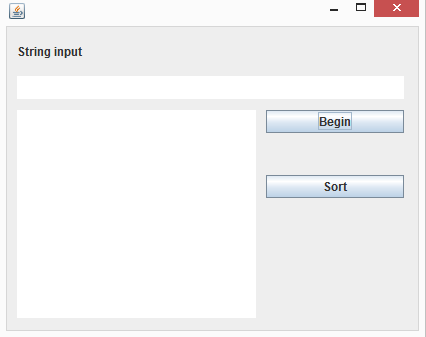


Рис. 6 – Главное окно приложения

1. Ввод в тестовое поле массива для сортировки
2. При нажатии на кнопку «Begin» введенный массив выводится на экран, стрелками указываются элементы, с которыми ведется работа в данный момент
3. При нажатии на кнопку «Sort» программа выполняет следующий шаг пузырьковой сортировки и обновляет вывод массива на экран

**6. Функциональные блоки приложения и схема навигации между ними (структура диалога)**

В приложении можно выделить следующие функциональные блоки:

- Режим работы;

- Демонстрация интерфейса

- Ввод массива;

- Начало сортировки, шаг сортировки;

- Вывод результата на экран

Режим работы

1. Ввод массива

2. Начало сортировки, шаг сортировки

**Пользователь**

Вывод результатов

Рис. 7 - Схема навигации

**7. Конечная концептуальная модель интерфейса, с применением UML нотации.**

Пользователь

Режим работы

Студент

Преподаватель

выбор действия над коллекцией

Выбор действия

Начало сортировки

Шаг сортировки

**Программный продукт Создание HTML страниц средствами C#**

**1. Цели создания продукта.**

Необходимо создать программу, создающую корректный HTML\_файл, содержащий весь текст из файла\_с\_текстом в котором помечены **жирным** и *наклонным* шрифтом все слова из файла\_словаря. HTML\_файл должен отображаться в любом Internet-браузере (IE, Firefox, Chrome и др.) корректно, показывая весь текст, с выделением указанных слов. Программу необходимо написать на языке C#.

При обработке файла\_с\_текстом нельзя целиком его загружать в память компьютера. Файл\_словаря наоборот нужно целиком загрузить в память, используя эффективные по времени структуры данных.

Для программы будет реализован консольный интерфейс, оповещающий о завершении работы программы.

Для считывания строк из входного файла и словаря будет использоваться потоковый ввод StreamReader, для хранения словаря реализуется отдельный класс на основе коллекции SortedSet.

**2. Предметная область и сфера применения программного продукта**

**2.1. Классический вариант в форме спецификаций**

Основным объектом предметной области является структура данных SortedSet и реализация работы с входными и выходными потоками.

**2.2 В форме пользовательских историй**

Пожелание пользователя приложения:

1. Как преподавателю, использующему данное приложения, мне хотелось бы видеть правильное использование структур данных языка C#.

2. Как студенту, использующему данное приложение мне хотелось бы научиться работать с потоковым чтением данных из файла и углубить свои знания языка С#.

**3. Целевая аудитория**

1. Преподаватели ФИСТ

Преподаватели имеют возможность использовать данное приложение для демонстрации студентам возможностей языка С# и демонстрации сложных структур данных.

2. Студенты.

Студенты имеют возможность использовать данное приложения в целях самостоятельного изучения потокового чтения из файла и создания HTML страниц средствами C#.

**4. Описательная модель пользователя (профиль).**

Пользователь ***студент***:

- Возраст 19-21 год.

- Уровень компьютерной грамотности: знание основ программирования, уверенная работа в среде программирования Visual Studio

Пользователь ***преподаватель***:

- Возраст 25-50 г.

- Род занятий: преподаватель

- Уровень компьютерной грамотности: знание основ программирования, уверенная работа в среде программирования Visual Studio

- Задачи, решаемые пользователем: демонстрация на лекциях и практических занятиях работы программного продукта.

**5. Сценарии поведения пользователей на основании составленной модели.**

1. Запустить проект в IDE Visual Studio
2. Появится консольное окно приложения
3. В папке с проектом создадутся входной файл и файл словаря, которые необходимо заполнить
4. При желании можно указать в консоли число, означающее сколько строк должно быть в каждом исходном файле

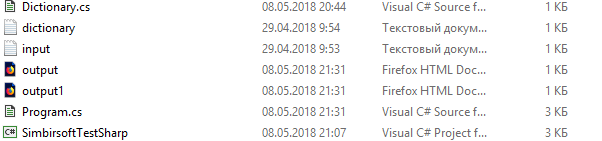


Рис. 8 – Файлы, созданные в результате работы приложения

**6. Функциональные блоки приложения и схема навигации между ними (структура диалога)**

В приложении можно выделить следующие функциональные блоки:

- Режим работы;

- Демонстрация консольного интерфейса

- Заполнение входного файла и словаря;

- Создание HTML файлов;

- Вывод результата на экран

Режим работы

1. Заполнение входных файлов

2. Создание HTML страниц

**Пользователь**

Вывод результатов

Рис. 9. Схема навигации

**7. Конечная концептуальная модель интерфейса, с применением UML нотации.**

Пользователь

Режим работы

Студент

Преподаватель

Демонстрация работы

Выбор действия

Просмотр результата

Заполнение файлов