МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ АКАДЕМИКА С. П. КОРОЛЁВА

(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)» (СГАУ)

Факультет Информатики

Кафедра Программных систем

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

к курсовому проекту

по теме «Автоматизированная система «Клавиатурный тренажер»

с функциями администратора»

Студенты А.Е. Акимова

Д.И. Ершова

О.В. Пищулина

А.И. Хвостов

Руководитель Л.С. Зеленко

Самара 2014 г.

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ УНИВЕРСИТЕТ**

**ИМЕНИ АКАДЕМИКА С.П.КОРОЛЕВА**

**(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)» (СГАУ)**

Факультет Информатики

Кафедра Программных систем

**ЗАДАНИЕ**

на курсовой проект по дисциплине

«Программная инженерия»

студентам группы № 6413 Б 200  
Акимовой А.Е.  
Ершовой Д.И.  
Пищулиной О.В.  
Хвостову А.И.

1. **Тема проекта:** «Автоматизированная система «Клавиатурный тренажер» с функциями администратора»
2. **Исходные данные к проекту:** см. приложение к заданию
3. **Перечень вопросов, подлежащих разработке в курсовом проекте:**
   1. Произвести анализ предметной области: изучить методику «слепого» набора текста, альтернативные способы скоропечатания и варианты расположения клавиш на клавиатуре
   2. Выполнить обзор существующих систем-аналогов
   3. Разработать информационно-логический проект по методологии UML
   4. Разработать и реализовать программное и информационное обеспечение, провести его тестирование и отладку
   5. Оформить документацию курсового проекта
   6. Подготовить презентацию по разработанной системе
4. **Перечень графических разработок**
   1. Диаграмма UML
   2. Схемы основных алгоритмов
5. **Календарный план выполнения работ**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | Содержание работы по этапам | Объем этапа в % к общему объему проекта | Срок  окончания | Фактическое выполнение |
| 1 | Оформление технического задания и его утверждение | 5 | 11.09.2014 |  |
| 2 | Анализ и описание предметной области | 10 | 25.09.2014 |  |
| 3 | Проектирование системы | 30 | 23.10.2014 |  |
| 3.1 | Разработка структурной схемы системы | 5 | 23.10.2014 |  |
| 3.2 | Разработка функциональной спецификации  системы | 10 | 23.10.2014 |  |
| 3.3 | Разработка информационно-логического проекта системы и его предъявление руководителю | 15 | 23.10.2014 |  |
| 4 | Реализация проекта, разработка контрольных примеров. Предъявление реализации руководителю | 45 | 04.12.2014 |  |
| 5 | Корректировка проекта и оформление документации проекта. Защита проекта с представлением презентации. | 10 | 18.12.2014 |  |

Задание приняли к исполнению \_\_\_\_\_\_\_\_\_\_\_ / Акимова А.Е. / 01.09.2014

\_\_\_\_\_\_\_\_\_\_\_\_ / Ершова Д.И. / 01.09.2014

\_\_\_\_\_\_\_\_\_\_\_\_ / Пищулина О.В. /01.09.2014

\_\_\_\_\_\_\_\_\_\_\_\_ / Хвостов А.И. / 01.09.2014

**ПРИЛОЖЕНИЕ  
к заданию на курсовой проект**студентов группы № 6413 Б 200  
Акимовой А.Е.  
Ершовой Д.И.  
Пищулиной О.В.  
Хвостова А.И.

Тема проекта: **«Автоматизированная система «Клавиатурный тренажер» с функциями администратора»**

**Исходные данные к проекту:**

1. **Характеристика объекта автоматизации:**
2. объект автоматизации: клавиатурный тренажер;
3. виды автоматизируемой деятельности:
   * процесс составления/генерирования упражнения;
   * процесс «гибкого» выбора сложности;
   * процесс администрирования системы;
   * процесс сбора статистических данных;
   * процесс визуализации работы с тренажером;
4. минимальное количество символов в упражнении – 30 символов;
5. максимальное количество символов в упражнении – 1000 символов;
6. количество возможных режимов печати – 4;
7. максимальное количество возможных ошибок в режиме итогового тестирования – не более 10% от длины упражнения;
8. минимальная скорость печати в режиме «На время» ­ 150 символов/мин;
9. количество способов создания упражнения – 2;
10. язык раскладки клавиатуры – русский;
11. количество уровней сложности – не менее 3;
12. количество видов визуализации – не менее 2.
13. **Требования к информационному обеспечению:**
14. информационное обеспечение разрабатывается на основе следующих документов:
    * Слепой метод печати [Электронный ресурс] – ru.wikipedia.org/wiki/Слепой\_метод\_печати(дата обращения 06.09.2014 г.);
    * Шахиджанян, В.В. Соло на пишущей машинке [Текст]. – М.: Книга, 1990. – 144 с.;
15. статистика, упражнения и информация о пользователях хранятся в файлах формата \*.xml, структура файлов определяется в процессе проектирования.
16. **Требования к техническому обеспечению:**
17. тип ЭВМ – IBM PC совместимый;
18. монитор с разрешающей способностью – не ниже 1024 х 768;
19. манипулятор – мышь;
20. тип раскладки клавиатуры – QWERTY;
21. технические характеристики определяются в процессе выполнения курсового проекта.
22. **Требования к программному обеспечению:**
23. тип операционной системы – Windows XP и выше;
24. язык программирования – С#;
25. среда программирования –Visual Studio 2013;
26. среда проектирования – StarUML 5.0.
27. **Общие требования к проектируемой системе:**

***5.1 Функции, реализуемые системой:***

1. общесистемные:
2. авторизация пользователя в системе;
3. аутентификация пользователя в системе;
4. настройка интерфейса пользователя в зависимости от роли;
5. визуализация процессов работы с тренажером;
6. выдача справочной информации о системе;
7. ведение статистики;
8. функции администратора:
9. работа с учетными данными пользователей:

* добавление пользователя;
* удаление пользователя;
* просмотр учетных записей;
* редактирование данных пользователя;

1. работа со статистикой с возможностью фильтрации данных по упражнениям или обучаемым;
2. создание упражнений;
3. функции обучаемого:
4. выбор сложности;
5. выбор упражнения;
6. просмотр статистики;
7. выполнение упражнения.

***5.2 Технические требования к системе:***

1. режим работы – диалоговый;
2. время автоматической генерации упражнения не более 1 минуты;
3. система должна удовлетворять санитарным правилам и нормам  
   СанПин 2.2.2./2.4.2198-07;
4. условия работы средств вычислительной техники (содержание вредных веществ, пыли и подвижность воздуха) должны соответствовать ГОСТ 12.1.005, 12.01.007.
5. температура окружающего воздуха – 15-25°С;
6. влажность 45-75%.

Руководитель   
курсового проекта \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Зеленко Л.С. / \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Задание приняли   
к исполнению \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Акимова А.Е. / \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Ершова Д.И. / \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Пищулина О.В. /\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Хвостов А.И. /\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**РЕФЕРАТ**

Пояснительная записка 100 с, 80 рисунков, 8 таблиц, 16 источников,  
2 приложения.

Графическая документация: 15 слайдов презентации Power Point.

**АВТОМАТИЗИРОВАННАЯ СИСТЕМА, КЛАВИАТУРНЫЙ ТРЕНАЖЕР, МЕТОД СЛЕПОЙ ПЕЧАТИ, ОБУЧЕНИЕ, ВИЗУАЛИЗАЦИЯ, ВИРТУАЛЬНАЯ КЛАВИАТУРА, РАСКЛАДКА КЛАВИАТУРЫ**

В рамках курсового проектирования разработана автоматизированная система «Клавиатурный тренажер» с функциями администратора.

В системе предусмотрены две роли: администратор и обучаемый. К функциям администратора относятся функции по созданию упражнений, работе с учетными записями пользователей и работе со статистикой. В функции обучаемого входит прохождение упражнений и просмотр собственной статистики. В системе реализован контроль вводимых пользователем данных.

Информация об учетных записях пользователей, статистике выполнения упражнений и сами упражнения хранятся в файлах формата \*.xml.

Программа написана на языке С# в среде Visual Studio 2013 и функционирует в операционной системе Windows’XP и выше. Проектирование велось с использованием среды проектирования Star UML 5.0.

СОДЕРЖАНИЕ

[Введение 10](#_Toc409739964)

[1 Описание и анализ предметной области 12](#_Toc409739965)

[1.1 Описание предметной области 12](#_Toc409739966)

[1.1.1 История появления клавиатурного тренажера 12](#_Toc409739967)

[1.1.2 Методы набора текста 12](#_Toc409739968)

[1.1.3 Варианты расположения клавиш на клавиатуре 13](#_Toc409739969)

[1.2 Обзор существующих систем-аналогов 16](#_Toc409739970)

[1.2.1 «Соло на клавиатуре» 16](#_Toc409739971)

[1.2.2 Stamina 18](#_Toc409739972)

[1.2.3 «Все 10» 19](#_Toc409739973)

[1.2.4 VerseQ 20](#_Toc409739974)

[1.2.5 RapidTyping 22](#_Toc409739975)

[1.3 Диаграмма объектов предметной области 24](#_Toc409739976)

[1.4 Постановка задачи 25](#_Toc409739977)

[2 Проектирование системы 29](#_Toc409739978)

[2.1 Структурная схема системы 29](#_Toc409739979)

[2.2 Спецификация требований к проектируемой системе 31](#_Toc409739980)

[2.3 Разработка прототипа интерфейса системы 52](#_Toc409739981)

[2.4 Информационно-логический проект системы 61](#_Toc409739982)

[2.4.1 Язык UML 61](#_Toc409739983)

[2.4.2 Диаграмма вариантов использования 62](#_Toc409739984)

[2.4.3 Диаграмма последовательности 74](#_Toc409739985)

[2.4.4 Диаграмма деятельности 76](#_Toc409739986)

[2.4.5 Диаграмма состояний 78](#_Toc409739987)

[2.4.6 Диаграмма классов 80](#_Toc409739988)

[2.5 Выбор и обоснование комплекса программных средств 83](#_Toc409739989)

[2.5.1 Выбор языка программирования и среды разработки 83](#_Toc409739990)

[2.5.2 Выбор операционной системы 84](#_Toc409739991)

[3 Реализация системы 85](#_Toc409739992)

[3.1 Разработка и описание интерфейса пользователя 85](#_Toc409739993)

[3.1.1 Разработка и описание пользовательского меню 85](#_Toc409739994)

[3.1.2 Описание контрольного примера в режиме администратора 92](#_Toc409739995)

[3.1.3 Описание контрольного примера в режиме обучаемого 94](#_Toc409739996)

[3.2 Диаграммы реализации 97](#_Toc409739997)

[3.2.1 Диаграмма компонентов 97](#_Toc409739998)

[3.2.2 Диаграмма развертывания 98](#_Toc409739999)

[3.3 Выбор и обоснование комплекса технических средств 99](#_Toc409740000)

[3.3.1 Расчет объема занимаемой памяти 99](#_Toc409740001)

[3.3.2 Минимальные требования, предъявляемые к системе 100](#_Toc409740002)

[Заключение 101](#_Toc409740003)

[Список использованной литературы 102](#_Toc409740004)

[Приложение А Руководство пользователя 104](#_Toc409740005)

[Приложение Б Листинг программы 105](#_Toc409740006)

Введение

В настоящее время компьютеры широко используются в повседневной жизни. Для эффективного обмена информацией её необходимо быстро и безошибочно заносить на электронные носители. Слепой десятипальцевый метод печати дает ряд преимуществ: минимизируются ошибки, повышается качество печати, заняты все пальцы и нагрузка на них распределена по зонам, набор текста становится механическим (задействуется мышечная память пальцев и кистей рук), сохраняется острота зрения, повышается производительность труда. Сейчас слепой метод является необходимым навыком любого квалифицированного сотрудника. Развить этот навык помогают клавиатурные тренажеры. Большое их количество представлено в бесплатном и платном пользовании. Платные версии не устраивают пользователей из-за условий их распространения. Рассмотренные бесплатные версии обладают недостаточными функциональными требованиями либо неудобным интерфейсом.

В рамках курсового проекта необходимо разработать клавиатурный тренажер с возможностью обучения слепому набору текста, выполнения упражнений, тестирования пользователей и просмотра статистики. В отличие от классических версий в системе должна быть реализована возможность создания упражнений и настройки уровней сложности.

При разработке системы будет использована технология RAD (Rapid Application Development – быстрая разработка приложений), позволяющая осуществить проектирования ПО небольшой командой разработчиков за срок порядка трех-четырех месяцев путем использования инкрементного прототипирования с применением инструментальных средств визуального моделирования и разработки., и методология UML (Unified Modeling Language – унифицированный язык моделирования), которая использует графические обозначения для создания абстрактной модели системы и позволяет определять, визуализировать, проектировать и документировать программные системы [1, 2].

1. Описание и анализ предметной области

1.1 Описание предметной области

Клавиатурный тренажер – это обучающая компьютерная программа или онлайн-сервис, позволяющие научиться быстро печатать на клавиатуре компьютера [3, 4].

1.1.1 История появления клавиатурного тренажера

История создания клавиатурного тренажера связана с появлением слепого метода печати, который был разработан в 1878 году стенографистом Франком Эдвардом Макгуррином в результате соперничества по скорости печати на новой пишущей машинке.

Первые российские клавиатурные тренажеры появились в 90-х годах прошлого века. Они были написаны для операционной системы DOS. Это такие тренажеры, как «Тренажер клавиатуры» (TRK), BabyType и «Алёнка». В конце двадцатого века появился российский тренажер «Соло на клавиатуре», который является воплощением методики скоропечатания, описанной в книге В.В. Шахиджаняна «Соло на пишущей машинке» 1991 года.

1.1.2 Методы набора текста

Слепой метод набора – методика набора текста «вслепую», используя все (или большинство) пальцы рук [4].

В обучении слепому методу предметом изучения является не расположение букв и клавиш на клавиатуре, а движения пальцев, которые необходимо соединить в подсознании с определённой буквой или знаком. Этих движений три вида:

* удар по клавишам из исходных позиций;
* движения от клавишей исходных позиций к другим клавишам той же зоны и удар по ним;
* движения от любой клавиши к любой клавише той же зоны и удар по этим клавишам.

Зрячий метод печати – методика ввода текста, при котором печатающий смотрит на клавиши клавиатуры [5]. Количество пальцев, используемых в печати, их [зоны](http://klavogonki.ru/wiki/index.php?title=%D0%97%D0%BE%D0%BD%D1%8B&action=edit&redlink=1), а также время, потраченное на подсматривание на клавиатуру, сугубо индивидуальны.

Основные виды зрячего набора:

* однопальцевый;
* двухпальцевый;
* подключение дополнительных пальцев.

Существуют две основные позиции начальной постановки рук при наборе текста:

* ФЫВА ОЛДЖ – самая распространённая [исходная позиция](http://klavogonki.ru/wiki/%D0%98%D1%81%D1%85%D0%BE%D0%B4%D0%BD%D0%B0%D1%8F_%D0%BF%D0%BE%D0%B7%D0%B8%D1%86%D0%B8%D1%8F) при [слепой печати](http://klavogonki.ru/wiki/%D0%A1%D0%BB%D0%B5%D0%BF%D0%BE%D0%B9_%D0%BC%D0%B5%D1%82%D0%BE%D0%B4_%D0%BF%D0%B5%D1%87%D0%B0%D1%82%D0%B8). К её особенностям следует отнести большую нагрузку на самые развитые и быстрые пальцы – указательные. Без учёта знаков препинания и цифр на их долю приходится до 60% нажатий [6].
* ЫВАМ ТОЛД – исходная [позиция](http://klavogonki.ru/wiki/%D0%98%D1%81%D1%85%D0%BE%D0%B4%D0%BD%D0%B0%D1%8F_%D0%BF%D0%BE%D0%B7%D0%B8%D1%86%D0%B8%D1%8F) при [слепой печати](http://klavogonki.ru/wiki/%D0%A1%D0%BB%D0%B5%D0%BF%D0%BE%D0%B9_%D0%BC%D0%B5%D1%82%D0%BE%D0%B4_%D0%BF%D0%B5%D1%87%D0%B0%D1%82%D0%B8). Впервые появилась в клавиатурном тренажёре [Stamina](http://klavogonki.ru/wiki/Stamina" \o "Stamina). Главное отличие от [ФЫВА ОЛДЖ](http://klavogonki.ru/wiki/%D0%A4%D1%8B%D0%B2%D0%B0-%D0%BE%D0%BB%D0%B4%D0%B6) – сдвиг кистей ближе к центру клавиатуры, это зачастую сопровождается другим взаимным расположением предплечий (локти разводятся под большим углом). За это приходится платить существенным увеличением нагрузки на мизинцы [7].

1.1.3 Варианты расположения клавиш на клавиатуре

В настоящее время основной русскоязычной раскладкой является раскладка «ЙЦУКЕН», название которой происходит от шести левых символов верхнего ряда.

Существует несколько вариантов этой раскладки. Они в основном отличаются расположением знаков препинания. Ниже на рисунке 1.1 и рисунке 1.2 представлены два вида раскладки «ЙЦУКЕН».

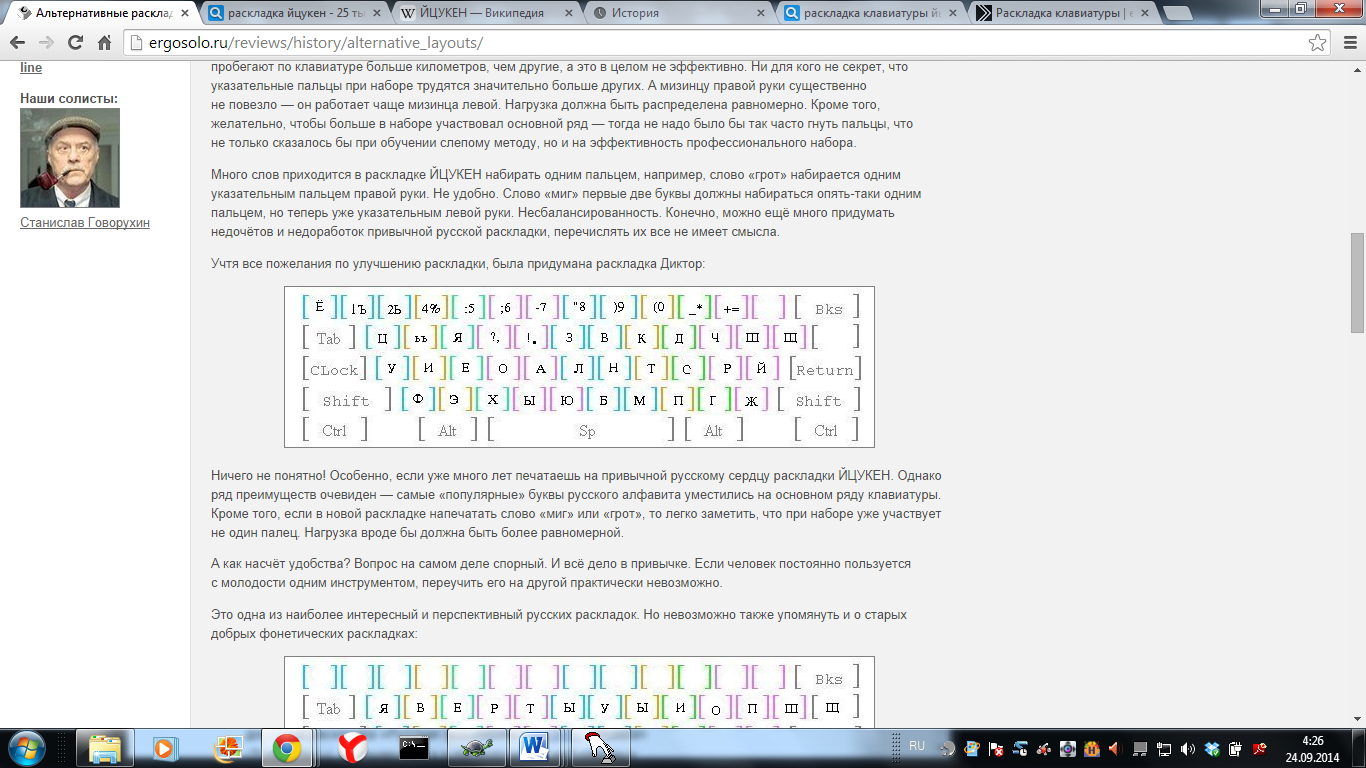
  
Рисунок 1.1 – Раскладка «Русская»

  
Рисунок 1.2 – Раскладка «Русская (машинопись)»

«Русская» раскладка используется по умолчанию в большинстве операционных систем для русского языка, «Русская (машинопись)» – в качестве альтернативного варианта. В последней для набора запятой, точки, точки с запятой и других знаков достаточно просто нажать одну кнопку в четвертом ряду, в то время как при обычной русской раскладке необходимо нажимать одновременно две клавиши, что не совсем удобно.

Считается, что данная раскладка далека от оптимальной. Во-первых, нагрузка по пальцам распределяется неравномерно, из-за чего одни пальцы «пробегают» по клавиатуре большие расстояния, чем другие. Например, указательные пальцы трудятся больше других, а мизинец правой руки работает чаще мизинца левой. Во-вторых, приходится часто гнуть пальцы, так как основной «домашний» ряд клавиатуры мало задействован. Также часто два и более символа нажимаются подряд в одной зоне, одним пальцем.

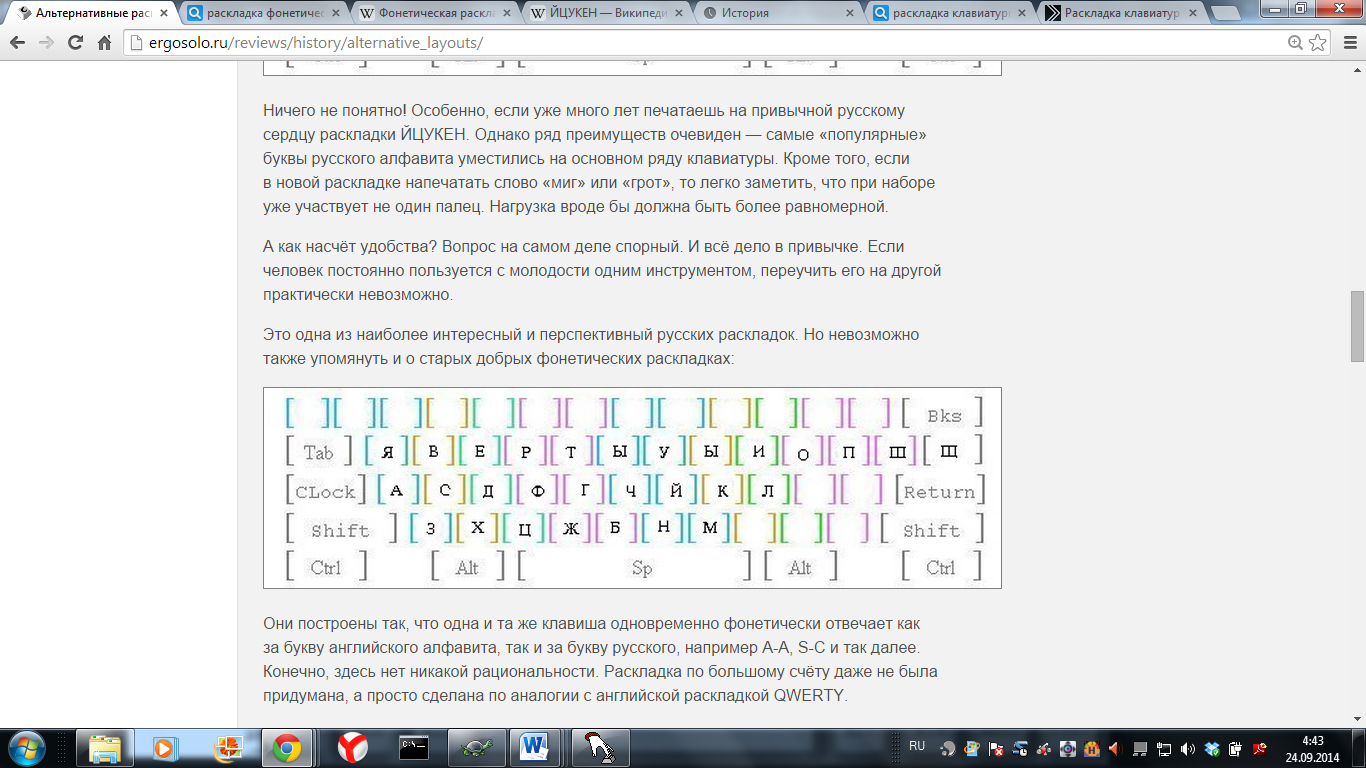
Поэтому были предложены новые раскладки, которые должны были облегчить труд наборщиков текста. Одна из таких раскладок называется «Диктор», изображенная на рисунке 1.3.

  
Рисунок 1.3 – Раскладка «Диктор»

Раскладка «Диктор» обладает рядом преимуществ. Во-первых, самые «популярные» буквы русского алфавита расположены в основном ряду. Также при наборе часто употребляемых слов используется чередование пальцев, что хорошо влияет на скорость печати. Кроме того, в новой раскладке нагрузка на пальцы должна быть распределена более равномерно.

Следует также выделить фонетическую раскладку. Она построена по принципу того, что одна и та же клавиша фонетически отвечает, как за букву английского алфавита, так и за букву русского. Т.е. данная раскладка сделана по аналогии с английской раскладкой «QWERTY».

Русская фонетическая раскладка считается лучшей для иностранцев, изучающих русский язык. На рисунке 1.4 представлена фонетическая раскладка.

  
Рисунок 1.4 – Фонетическая раскладка

В разрабатываемой системе будет использована стандартная раскладка «ЙЦУКЕН».

1.2 Обзор существующих систем-аналогов

1.2.1 «Соло на клавиатуре»

Обзор систем-аналогов начнем с самого известного в странах Содружества Независимых Государств клавиатурного тренажера – "Соло на клавиатуре" [3]. Программа является полноценным обучающим курсом для изучения метода слепой печати. Согласно отзывам пользователей программы – она обладает высокой эффективностью, но требует от пользователей усидчивости и большое количество свободного времени.

Тренажер «Соло на клавиатуре» имеет следующие особенности:

* более 100 упражнений, которые в свою очередь могут разделяться на подзадания (рисунок 1.5);

  
Рисунок 1.5 – Упражнения тренажера «Соло на клавиатуре»

* тренировочный экзамен, который по желанию может пройти пользователь, чтобы проверить свое владение клавиатурой;
* большое число психологических тестов и статей, направленных на развитие личностных качеств пользователей (рисунок 1.6);

  
Рисунок 1.6 – Психологические тесты тренажера «Соло на клавиатуре»

* подробная статистика (рисунок 1.7);

  
Рисунок 1.7 – Статистика тренажера «Соло на клавиатуре»

* блок виртуальной клавиатуры с выделенными зонами;
* комплекс статей, посвященных слепому методу печати, работе с компьютером, технике положения тела;
* различные игры;
* выбор языка.

Был выделен ряд недостатков тренажера:

* прохождение тестового задания требует слишком много времени;
* неудобный интерфейс выбора начала упражнения, так как кнопка начала находится в неудобном месте;
* «Соло на клавиатуре» является платным тренажером.

1.2.2 Stamina

Клавиатурный тренажер «Stamina» – платный тренажер, в котором присутствует возможность редактирования списка пользователей. В программе присутствуют курсы скоропечатания для русского и английского языков. Программа обладает простым дизайном и понятными настройками [8].

В тренажере «Stamina» присутствуют упражнения на суффиксы, приставки, окончания, определенный набор символов, что отличает программу от других систем-аналогов (рисунок 1.8). Также есть возможность редактирования упражнений и альтернативная постановка пальцев на клавиатуре.

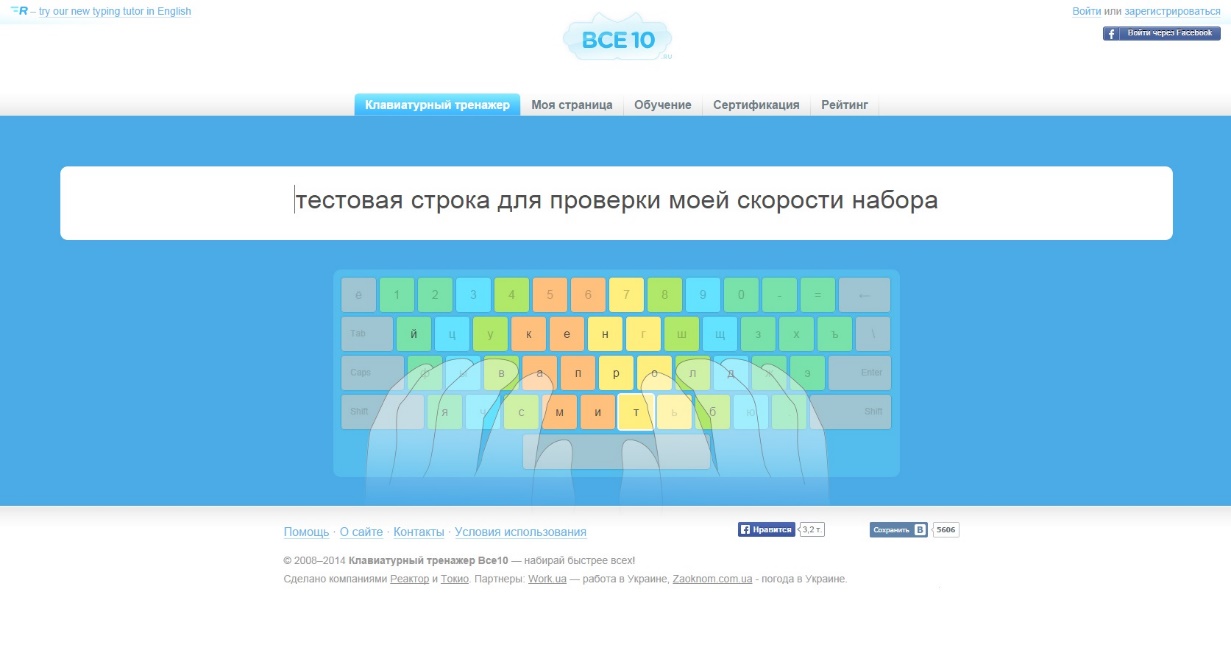
  
Рисунок 1.8 – Главная страница клавиатурного тренажера «Stamina»

По завершении упражнения пользователю отображается статистика выполнения этого упражнения (рисунок 1.9).

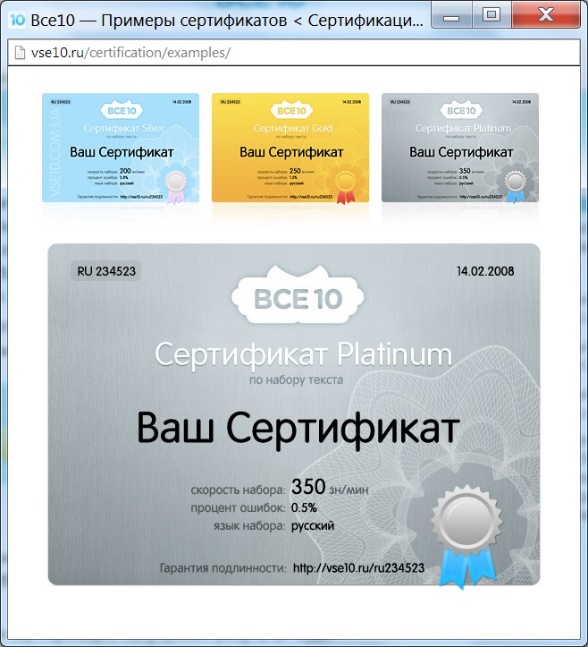
  
Рисунок 1.9 – Статистика выполнения упражнения на тренажере «Stamina»

1.2.3 «Все 10»

Клавиатурный тренажер «Все 10» – бесплатный онлайн-тренажер, не требующий установки программы на компьютер, что делает его хорошей альтернативой прочим программам [9]. Данный тренажер требует регистрации. Перед прохождением курса обучения на тренажере «Все 10» необходимо пройти тестовое упражнение (рисунок 1.10).

  
Рисунок 1.10 – Тест на клавиатурном тренажере «Все 10»

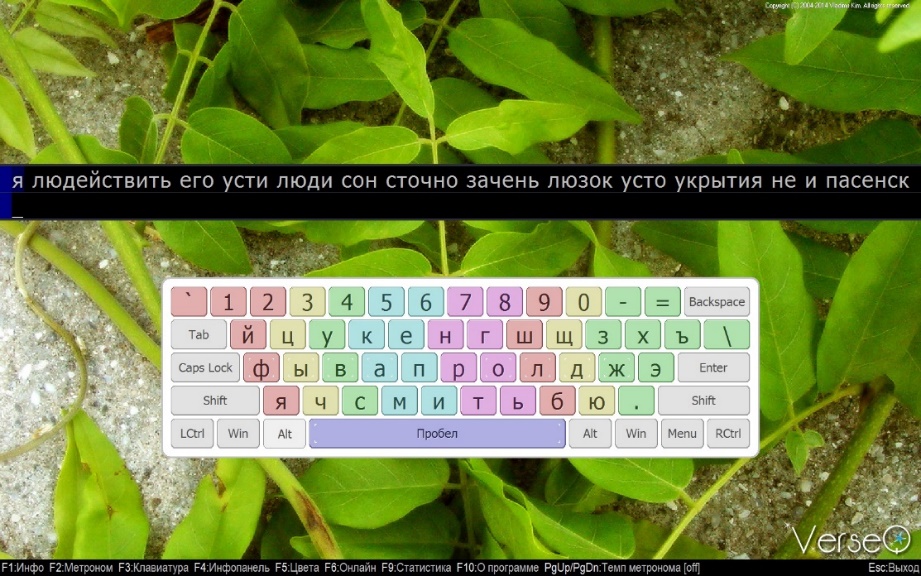
После обучения вы можете получить сертификат, удостоверяющий о результатах прохождения итогового тестирования (рисунок 1.11).

  
Рисунок 1.11 – Примеры сертификатов

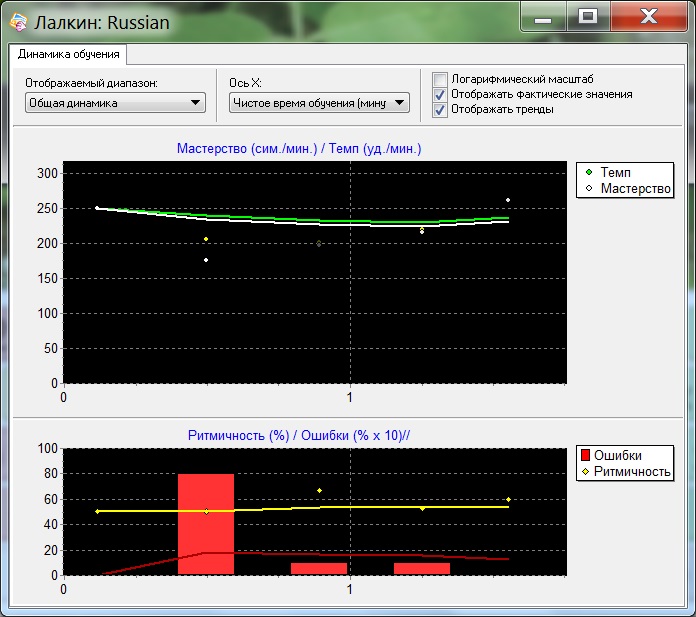
Программа содержит ряд упражнений, статистику их выполнения и сетевой рейтинг пользователей. Также тренажер «Все 10» содержит методические указания по правильному положению тела при работе за компьютером.

1.2.4 VerseQ

Клавиатурный тренажер VerseQ – является платным клавиатурным тренажером, в котором присутствует возможность регистрации пользователей [10]. Программа содержит некоторые методические указания. На рисунке 1.12 отображен клавиатурный тренажер VerseQ в ходе выполнения упражнения.

  
Рисунок 1.12 – Клавиатурный тренажер VerseQ

Тренажер динамически подстраивается под ошибки пользователя и автоматически генерирует задания на основе наиболее часто встречающихся ошибок (рисунок 1.13). Программа содержит отключаемую виртуальную клавиатуру.

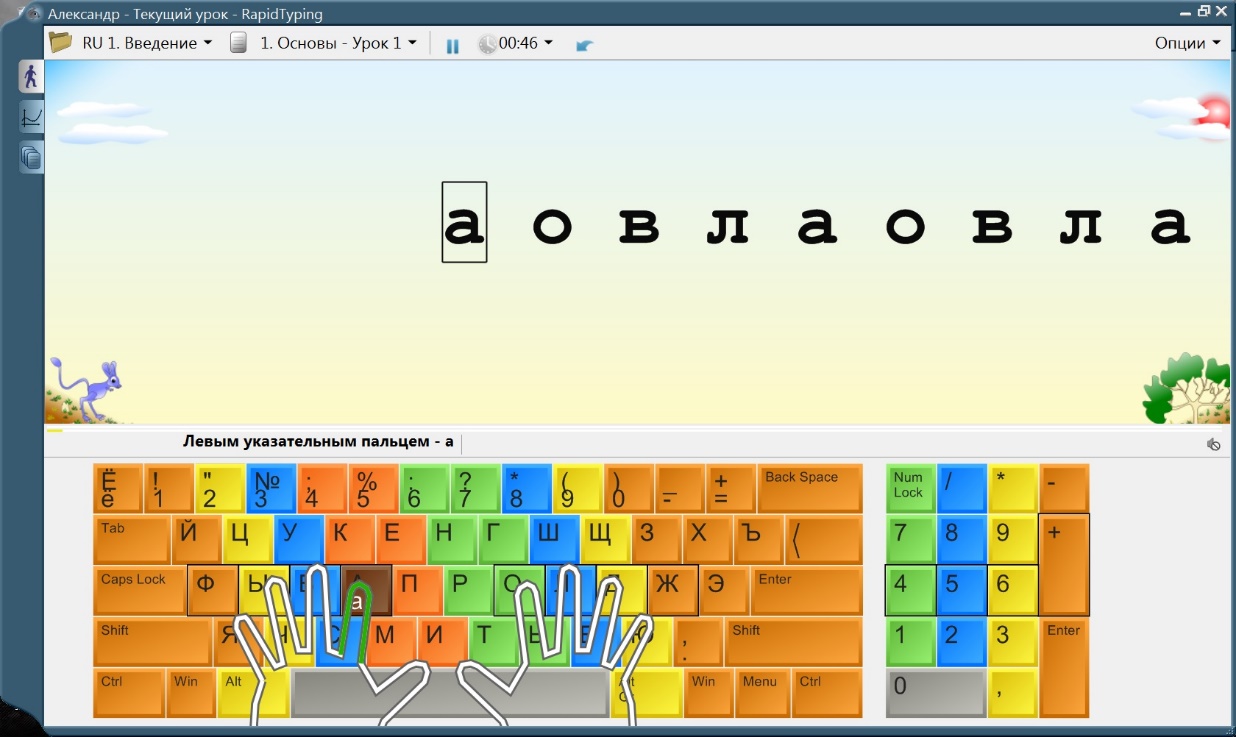
  
Рисунок 1.13 – Динамика обучения пользователя на тренажере VerseQ

В программе отсутствует типичный для других тренажеров режим печати только одного символа – обучение печатанию начинается сразу со всей клавиатуры.

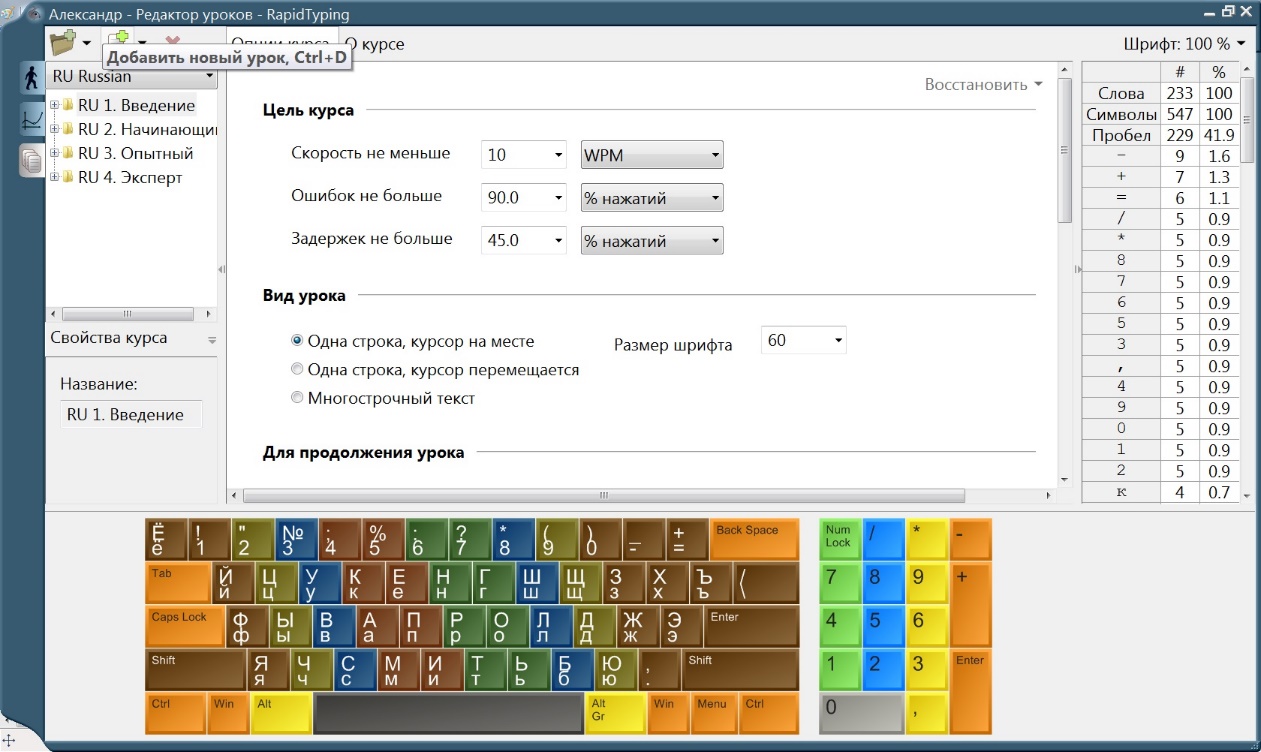
По сравнению с другими тренажерами не содержит достаточного функционала, чтобы являться полноценным обучающим продуктом, так как представляет из себя только строку с упражнением и виртуальную клавиатуру. При стоимости, большей стоимости "Соло на клавиатуре", не идет с ней ни в какое сравнение по обучающей составляющей.

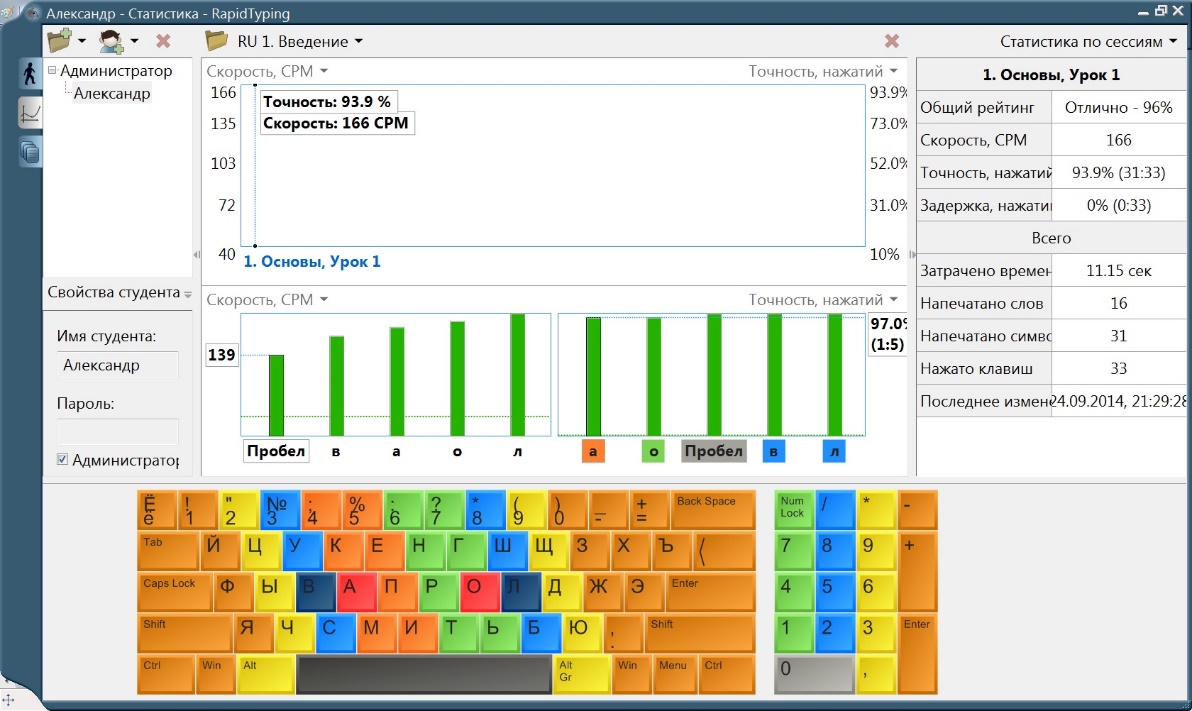
1.2.5 RapidTyping

Клавиатурный тренажер RapidTyping – бесплатный тренажер, в стандартной редакции содержащий 4 уровня сложности для русского языка и 5 для английского [11]. Кроме того, существует возможность загрузки других языков из большой базы. Каждый уровень сложности содержит ряд уроков, которые в свою очередь разделяются на подзадания (рисунок 1.14).

  
Рисунок 1.14 – Клавиатурный тренажер RapidTyping

В тренажере присутствует режим администрирования и редактирования упражнений (рисунок 1.15), имеется большое количество различных настроек и подробная статистика (рисунок 1.16).

  
Рисунок 1.15 – Режим редактирования упражнений

  
Рисунок 1.16 – Статистика выполнения упражнений пользователя

По мере выполнения упражнений отображаются подсказки по расположению пальцев на клавиатуре.

1.3 Диаграмма объектов предметной области

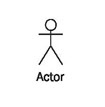
При построении данной диаграммы нужно помнить о том, что в данном случае объект – это «конкретная материализация абстракции», а не экземпляр класса, т.е. пока это понятие не программистское. Диаграмма объектов представляет статическую составляющую взаимодействующих между собой объектов, она должна включить в себя только те объекты предметной области, которые потом преобразуются в диаграмму классов. Связи между объектами показывают отношения между ними, при необходимости в диаграмме можно привести и атрибуты (свойства) объектов [13]. На рисунке 1.17 приведена диаграмма объектов предметной области.

Клавиатурный тренажер

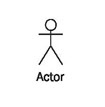
Упражнение

Статистика с

Виртуальная клавиатура



Администратор



Обучаемый

База данных

Хранится в

Просматривает

Выполняет

Отображается на

Составляет

Накапливает

Рисунок 1.17 – Диаграмма объектов предметной области

Основным объектом программы «Клавиатурный тренажер» является упражнение. Упражнения создаются администратором вручную или автоматически генерируются системой, после чего они сохраняются в файле. Обучаемый выбирает необходимый уровень сложности и проходит упражнение, приобретая навыки. Также во время выполнения упражнение отображается на виртуальной клавиатуре (подсвечивается текущий символ упражнения).

Все результаты (количество выполненных упражнений, время выполнения упражнения, отношение количества допущенных ошибок к максимально допустимому количеству ошибок) прохождения упражнений сохраняются в разделе статистика. Обучаемый может просмотреть свою статистику. Администратор же может просмотреть статистику и по определенному упражнению, и по определенному обучаемому.

В разрабатываемой системе «Клавиатурный тренажер» предусмотрены две роли пользователя: администратор и обучаемый.

1.4 Постановка задачи

В рамках курсового проектирования поставлена задача – разработать автоматизированную систему «Клавиатурный тренажер с функциями администрирования», которая позволит вести административный контроль учетных записей пользователей, создавать и выполнять упражнения, собирать статистику и визуализировать процесс выполнения упражнений. Обучение методу слепого набора текста будет проводится для клавиатуры с русской раскладкой «ЙЦУКЕН».

В системе должны быть предусмотрены две роли: администратор и обучаемый.

Перед началом работы пользователь должен пройти процедуру авторизации, указав логин и пароль, или зарегистрироваться в системе. После этого система должна выполнить проверку введенных данных и, в случае успеха, настроить интерфейс работы с клавиатурным тренажером в соответствии с ролью пользователя.

В режиме администрирования пользователю должны быть доступны создание и редактирование упражнений, просмотр расширенной статистики выполнения упражнений пользователями и работа с учетными записями пользователей, которая включает в себя создание и удаление учетных записей, изменение личной информации пользователей и изменение уровня сложности.

В системе должно быть реализовано 3 уровня сложности прохождения упражнений (новичок, любитель, профессионал), которые будут отличаться друг от друга количеством символов в упражнении (минимальное количество символов – 30, максимальное – 1000), минимальной скоростью набора символов (не менее 150 символов/мин) и максимальным количеством ошибок при прохождении упражнений (не более 10% от длины упражнения).

Информация об учетных записях и настройках пользователей, статистике выполнения и тексте упражнений должна храниться в файлах формата \*.xml (eXtensible Markup Language – расширяемый язык разметки).

В начале работы в режиме обучаемого каждый пользователь должен будет пройти входное тестирование, которое позволит определить уровень владения навыком слепого метода печатания, по результатам которого (учитывается количество сделанных ошибок и средняя скорость набора символов) определяются начальные настройки сложности для конкретного пользователя.

После прохождения входного тестирования пользователь должен выбрать режим набора текста («На время», «Обучение», «Итоговое тестирование») и приступить к выполнению упражнения.

В режиме «Обучение» пользователь должен выбрать упражнение из предложенного списка, он не будет ограничен по времени выполнения и количеству ошибок в упражнении. По окончанию выполнения упражнения пользователю представляется результаты и рекомендации по дальнейшему обучению.

В режиме «На время» пользователь ограничен по времени выполнения упражнения, при выполнении которого должна учитываться средняя скорость набора символов.

Режим «Итоговое тестирование» позволяет пользователю определить уровень приобретенного навыка скоропечатания. При превышении пользователем максимального количества ошибок итоговое тестирование должно быть прервано, а пользователю сообщено о превышении максимального количества ошибок.

Генерация упражнения должна проводиться в двух режимах: ручном, когда упражнение создается администратором системы, и автоматическом, когда упражнение генерируется системой по заданным параметрам (тип упражнения и уровень сложности выполнения упражнения).

В системе также должна быть реализована возможность получения справочной информации о самой системе, ее возможностях и теоретических основах десятипальцевого слепого метода набора текста.

Таким образом, система должна выполнять следующие функции:

1. общесистемные:
2. авторизация пользователя в системе;
3. аутентификация пользователя в системе;
4. настройка интерфейса пользователя в зависимости от роли;
5. визуализация процессов работы с тренажером;
6. выдача справочной информации о системе;
7. ведение статистики;
8. функции администратора:
9. работа с учетными данными пользователей:

* добавление пользователя;
* удаление пользователя;
* просмотр учетных записей;
* редактирование данных пользователя;

1. работа со статистикой с возможностью фильтрации данных по упражнениям или обучаемым;
2. создание упражнений;
3. функции обучаемого:
4. выбор сложности;
5. выбор упражнения;
6. просмотр статистики;
7. выполнение упражнения.
8. Проектирование системы

2.1 Структурная схема системы

При разработке структурной схемы используется методология структурного проектирования, в основе которой лежит алгоритмическая декомпозиция и иерархия вида «часть-целое», учитывающая, что внутренние связи элементов внутри подсистем сильнее, чем связь между подсистемами. Декомпозиция системы может повторяться многократно, вплоть до уровня конкретных процедур, при этом должна быть обеспечена целостность системы, а все составляющие компоненты взаимоувязаны. Для этого используются такие принципы разработки, как «сверху-вниз», «разделяй и властвуй», «иерархическое упорядочивание» и другие.

Система должна представлять собой совокупность элементов (объектов, субъектов), находящихся между собой в определенной зависимости и составляющих некоторое единство (целостность), направленное на достижение определенной цели.

Система может являться элементом другой системы более высокого порядка (надсистема) и включать в себя системы более низкого порядка (подсистемы). То есть систему можно рассматривать как набор подсистем, организованных для достижения определенной цели и описанных с помощью набора моделей (возможно, с различных точек зрения), а подсистему – как группу элементов, часть которых составляет спецификацию поведения, представленного другими ее составляющими [13].

На рисунке 2.1 приведена структурная схема системы.

В состав системы входят следующие подсистемы:

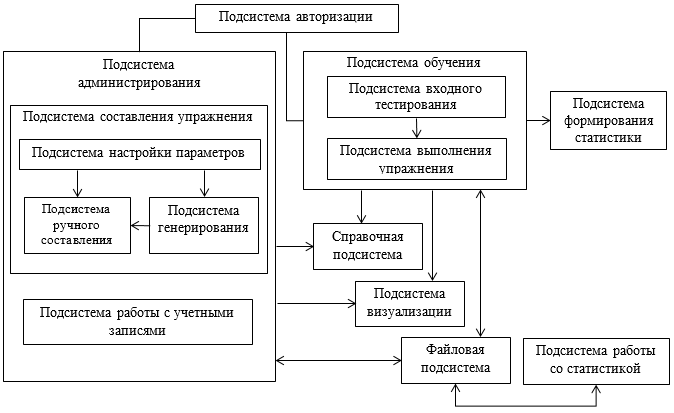
1. *Подсистема авторизации*, которая отвечает за авторизацию пользователя в системе и проверку системой правильности введенных данных;
2. *Подсистема администрирования*, в состав которой входят:
3. *подсистема работы с учетными записями*, которая позволяет добавлять и удалять пользователей, редактировать и просматривать данные пользователей;
4. *подсистема составления упражнения*,которая состоит из следующий подсистем:

* *подсистема настройки параметров*, которая отвечает за выбор уровня сложности и тип составляемого упражнения;
* *подсистема генерирования,* которая отвечает за автоматический режим составления упражнения;
* *подсистема ручного составления*,которая позволяет администратору составлять упражнения, в зависимости от выбранных параметров;

1. *Подсистема обучения*, в состав которой входят:

* *подсистема выполнения упражнения*,которая отвечает за выбор пользователем типа упражнения, в зависимости от уровня сложности, и за процесс выполнения упражнения;
* *подсистема входного тестирования,* которая позволяет определить начальный уровень владения пользователем слепого метода печати и определить начальные настройки сложности для каждого пользователя;

1. *Файловая подсистема*, которая отвечает за хранение учетных записей и настроек пользователей, статистики выполнения и текстов упражнений в файлах, а также отвечает за добавление и получение этих данных;
2. *Подсистема формирования статистики*, которая отвечает за ведение статистики прохождения упражнений конкретным пользователем;
3. *Подсистема работы со статистикой,* которая позволяет обучаемому просматривать свои результаты, а администратору – просматривать расширенную статистику обучения пользователей;
4. *Подсистема визуализации*, которая отвечает за визуализацию процессов работы с тренажером;
5. *Справочная подсистема*, которая содержит сведения о системе (руководство пользователю) и ее разработчиках.

  
Рисунок 2.1 – Структурная схема системы

2.2 Спецификация требований к проектируемой системе

Разработка программной системы начинается с формулировки требований, в которой, исходя из пожеланий к ней, должен быть получен документ, в котором должно быть достаточно точно определены требования к разработчикам. Этот документ называется спецификацией требований. Иными словами, спецификация – это документ, который понятно и точно описывает существенные технические требования для объектов, материалов или операций.

Спецификации помогают устранить дублирование и несоответствия, позволяют точно оценить необходимые действия и ресурсы, выступают в качестве согласующего и справочного документов о внесенных изменениях.

Функциональная спецификация не определяет операции, происходящие внутри данной системы, а также каким образом будет реализована её функция. Вместо этого она рассматривает взаимодействие с внешними агентами (например, персонал, использующий программное обеспечение, периферийные устройства компьютера или другие компьютеры), которые могут "следить", взаимодействуя с системой.

Функциональная спецификация включает в себя:

* перечень всех функций системы с привязкой их к конкретной подсистеме и к информационной среде (входные и выходные данные), фрагмент функциональной спецификации приведен в таблице 1;
* перечень исключительных ситуаций и реакцию системы на их возникновение, при необходимости приводится перечень ошибок, которые могут возникать в системе и соответствующие им системные сообщения, примеры исключительных ситуаций приведены в таблице 2.

Функциональная спецификация должна в полном объёме отображать информационные связи проектируемой системы как с внешним миром, так и между подсистемами. При необходимости расписываются информационные связи для сложных подсистем (спецификация второго уровня).

Исключительная ситуация – это ситуация, при которой система не может выполнить возложенных на нее функций или которая может привести к денормализации работы системы [13].

| Название подсистемы | Название функции | Информационная среда | | | |
| --- | --- | --- | --- | --- | --- |
| Входные данные | | Выходные данные | |
| Название | Тип, ограничения | Название | Тип, ограничения |
| 1 Справочная | 1.1 Выдать сведения о разработчиках и о системе | Сведения о разработчиках системы и о системе | Текстовый | Визуальное отображение информации | - |
| 1.2 Выдать сведения о возможностях системы | Файл справки | Текстовый, \*HTML | Визуальное отображение информации | - |
| Код ошибки | Целое |
| 2 Авторизации | 2.1. Ввести логин | Набор допустимых символов | Латинский алфавит, цифры | Логин | Строка |
| Код ошибки | Целое |
| 2.2 Ввести пароль | Набор допустимых символов | Латинский алфавит, цифры | Пароль | Строка |
| Код ошибки | Целое |
| 2.3 Подтвердить пароль | Пароль1 | Строка | Результат сравнения пароля1 и пароля2 | Логический |
| Пароль2 | Строка |
| 2.4 Ввести фамилию | Набор допустимых символов | Русский алфавит, пробел, ‘-‘ | Фамилия | Строка |
| Код ошибки | Целое |
| 2 Авторизации | 2.5 Ввести имя | Набор допустимых символов | Русский алфавит | Имя | Строка |
| Код ошибки | Целое |
| 2.6 Ввести отчество | Набор допустимых символов | Русский алфавит | Отчество | Строка |
| Код ошибки | Целое |
| 2.7 Ввести e-mail | Набор допустимых символов | Латинский алфавит, цифры, ‘@’, ‘.’, ‘\_’, ‘-’ | E-mail | Строка |
| Код ошибки | Целое |
| 2.8 Аутентифицировать | Логин  Пароль | Строка  Строка  (поля объекта типа «User») | Результат прохождения аутентификации | Логический |
| 2.9 Зарегистрировать | Данные пользователя | Объект типа «User» | Запись в файле с данными пользователей | Текстовый, USERS.XML |
| Код ошибки | Целое |
| 3 Составления упражнений | 3.1 Выбрать режим/тип упражнения | Список режимов упражнения | «Входное тестирование»,  «На время», «Обучение», «Итоговое тестирование» | Выбранный режим упражнения | Целое,  0..3 |
| 3.2 Выбрать сложность упражнения | Список уровней сложности | «Новичок», «Любитель», «Профессионал» | Выбранный уровень сложности упражнений | Целое,  0..2 |
| 3.3 Выбрать упражнение из списка | Список из N доступных упражнений | Список  объектов типа «Упражнение» | Номер упражнения | Целое,  1..N |
| 3.4 Удалить упражнение | Номер упражнения | Целое | Обновленный файл с упражнениями | Текстовый, EXERCISE.XML |
| 4 Настройки параметров | 4.1 Ввести название упражнения | Набор допустимых символов | Русский алфавит, пробел, ‘-‘ | Название | Строка |
| 4.2 Задать сложность упражнения | Допустимые уровни сложности | «Новичок», «Любитель», «Профессионал» | Текущий уровень сложности | Целое |
| 4 Настройки параметров | 4.3 Выбрать режим/тип упражнения | Список режимов упражнения | «Входное тестирование»,  «На время», «Обучение», «Итоговое тестирование» | Выбранный режим упражнения | Целое,  0..3 |
| 4.4 Задать минимальную скорость набора символов (для режима «На время») | Минимальная средняя скорость набора символов | Целое,  150 | Текущая средняя скорость набора символов | Целое |
| 4.5 Задать допустимое количество ошибок в процентном соотношение к длине упражнения (для итогового тестирования) | Диапазон количества ошибок: | Целое | Текущее допустимое количество ошибок | Целое |
| минимальное  максимальное | 1  10 |
| 5 Ручного составления упражнения | 5.1 Ввести текст упражнения | Набор допустимых символов | Русский алфавит, цифры, знаки препинания | Текст упражнения | Текстовый |
| 5.2 Создать упражнение | Название | Строка | Запись в файле с упражнениями | Файл EXERCISE.XML |
| Уровень сложности | Целое |
| Режим | Целое |
| Средняя скорость | Целое |
| Количество ошибок | Целое |
| Текст упражнения | Текстовый |
| 6 Генерирования | 6.1 Задать длину упражнения | Диапазон количества символов в упражнение | Целое | Текущее значение длины упражнения | Целое |
| минимальное  максимальное | 30  1000 |
| 6.2 Задать символы, участвующие в упражнении | Набор допустимых символов | Строка | Множество используемых символов | Строка |
| 6 Генерирования | 6.3 Сгенерировать текст упражнения | Название | Строка | Запись в файле с упражнениями | Файл EXERCISE.XML |
| Уровень сложности | Целое |
| Режим | Целое |
| Средняя скорость | Целое |
| Количество ошибок | Целое |
| Множество символов | Строка |
| 7 Работы с учетными записями | 7.1 Ввести критерий для поиска | Набор допустимых символов | Латинский и русский алфавит, цифры | Критерий поиска | Строка |
| 7.2 Найти пользователей по заданному критерию | Критерий поиска Список всех пользователей | Строка  Список объектов типа «User» | Список найденных пользователей | Список объектов типа «User» |
| 7.3 Выбрать пользователя | Список пользователей | Список объектов типа «User» | Номер пользователя | Целое |
| 7.4 Ввести/изменить логин | Набор допустимых символов | Латинский алфавит, цифры | Логин | Строка |
| Код ошибки | Целое |
| 7 Работы с учетными записями | 7.5 Ввести/изменить пароль | Набор допустимых символов | Латинский алфавит, цифры | Пароль | Строка |
| Код ошибки | Целое |
| 7.6 Ввести/изменить  Фамилию | Набор допустимых символов | Русский алфавит, пробел, ‘-‘ | Фамилия | Строка |
| Код ошибки | Целое |
| 7.7 Ввести/изменить  Имя | Набор допустимых символов | Русский алфавит | Имя | Строка |
| Код ошибки | Целое |
| 7.8 Ввести/изменить отчество | Набор допустимых символов | Русский алфавит | Отчество | Строка |
| Код ошибки | Целое |
| 7.9 Ввести/изменить  e-mail | Набор допустимых символов | Латинский алфавит, цифры, ‘@’, ‘.’, ‘\_’, ‘-’ | E-mail | Строка |
| Код ошибки | Целое |
| 7.10 Добавить пользователя | Данные пользователя | Массив строк | Пользователь | Объект типа «User» |
| 7.11 Сохранить данные пользователей в системе | Список пользователей | Список объектов типа «User» | Файл с пользователями | USERS.XML |
| 7 Работы с учетными записями | 7.12 Удалить пользователя | Номер пользователя | Целое | Список пользователей | Список объектов типа «User» |
| Список пользователей | Список объектов типа «User» |
| 8 Подсистема входного тестирования | 8.1 Загрузить упражнение | Имя файла | Строка, EXERCISE.XML | Текст упражнения | Текстовый |
| 8.2 Ввести следующий символ текста упражнения | Текущий символ | Символ | Результат сравнения текущего и введенного символов | Логический |
| Введенный символ | Символ |
| 8 Подсистема входного тестирования | 8.3 Вычислить количество напечатанных символов | Количество напечатанных символов | Целое | Текущее количество напечатанных символов | Целое |
| Результат сравнения текущего и введенного символов | Логический |
| Характеристики уровней | Двумерный массив целых чисел |
| 8.4 Вычислить количество допущенных ошибок | Кол-во ошибок | Целое | Текущее количество допущенных ошибок | Целое |
| Результат сравнения текущего и введенного символов | Логический |
| 8.5 Вычислить время прохождения упражнения | Время начала прохождения упражнения | DateTime | Текущее время прохождения упражнения (в сек) | Целое |
| Текущее время | DateTime |
| 8 Подсистема входного тестирования | 8.6 Вычислить среднюю скорость печати | Длина упражнения | Целое | Скорость печати (в симв/мин) | Целое |
| Время прохождения (в сек) | Целое |
| 8.7 Определить уровень ученика | Кол-во ошибок | Целое | Уровень | Целое |
| Время прохождения (в сек) | Целое |
| Скорость печати (в симв/мин) | Целое |
| 9 Подсистема выполнения упражнения | 9.1 Загрузить упражнения | Имя файла | Строка, EXERCISE.XML | Список упражнений | Список объектов типа «Упражнение |
| 9.2 Выбрать упражнение для выполнения | Список упражнений | Список объектов типа «Упражнение» | Номер упражнения | Целое |
| 9.3 Ввести следующий символ текста упражнения | Текущий символ | Символ | Результат сравнения текущего и введенного символов | Логический |
| Введенный символ | Символ |
| 9 Подсистема выполнения упражнения | 9.4 Вычислить количество напечатанных символов | Количество напечатанных символов | Целое | Текущее количество напечатанных символов | Целое |
| Результат сравнения текущего и введенного символов | Логический |
| 9.5 Вычислить количество допущенных ошибок | Кол-во ошибок | Целое | Текущее количество допущенных ошибок | Целое |
| Результат сравнения текущего и введенного символов | Логический |
| 9.6 Вычислить время прохождения упражнения | Время начала прохождения упражнения | DateTime | Текущее время прохождения упражнения (в сек) | Целое |
| Текущее время | DateTime |
| 9.7 Вычислить скорость печати | Длина упражнения | Целое | Скорость печати (в симв/мин) | Целое |
| Время прохождения (в сек) | Целое |
| 9 Подсистема выполнения упражнения | 9.8 Сравнить скорость печать с заданной минимальной скоростью печати (для режима «На время») | Скорость печати (в симв/мин) | Целое | Результат сравнения скоростей | Логический |
| Минимальная скорость печати (в симв/мин) | Целое |
| 9.9 Сравнить количество ошибок с допустимым количеством ошибок (для режима «Итоговое тестирование») | Количество ошибок | Целое | Результат сравнения количества ошибок и допустимого количества ошибок | Логический |
| Длина упражнения | Целое |
| Допустимое количество ошибок | Целое |
| 10 Подсистема формирования статистики | Сохранить результаты прохождения упражнения | Логин пользователя | Строка | Запись в файле со статистикой | Файл STATISTIC.XML |
| Дата | DateTime |
| Номер упражнения | Целое |
| Сложность | Целое |
| Режим | Целое |
| Количество ошибок | Целое |
| Время прохождения | Целое |
| Скорость печати | Целое |
| 11 Подсистема работы со статистикой | 11.1 Ввести критерий для поиска пользователей | Набор допустимых символов | Латинский алфавит, цифры | Критерий поиска  пользователей | Строка |
| 11.2 Найти пользователей по заданному критерию | Критерий поиска пользователей | Строка | Список найденных пользователей | Список строк (полей объекта типа «User») |
| Список логинов всех пользователей | Список строк (полей объектов типа «User» |
| 11 Подсистема работы со статистикой | 11.3 Выбрать пользователя | Список логинов пользователей | Список строк (полей объектов типа «User») | Список логин выбранных пользователя | Строка  (поле объекта типа «User») |
| 11.4 Выбрать режим упражнения | Список возможных режимов упражнения | «Все», «На время», «Обучение», «Итоговое тестирование» | Выбранные режимы упражнения | Строка |
| 11.5 Ввести критерий поиска упражнения | Набор допустимых символов | Русский алфавит, цифры | Критерий поиска упражнения | Строка |
| 11.6 Найти упражнение по заданному критерию | Критерий поиска упражнений | Строка | Список найденных упражнений | Список объектов типа «Exercise» |
| Список всех пользователей | Список объектов типа «Exercise» |
| 11.7 Выбрать упражнение | Список упражнений | Список объектов типа «Exercise» | Номер упражнения | Целое |
| 11 Подсистема работы со статистикой | 11.6 Ввести начало периода времени | Допустимый диапазон времени: | DateTime | Дата1 | Временной,  DateTime |
| нижний предел | 1.09.2014 |
| верхний предел | Текущая дата |
| 11.7 Ввести конец периода времени | Допустимый диапазон времени: | DateTime | Дата2 | Временной,  DateTime |
| нижний предел | Дата1 |
| верхний предел | Текущая дата |
| 11.7 Отфильтровать статистику | Список логинов пользователей | Список строк (полей объекта типа «User) | Выводимая статистика | Массив объектов типа «Statistic» |
| Режимы | Строка |
| Список упражнений | Список номеров упражнений (полей объектов типа «User») |
| Начало периода времени | DateTime |
| Конец периода времени | DateTime |
| 12 Файловая подсистема | 12.1 Загрузить упражнения | Имя файла | Строка, EXERCISE.XML | Упражнения | Список объектов типа «Exercise» |
| Код ошибки | Целое |
| 12.2 Загрузить пользователей | Имя файла | Строка, USER.XML | Пользователи | Список объектов типа «User» |
| Код ошибки | Целое |
| 12.3 Загрузить статистику | Имя файла | Строка, STATISTIC.XML | Статистика | Список объектов типа «Statistic» |
| Код ошибки | Целое |
| 13 Подсистема визуализации | 13.1 Отобразить сведения о пользователе на экране | Данные пользователя | Объект типа пользователь | Визуальное отображение | - |
| 13.2 Отобразить список упражнений | Уровень сложности | Целое |
| Режим | Целое |
| Список упражнений | Файл с упражнениями .XML |
| 13.3 Отобразить текст упражнения | Текст упражнения | Текст |
| 13.4 Отобразить количество напечатанных символов | Количество напечатанных символов | Целое |
| 13.5 Отобразить количество допущенных ошибок | Количество допущенных ошибок | Целое |
| 13 Подсистема визуализации | 13.6 Отобразить время прохождения упражнения | Время прохождения упражнения (в сек) | Целое | Визуальное отображение | - |
| 13.7 Отобразить введенный символ | Введенный символ | Символ |
| 13.8 Выделить текущий символ в тексте упражнения | Позиция текущего символа | Целое |
| 13.9 Отобразить текущий символ на виртуальной клавиатуре | Текущий символ | Символ |
| 13.10 Завершить упражнение (для режимов «На время» и «Итогового тестирования») | Результат сравнения скоростей или результат сравнения ошибок | Логический |
| 13 Подсистема визуализации | 13.11 Отобразить результат прохождения упражнения | Количество ошибок | Целое | Визуальное отображение | - |
| Время прохождения упражнения | Целое |
| Скорость печати | Целое |
| Уровень пользователя (для «Итоговое тестирование») | Целое |
| 13.12 Отобразить статистику | Статистика | Массив объектов типа «Statistic» |

Таблица 2 – Перечень исключительных ситуаций

|  |  |  |
| --- | --- | --- |
| Название  подсистемы | Название исключительной ситуации | Реакция системы |
| 1 Справочная | 1.1 Не возможно открыть файл справки | Выдача сообщения «Файл справки поврежден» |
| 1.2 Не возможно найти файл справки | Выдача сообщения «Отсутствует файл справки» |
| 2 Файловая | 2.1 Файл с заданным именем не существует | Создание файла и запись данных в файл |
| 2.2 Попытка извлечь данные из файла, структура которого изменена | Создание файла и запись данных в файл |
| 2.3 Попытка обратиться к файлу, структура которого изменена | Создание файла и запись данных в файл |

2.3 Разработка прототипа интерфейса системы

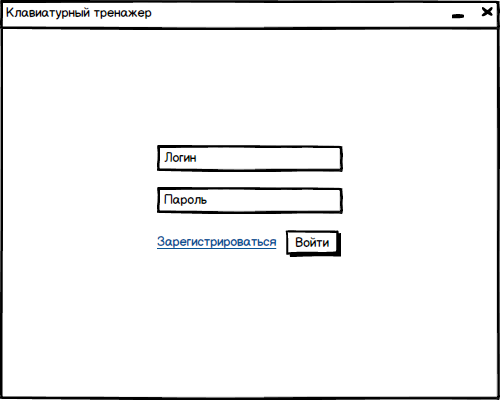
Процесс создания интерфейса начинается с определения целей проекта, а также внутренних и внешние обстоятельств, которые вы должны принять во внимание. Для того чтобы правильно расставить приоритеты, необходимо учитывать:

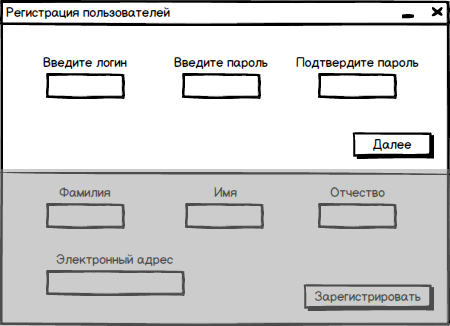
* опыт работы пользователей с компьютером, типовые ситуации использования;
* какая информация необходима и когда, какие результаты должны быть получены;
* технологию разработки и платформа, на которой будут работать пользователи.

Начальной фазой разработки интерфейса является разработка прототипов экранных форм [13].

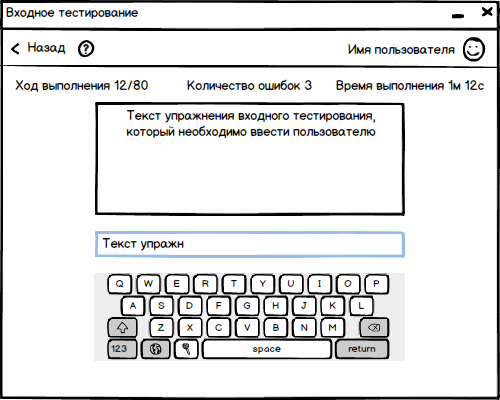
На рисунках 2.2 – 2.17 изображены прототипы экранных форм клавиатурного тренажера.

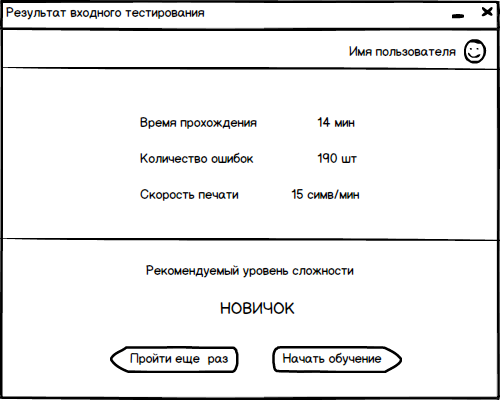
Из формы входа в систему (рисунок 2.2) можно перейти в форму регистрации (рисунок 2.3) при нажатии ссылки «Зарегистрироваться».

  
Рисунок 2.2 – Прототип экранной формы входа в систему

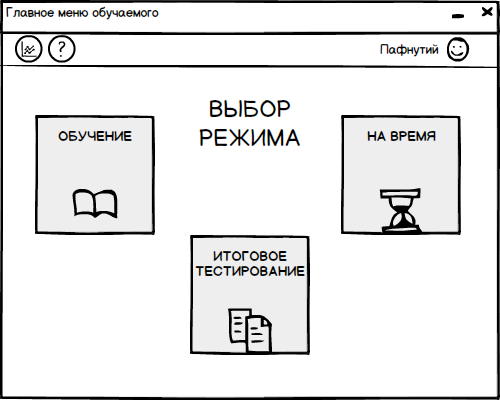
  
Рисунок 2.3 – Прототип экранной формы регистрации пользователей

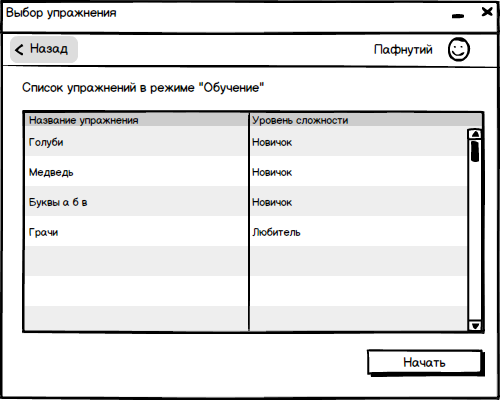
Если пользователь войдет в систему с ролью администратора, то будет открыта форма главного меню администратора (рисунок 2.11). Если пользователь войдет в систему с ролью обучаемого, то, в зависимости от того, проходил ли он входное тестирование, будет открыта либо форма входного тестирования (рисунок 2.4), либо форма главного меню обучаемого (рисунок 2.6).

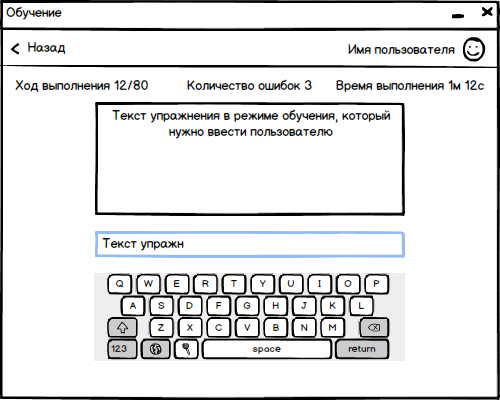
  
Рисунок 2.4  – Прототип экранной формы входного тестирования

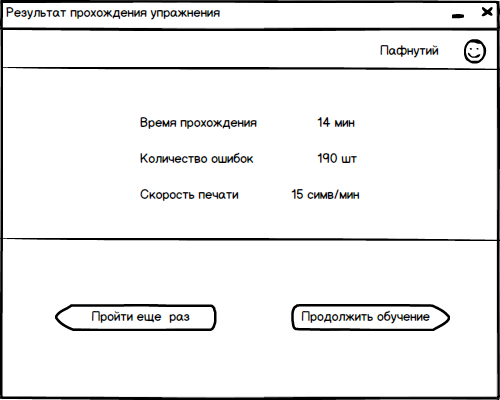
  
Рисунок 2.5 – Прототип экранной формы результата входного тестирования

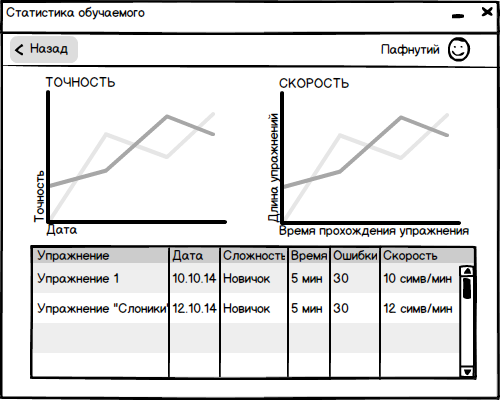
В главном меню обучаемого (рисунок 2.6) при выборе каждого режима будет открыта форма выбора упражнения (рисунок 2.7), список упражнений в которой будет соответствовать выбранному режиму.

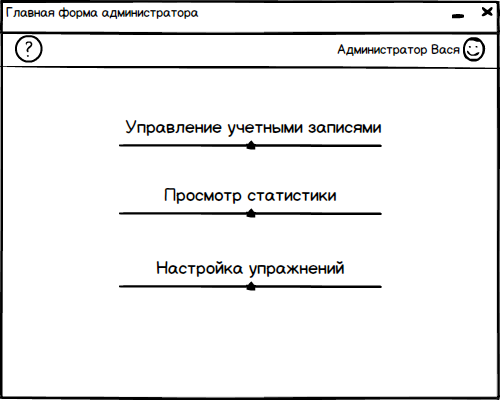
  
Рисунок 2.6 – Прототип экранной формы главного меню обучаемого

  
Рисунок 2.7 – Прототип экранной формы выбора упражнения в режиме «Обучение»

  
Рисунок 2.8 – Прототип экранной формы прохождения упражнений в режиме «Обучение»

  
Рисунок 2.9 – Прототип экранной формы результата прохождения упражнений

  
Рисунок 2.10 – Прототип экранной формы статистики обучаемого

  
Рисунок 2.11 – Прототип экранной формы главного меню администратора

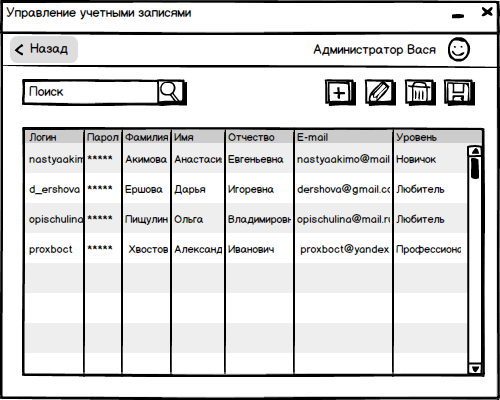
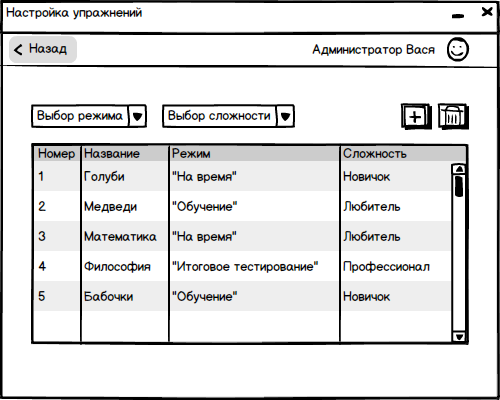
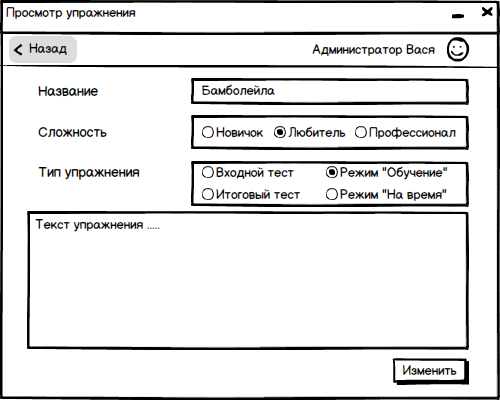
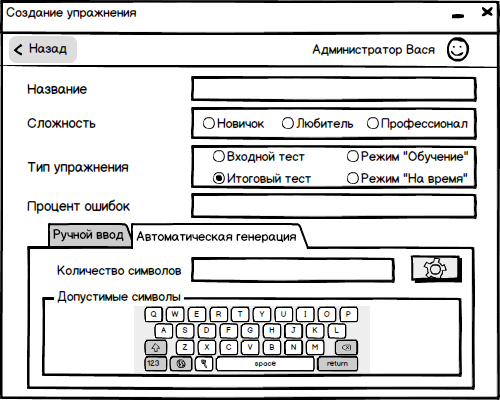
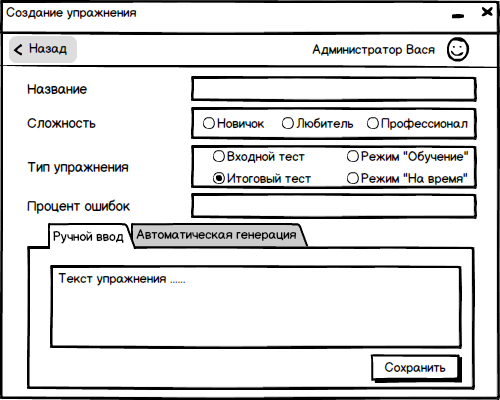
  
Рисунок 2.12 – Прототип экранной формы управления учетными записями

  
Рисунок 2.13 – Прототип экранной формы работы с расширенной статистикой

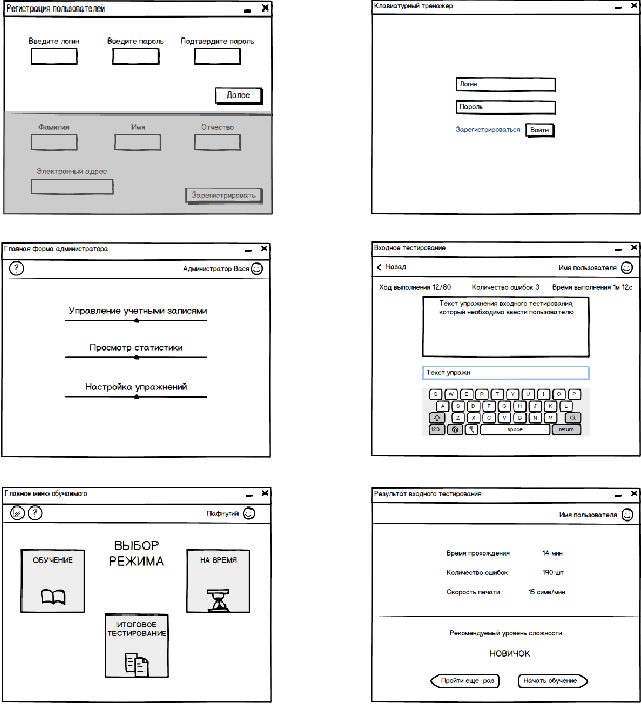
  
Рисунок 2.14 – Прототип экранной формы настройки упражнений

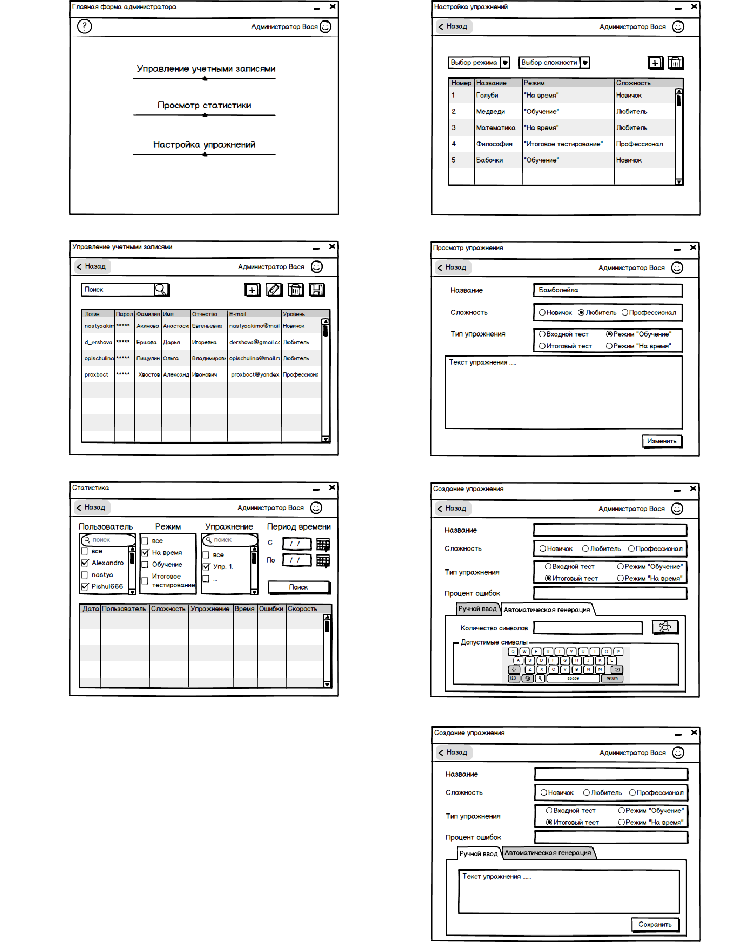
  
Рисунок 2.15 – Прототип экранной формы просмотра упражнения

  
Рисунок 2.16 – Прототип экранной формы создания упражнения с автоматической генерацией текста

  
Рисунок 2.17 – Прототип экранной формы создания упражнения с ручным вводом текста упражнения

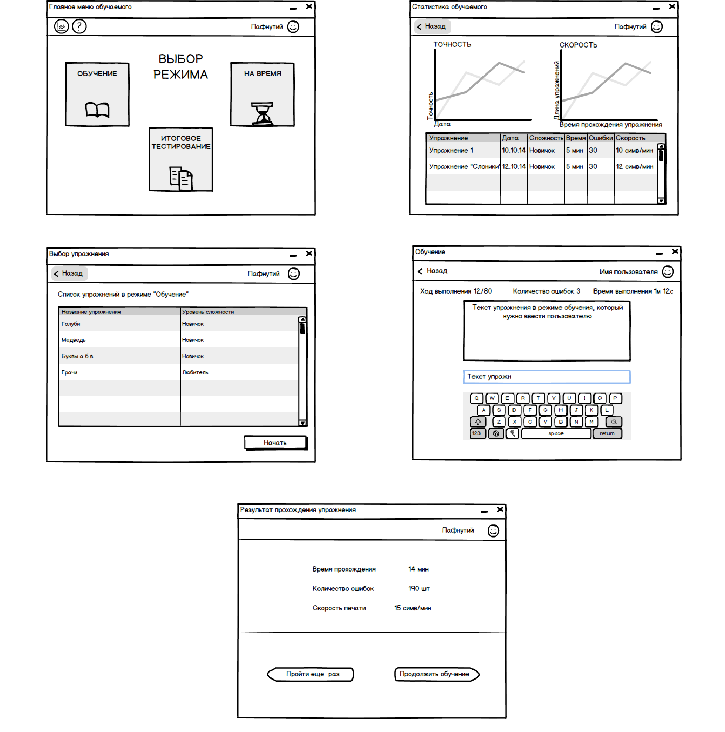
Навигационная модель показывает, как необходимо распределять функции или задачи между окнами создаваемой программы, она определяет, как пользователи смогут перемещаться как между различными задачами, так и внутри отдельной задачи. На рисунках 2.18 – 2.20 приведена навигационная модель для разрабатываемой системы «Клавиатурный тренажер».

  
Рисунок 2.18 – Навигационная модель системы   
(общая часть)

  
Рисунок 2.19 – Навигационная модель системы   
(режим администратора)

Справка

Двойной клик

  
Рисунок 2.20 – Навигационная модель системы   
(режим обучаемого)

Переход на форму выбора, с соответствующим списком упражнений

Справка

2.4 Информационно-логический проект системы

2.4.1 Язык UML

Унифицированный язык моделирования (Unified Modeling Language – UML) – это стандартный инструмент для разработки «чертежей» программного обеспечения. Его можно использовать для визуализации, спецификации, конструирования и документирования артефактов программных систем. Язык UML предоставляет стандартный способ написания проектной документации на системы, включая концептуальные аспекты, такие как бизнес-процессы и функции системы, а также конкретные аспекты, такие как выражения языков программирования, схемы баз данных и повторно используемые компоненты ПО.

В нотации языка UML определены следующие виды канонических диаграмм:

* вариантов использования (use case diagram);
* классов (class diagram);
* кооперации (collaboration diagram);
* последовательности (sequence diagram);
* состояний (statechart diagram);
* деятельности (activity diagram);
* компонентов (component diagram);
* развертывания (deployment diagram) [13].

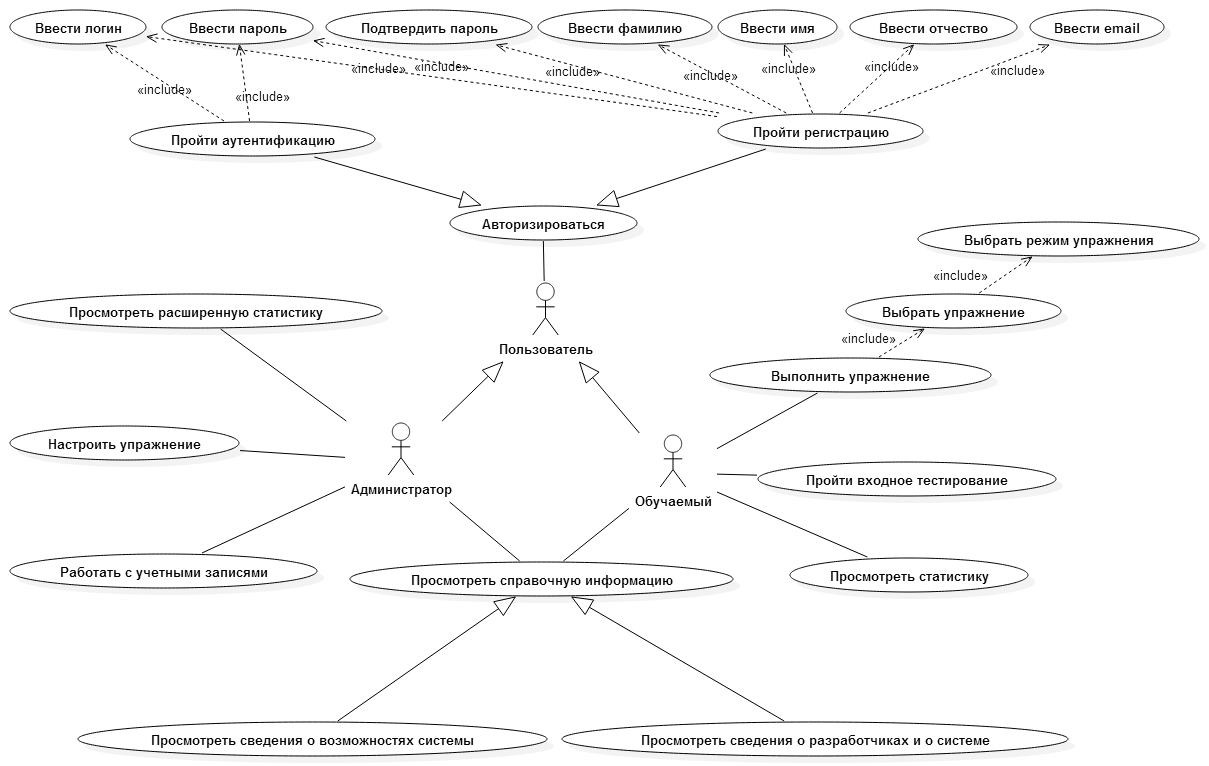
2.4.2 Диаграмма вариантов использования

Диаграмма вариантов использования описывает функциональное назначение системы и является исходным концептуальным представлением в процессе ее проектирования и разработки. На ней изображаются отношения между актерами и вариантами использования [13].

Актер – внешняя по отношению к ПС сущность, которая может взаимодействовать с системой. Актер – это не конкретный человек или устройство, а роль (должностная обязанность), в которой он выступает по отношению к программной системе [14].

Вариант использования – внешняя спецификация последовательности действий, которые система или другая сущность могут выполнять в процессе взаимодействия с актерами (он определяет набор действий, совершаемый системой при диалоге с актером) [13].

На рисунке 2.21 приведен фрагмент диаграммы вариантов использования системы (он соответствует функциональной спецификации, приведенной в таблице 1, и дополняет ее новыми функциями).

  
Рисунок 2.21 –Диаграмма вариантов использования   
(общая часть)

Перед тем как войти в систему, зарегистрированный пользователь должен пройти процедуру авторизации (ввести логин и пароль). Незарегистрированный пользователь должен перед авторизацией зарегистрироваться в системе. Система проверит введенные данные и настроит интерфейс пользователя на соответствующую роль.

В системе должны быть предусмотрены две роли пользователя: администратор и обучаемый. Любой пользователь системы должен иметь возможность просмотреть справочную информацию (сведения о разработчиках и сведения о самой системе).

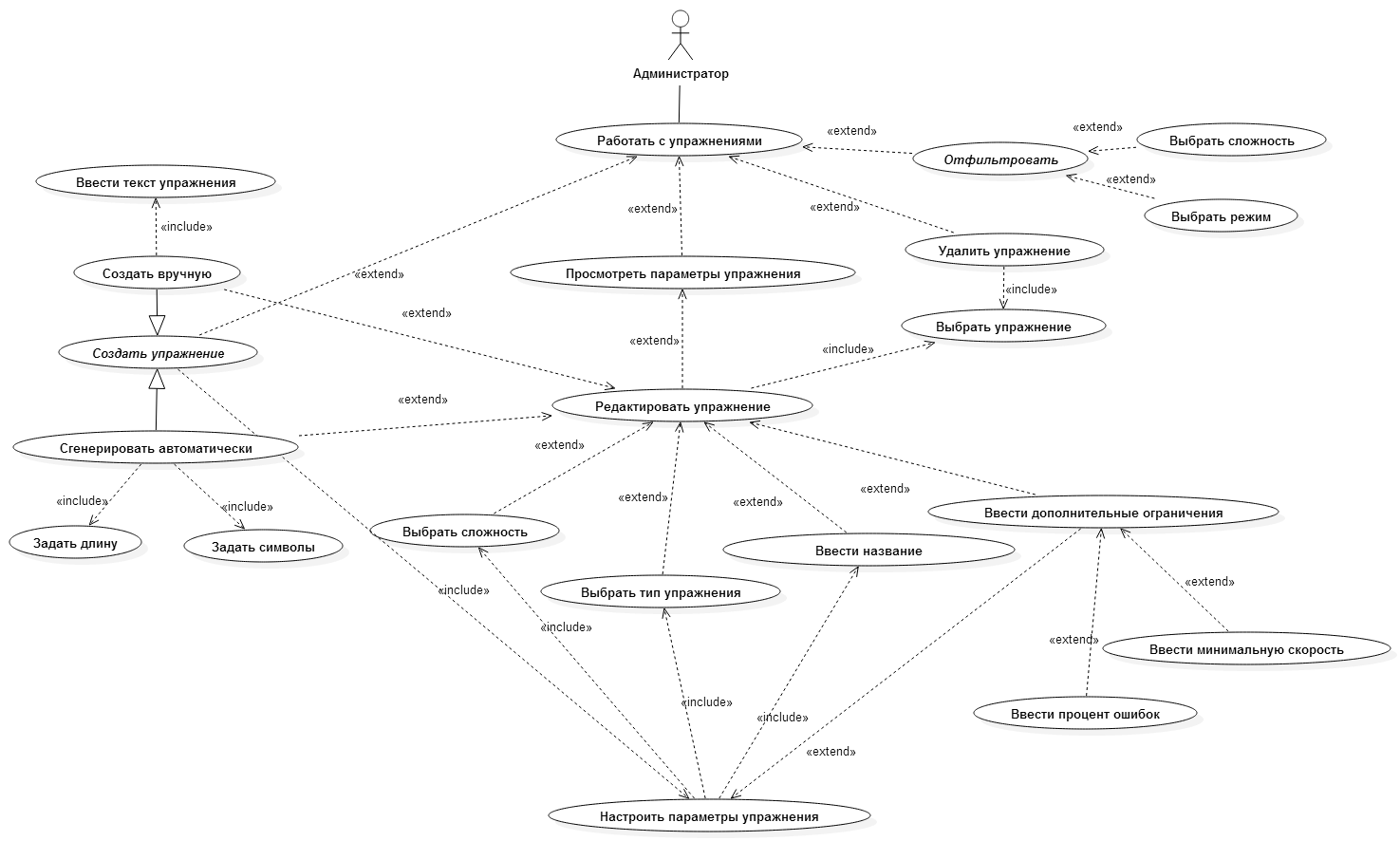
Администратор может работать с учетными записями пользователей, просматривать расширенную статистику пользователей и работать с упражнениями.

Обучаемый в начале обучения должен пройти входное тестирование, после чего он может проходить упражнения и просматривать собственную статистику.

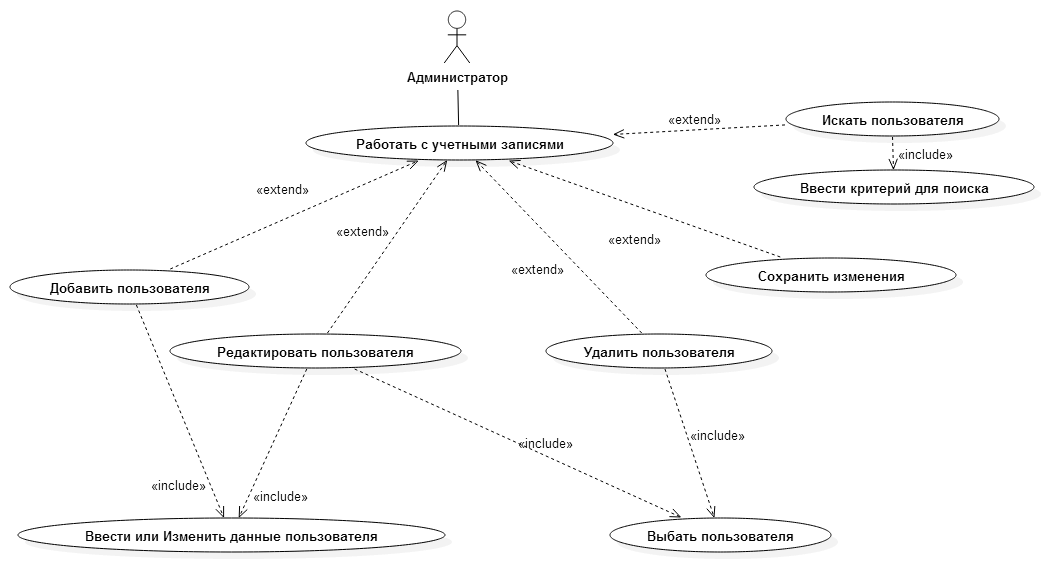
На рисунке 2.22 представлен вариант использования «Просмотреть расширенную статистику», согласно которому администратор может провести фильтрацию расширенной статистики по конкретному пользователю, упражнению, режиму упражнений или по периоду времени.

  
Рисунок 2.22 –Диаграмма варианта использования «Просмотреть расширенную статистику»

На рисунке 2.23 представлен вариант использования «Работать с упражнениями». Администратор может применить фильтрацию к списку всех упражнений по сложности или режиму, просматривать упражнения, удалять, редактировать или создавать новые упражнения. Для создания упражнения администратору потребуется настроить параметры упражнения – ввести название, текст, сложность и режим упражнения (при выборе режимов упражнения «На время» или «Итоговое тестирование» необходимо будет ввести минимальную скорость прохождения упражнения или максимальное количество ошибок, соответственно). Администратор может задавать текст упражнения двумя способами – вручную или автоматически, задав длину и участвующие в автоматической генерации символы.

  
Рисунок 2.23 – Диаграмма варианта использования «Работать с упражнениями»

На рисунке 2.24 представлена диаграмма варианта использования «Работать с учетными записями». Администратор может осуществлять поиск учетных записей по какому-то критерию, добавить новых пользователей в систему, редактировать учетные записи зарегистрированных пользователей или удалять пользователей из системы.

  
Рисунок 2.24 – Диаграмма варианта использования «Работать с учетными записями»

Приведенные диаграммы вариантов использования можно дополнить с помощью текстовых сценариев, которые уточняют или детализируют последовательность действий, совершаемых системой при выполнении ее вариантов использования.

Сценарий – определенная последовательность действий, которая описывает действия актеров и поведение моделируемой системы в форме обычного текста [13].

Рассмотрим вариант использования «Работать с упражнениями» (прототип экранной формы приведен на рисунке 2.14).

**Вариант использования: Работать с упражнениями**

*Краткое описание.* Даёт возможность пользователю с правами администратора выбирать, просматривать, добавлять, удалять и изменять упражнения.

*Актант*. Администратор.

*Предусловия*. Компьютер пользователя включён, приложение запущено, на экране – приложение с главной формой, настроенной на права администратора. Пользователь нажимает на пункт меню «Настройка упражнений».

*Основной поток событий.*

1) На экране появляется форма настройки упражнений с двумя выпадающими списками, отвечающими за фильтрацию списка упражнений: поле «Выбор режима» и поле «Выбор сложности», кнопками «Добавить», «Удалить», «Свернуть», «Закрыть» и «Назад», списком упражнений. В верхней части формы выводится имя и роль пользователя.

2) Пользователь открывает упражнение из списка (два щелчка левой кнопки мыши).

А1: Щелкнута кнопка «Выход».

А2: Щелкнута кнопка «Свернуть».

А3: Щелкнута кнопка «Назад».

А4: Щелкнута кнопка «Добавить».

А5: Щелкнута кнопка «Удалить».

А6: Выбран один из режимов прохождения упражнения в выпадающем списке.

А7: Выбран один из уровней сложности упражнения в выпадающем списке.

А8: Выбрано упражнение из списка упражнений (один щелчок левой кнопки мыши).

3) Открывается форма просмотра параметров упражнения. Переход к варианту использования «Просмотреть параметры упражнения».

*Альтернативы*.

А1: Щелкнута кнопка «Выход».

А1.1 Система закрывает форму настройки упражнений. Отображается состояние экрана до запуска приложения. Вариант использования завершается.

А2: Щелкнута кнопка «Свернуть».

А2.1 Отображается состояние экрана до запуска приложения.

А3: Щелкнута кнопка «Назад».

А3.1 Открывается главная форма администратора. Любые изменения не сохраняются. Вариант использования завершается.

А4: Щелкнута кнопка «Добавить».

А4.1 Открывается форма создания упражнения. Переход к варианту использования «Создание упражнения».

А5: Щелкнута кнопка «Удалить».

А5.1 Открывается форма с сообщением об ошибке «Необходимо выбрать упражнение из списка для удаления».

А5.2 Пользователь нажимает кнопку «Закрыть», форма закрывается, переход к п.1 основного потока событий.

А6: Выбран один из режимов прохождения упражнения в выпадающем списке.

А6.1 На форме отображается отфильтрованный список по критерию режима прохождения упражнения.

А7: Выбран один из уровней сложности упражнения в выпадающем списке.

А7.1 На форме отображается отфильтрованный список по критерию режима прохождения упражнения.

А8: Выбрано упражнение из списка упражнений (один щелчок левой кнопки мыши).

А8.1 На форме отображается список с выделенным упражнением.

А8.1.1 Пользователь нажимает кнопку «Удалить», система удаляет упражнение из файла с упражнениями. Список упражнений обновляется. Переход к п. 1 главного потока событий.

*Постусловия*. При успешном завершении действий система сохраняет новое или отредактированное упражнение.

Рассмотрим вариант использования «Создать упражнение» (прототип экранной формы представлен на рисунках 2.16, 2.17).

**Вариант использования: Создать упражнение**

*Краткое описание.* Даёт возможность пользователю с правами администратора создавать упражнения после нажатия кнопки «Добавить» в окне формы настройки упражнения.

*Актант*. Администратор.

*Предусловия*. Компьютер пользователя включён, приложение запущено, на экране – приложение с формой создания упражнения. Нажата кнопка «Добавить».

*Основной поток событий.*

1) На экране появляется форма создания упражнения с кнопками «Свернуть», «Закрыть», «Назад», «Сохранить», поле ввода «Название», поля для выбора одного из уровней сложности и одного из типов упражнений, поле для ввода ограничения (в зависимости от выбранного типа упражнения, оно может отсутствовать), две вкладки для выбора способов создания упражнения: «Ручной ввод» и «Автоматическая генерация». В верхней части формы выводится имя и роль пользователя.

2) Пользователь вводит название упражнения, выбирает сложность и тип упражнения.

А1: Щелкнута кнопка «Выход».

А2: Щелкнута кнопка «Свернуть».

А3: Щелкнута кнопка «Назад».

А4: Щелкнута кнопка «Сохранить».

А5: Выбран тип упражнения «Итоговый тест».

А6: Выбран тип упражнения «На время».

3) Пользователь нажимает кнопку «Сохранить».

4) Система записывает созданное упражнение в файл с упражнениями. Переход к варианту использования «Просмотр параметров упражнения».

*Альтернативы.*

А1: Щелкнута кнопка «Выход».

А1.1 Система закрывает форму создания упражнения. Отображается состояние экрана до запуска приложения. Вариант использования завершается.

А2: Щелкнута кнопка «Свернуть».

А2.1 Отображается состояние экрана до запуска приложения.

А3: Щелкнута кнопка «Назад».

А3.1 Открывается форма настройки упражнений. Вариант использования завершается. Переход к варианту использования «Работа с упражнениями».

А4: Щелкнута кнопка «Сохранить».

А4.1 Открывается форма с сообщением об ошибке. Поля не заполнены.

А5: Выбран тип упражнения «Итоговый тест».

А5.1 На форме создания упражнения появляется поле для ввода ограничения: допустимый процент ошибок. Переход к варианту использования «Ввести процент ошибок».

А6: Выбран тип упражнения «На время».

А6.1 На форме создания упражнения появляется поле для ввода ограничения: минимальная скорость набора текста. Переход в варианту использования «Ввести минимальную скорость»

*Постусловия*. При успешном завершении действий система сохраняет созданное упражнение.

Рассмотрим вариант использования «Работать с учетными записями» для клавиатурного тренажера (прототип экранной формы приведен на рисунке 2.12).

**Вариант использования: Работать с учетными записями**

*Краткое описание.* Вариант использования позволяет администратору просматривать учетные записи пользователей, осуществлять поиск пользователя по заданному критерию, а также добавлять, редактировать, удалять и сохранять данные пользователей. Администратор на главной форме одинарным щелчком выбирает пункт меню «Управление учетными записями». Открывается окно управления учетными записями.

*Актант*. Администратор.

*Предусловия*. Компьютер пользователя включен, пользователь вошел в систему с ролью администратора, на экране открыта «Главная форма администратора».

*Основной поток событий.*

1) На экране появляется форма управления учетными записями с полем ввода «Поиск» и связанной с ним кнопкой «Поиск», кнопками «Назад», «Добавить», «Редактировать», «Удалить», «Сохранить», «Свернуть» и «Выход», а также отображается таблица, в которой хранятся данные ранее зарегистрированных пользователей, загруженные системой из файла. Данные в ячейках таблицы редактировать нельзя.

А1: Файл не найден или его структура повреждена.

2) Администратор вводит критерий поиска пользователя и нажимает кнопку «Поиск».

А2: Щелкнута кнопка «Назад».

А3: Щелкнута кнопка «Свернуть».

А4: Щелкнута кнопка «Выход».

А5: Щелкнута кнопка «Добавить».

А6: Щелкнута кнопка «Редактировать».

А7: Щелкнута кнопка «Удалить».

А8: Щелкнута кнопка «Сохранить».

3) Система проверяет введенный критерий поиска.

4) Система выводит в таблицу учетные записи тех пользователей, чьи данные удовлетворяют условию поиска.

А9: Ни одна учетная запись не удовлетворяет критерию поиска.

*Альтернативы.*

А1: Файл не найден или его структура повреждена.

А1.1 Система пересоздает файл, в котором будет содержаться только учетная запись администратора. Переход к п.1 основного потока событий.

А2: Щелкнута кнопка «Назад».

А2.1 Отображается «Главная форма администратора». Вариант использования успешно завершается.

А3: Щелкнута кнопка «Свернуть».

А3.1 Отображается состояние экрана до запуска приложения.

А4: Щелкнута кнопка «Выход».

А4.1 Система закрывает окно «Управление учетными записями» и завершает работу клавиатурного тренажера. Отображается состояние экрана до запуска приложения. Вариант использования успешно завершается.

А5: Щелкнута кнопка «Добавить».

А5.1 Переход к варианту использования «Добавить пользователя».

А6: Щелкнута кнопка «Редактировать».

А6.1 Переход к варианту использования «Редактировать пользователя».

А7: Щелкнута кнопка «Удалить».

А7.1 Переход к варианту использования «Удалить пользователя».

А8: Щелкнута кнопка «Сохранить».

А8.1 Переход к варианту использования «Сохранить изменения».

А9: Ни одна учетная запись не удовлетворяет критерию пользователя.

А9.1 Система выводит сообщение о том, что ни одного совпадения с заданным условием поиска не найдено. Переход к пункту 2 основного потока.

*Постусловия*. При успешном завершении на экране – окно «Главная форма администратора».

Рассмотрим вариант использования «Сохранить изменения» для клавиатурного тренажера (прототип экранной формы приведен на рисунке 2.12).

**Вариант использования: Сохранить изменения**

*Краткое описание.* Вариант использования позволяет администратору сохранять изменения учетных записей пользователей.

*Актант*. Администратор.

*Предусловия*. Компьютер пользователя включен, пользователь вошел в систему с ролью администратора, на экране открыта форма «Управление учетными записями». Администратор нажал на кнопку «Сохранить».

*Основной поток событий.*

1) Система проверяет введенные данные.

2) Система сохраняет все изменения с учетными записями пользователей в файл, сообщает администратору об успешном сохранении изменений. Переход к варианту использования «Работать с учетными записями».

А1: Данные пользователя введены неверно.

3) . Вариант использования успешно завершается.

А2: Не удалось сохранить изменения в файл.

*Альтернативы.*

А1: Данные пользователя введены неверно.

А1.1 Система выводит сообщение о том, что данные определенного пользователя содержат запрещенные символы и просит исправить учетную запись этого пользователя. Вариант использования завершается. Переход к варианту использования «Работать с учетными записями».

А2: Не удалось сохранить изменения в файл.

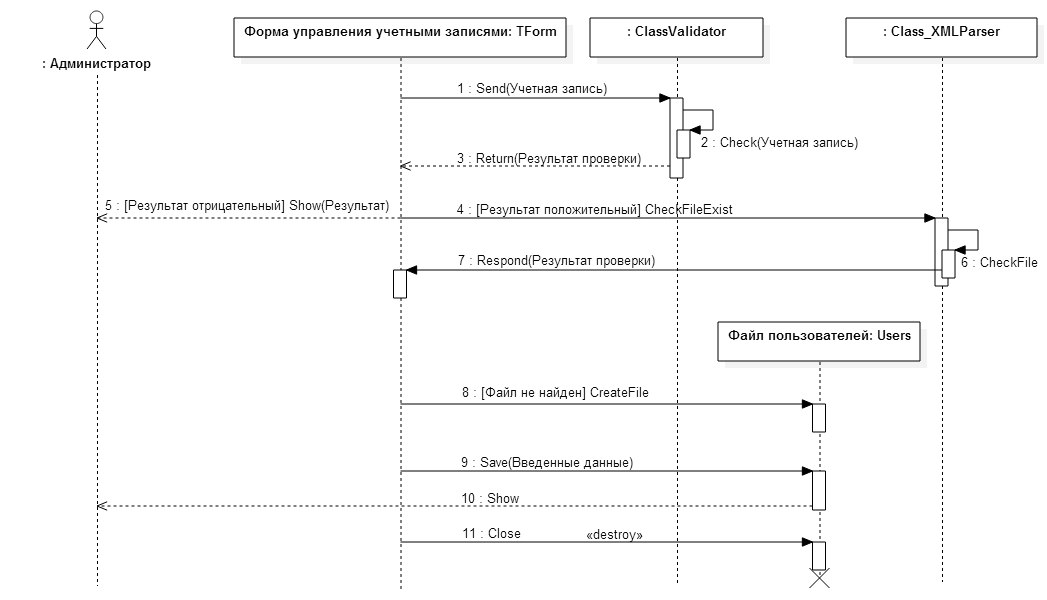
А2.1 Система выводит соответствующее сообщение об ошибке файла («Файл поврежден или недопустимого формата», «Файл с именем «USERS» не существует» или «Структура файла «USERS.xml» нарушена»). Администратор закрывает окно об ошибке. Система закрывает окно «Управление учетными записями» и завершает работу клавиатурного тренажера. Отображается состояние экрана до запуска приложения. Вариант использования завершается.

*Постусловия*. При успешном завершении на экране – окно «Управление учетными записями».

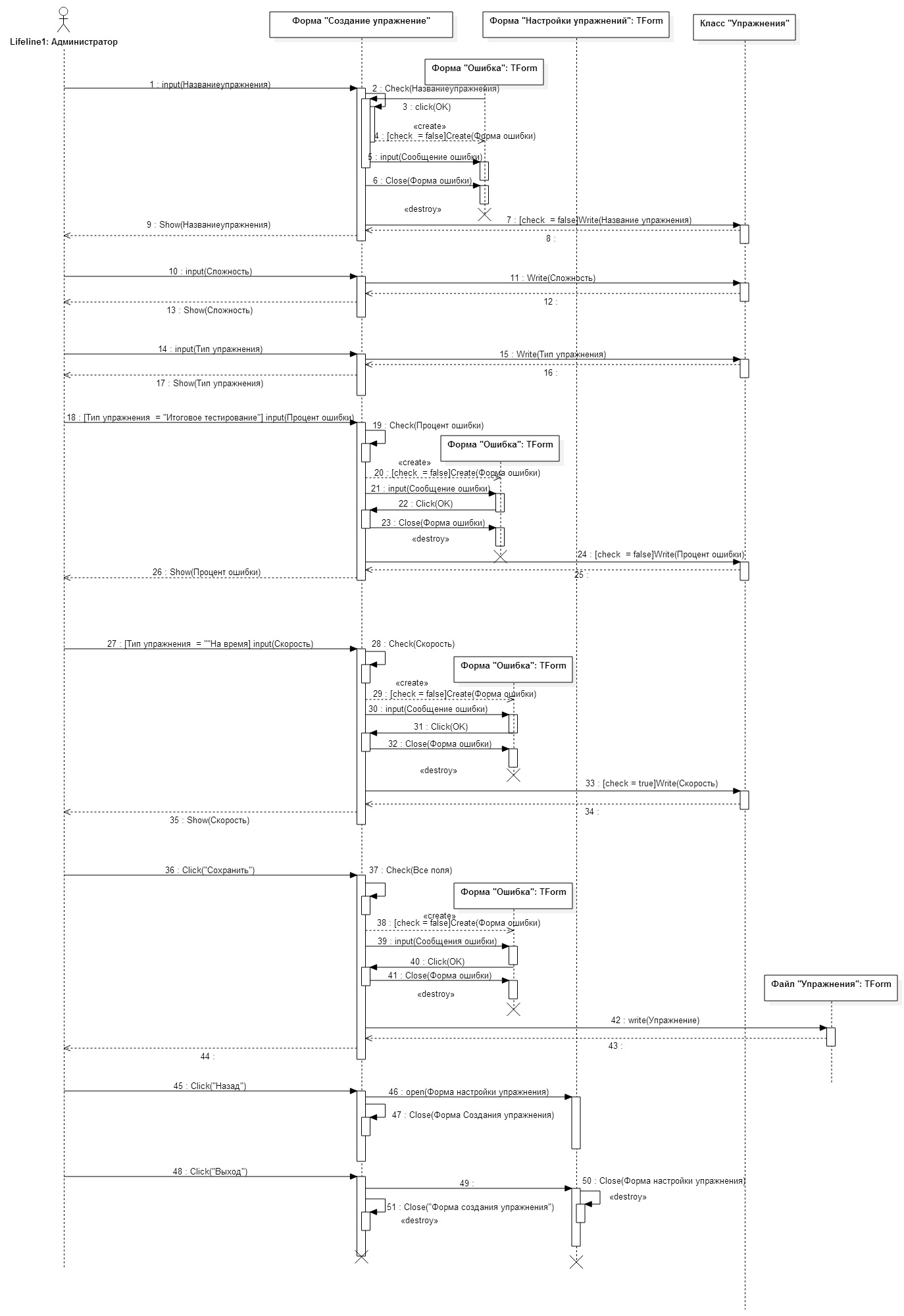
2.4.3 Диаграмма последовательности

Диаграмма последовательности – диаграмма, на которой показано взаимодействие объектов (обмен между ними сигналами и сообщениями), упорядоченное по времени, с отражением продолжительности обработки и последовательности их проявления [15].

На рисунке 2.25 представлена диаграмма последовательности для варианта использования «Сохранить изменения». Учетная запись пользователя, созданная администратором передается классу ClassValidator, который отвечает за проверку вводимых данных. Если результат проверки отрицательный, то система сообщает администратору о допущенных ошибках, иначе – учетная запись сохраняется в файл.

  
Рисунок 2.25 – Диаграмма последовательности для варианта использования «Сохранить изменения»

На рисунке 2.26 представлена диаграмма последовательности для варианта использования создания упражнений.

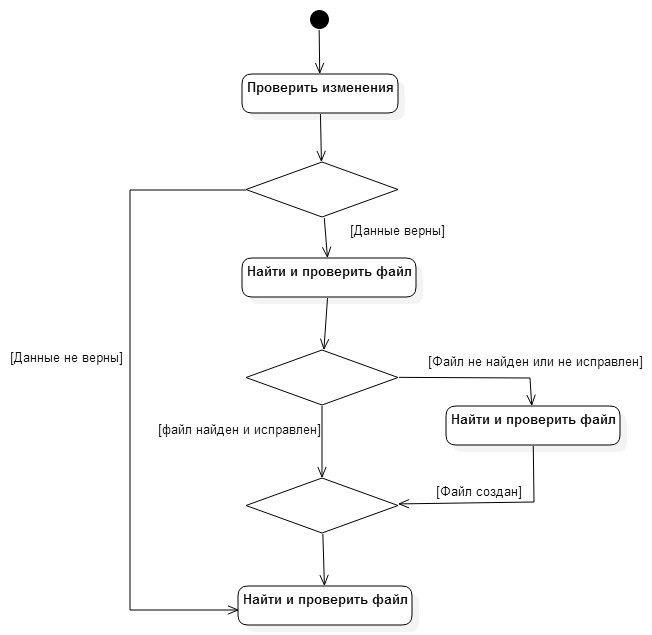
  
Рисунок 2.26 – Диаграмма последовательности для варианта использования «Создать упражнение»

2.4.4 Диаграмма деятельности

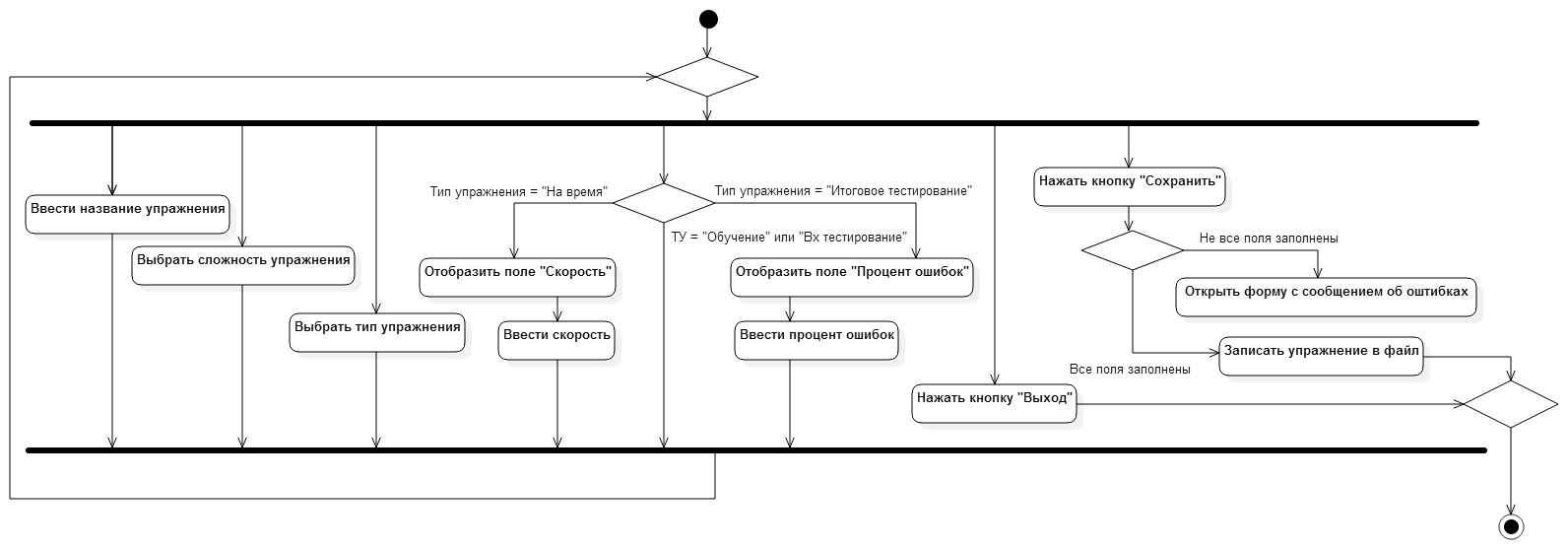
Диаграмма деятельности – UML-диаграмма, на которой показано разложение некоторой деятельности на её составные части.

Под деятельностью понимается спецификация исполняемого поведения в виде координированного последовательного и параллельного выполнения подчинённых элементов – вложенных видов деятельности и отдельных действий, соединённых между собой потоками, которые идут от выходов одного узла ко входам другого [15].

На рисунке 2.27 представлена диаграмма деятельности для варианта использования «Сохранить изменения».

  
Рисунок 2.27 – Диаграмма деятельности для варианта использования «Сохранить изменения»

На рисунке 2.28 представлена диаграмма деятельности для варианта использования «Создать упражнение».

  
Рисунок 2.28 – Диаграмма деятельности для варианта использования «Создать упражнение»

2.4.5 Диаграмма состояний

Диаграмма состояний (Statechart diagram) показывает автомат, фокусируя внимание на потоке управления от состояния к состоянию, используется для моделирования динамических аспектов системы.

Автомат (State machine) – это описание последовательности состояний, через которые проходит объект на протяжении своего жизненного цикла, реагируя на события, – в том числе описание реакций на эти события.

Состояние (State) – это ситуация в жизни объекта, на протяжении которой он удовлетворяет некоторому условию, осуществляет определенную деятельность или ожидает какого-то события.

Событие (Event) – это спецификация существенного факта, который происходит во времени и пространстве. В контексте автоматов событие – это стимул, способный вызвать срабатывание перехода.

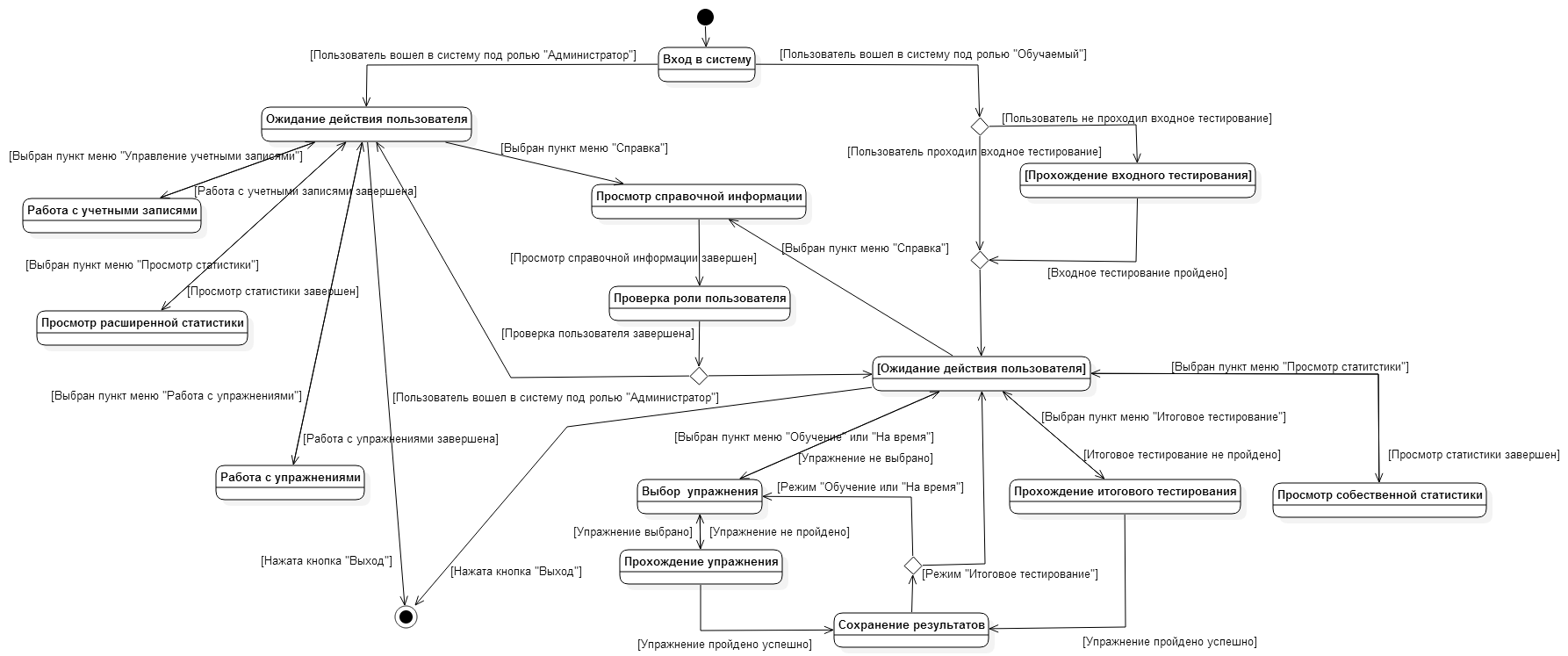
Переход (Transition) – это отношение между двумя состояниями, показывающее, что объект, находящийся в первом состоянии, должен выполнить некоторые действия и перейти во второе состояние, как только произойдет определенное событие и будут выполнены заданные условия [16].

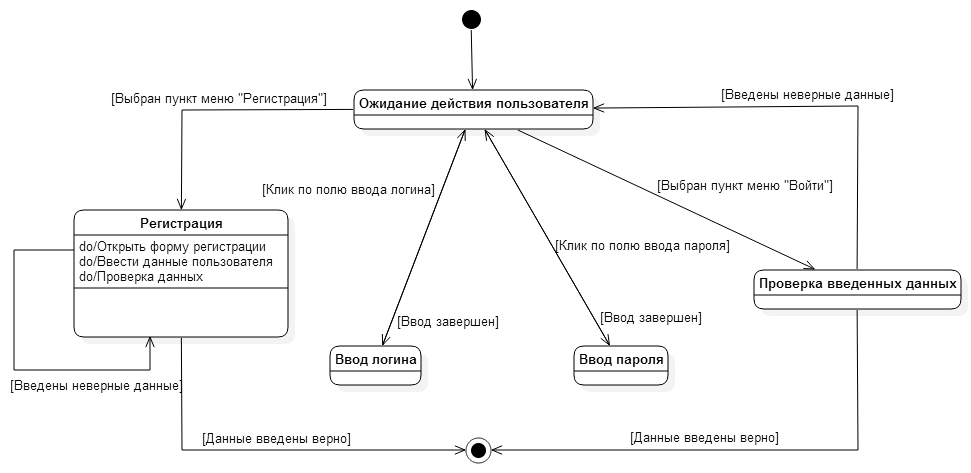
Деятельность (Activity) – это продолжающееся неатомарное вычисление внутри автомата.

Действие (Action) – это атомарное вычисление, которое приводит к смене состояния или возврату значения.

На рисунке 2.29 представлена диаграмма состояний системы, описывающая основные состояния системы после авторизации пользователя.

Рисунок 2.30 содержит диаграмму состояний системы при входе пользователя в систему.

  
Рисунок 2.29 – Диаграмма состояний системы

  
Рисунок 2.30 – Диаграмма состояний «Войти в систему»

2.4.6 Диаграмма классов

Диаграммой классов (class diagram) называют диаграмму, на которой показано множество классов, интерфейсов, коопераций и отношений между ними.

Диаграммы классов применяют для моделирования статического вида системы с точки зрения проектирования. В этом представлении удобнее всего описывать функциональные требования к системе – услуги, которые она предоставляет конечному пользователю [16].

На рисунке 2.31 представлена диаграмма классов системы, поля и методы которых описаны в таблицах 3-7.

  
Рисунок 2.31 – Диаграмма классов системы

Таблица 3– Описание класса «Пользователь»

|  |  |  |
| --- | --- | --- |
| Поля | | |
|  | Название | Тип |
| + | Идентификатор | Строка |
| + | Логин | Строка |
| + | Пароль | Строка |
| + | Фамилия | Строка |
| + | Имя | Строка |
| + | Отчество | Строка |
| + | Электронная почта | Строка |
| + | Является администратором | Логический |
| + | Уровень сложности | Целое |

Таблица 4– Описание класса «Упражнение»

|  |  |  |
| --- | --- | --- |
| Поля | | |
|  | Название | Тип |
| + | Идентификатор | Строка |
| + | Название | Строка |
| + | Сложность | Целое |
| + | Режим | Целое |
| + | Текст упражнения | Строка |
| + | Процент ошибок | Целое |
| + | Скорость | Целое |

Таблица 5– Описание класса «Валидатор»

|  |  |
| --- | --- |
| + | Методы |
| + | Проверить учетную запись пользователя |
| + | Проверить упражнение |
| + | Проверить существование пользователя |

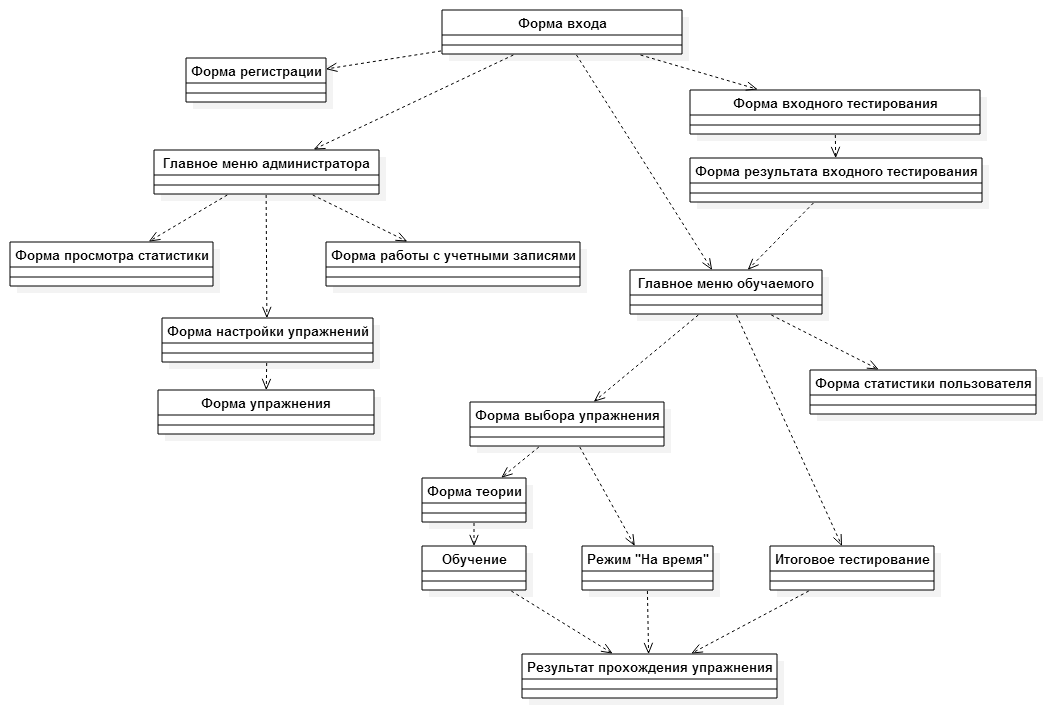
Таблица 6– Описание класса «Статистика»

|  |  |  |
| --- | --- | --- |
| Поля | | |
|  | Название | Тип |
| + | Идентификатор | Строка |
| + | Логин | Строка |
| + | Название упражнение | Строка |
| + | Дата прохождения | Целое |
| + | Сложность | Целое |
| + | Режим | Целое |
| + | Количество ошибок | Целое |
| + | Скорость | Целое |
| + | Время | Целое |
| + | Длина упражнения | Целое |

Таблица 7– Описание класса «Обработчик XML файлов»

|  |  |
| --- | --- |
|  | Методы |
| + | Проверить существование файла |
| + | Создать файл |
| + | Получить список пользователей |
| + | Получить список упражнений |
| + | Получить статистику всех пользователей |
| + | Получить статистику пользователя |
| + | Добавить пользователя |
| + | Добавить упражнение |
| + | Добавить статистику пользователя |
| + | Удалить пользователя |
| + | Удалить упражнение |
| + | Удалить статистику |

На рисунке 2.32 представлена диаграмма классов системы, описывающих пользовательский интерфейс.

  
Рисунок 2.32 – Диаграмма классов (пользовательские интерфейсы)

2.5 Выбор и обоснование комплекса программных средств

2.5.1 Выбор языка программирования и среды разработки

Для реализации автоматизированной системы «Клавиатурный тренажер» с функциями администратора был выбран язык C# по следующим причинам:

1. поддержка событийно-ориентированного программирования;
2. безопасный (по сравнению с языками C и C++) код;
3. подлинная объектная ориентированность;
4. унифицированная система типизации;
5. компонентно-ориентированное программирование.

Для написания программы на языке C# была выбрана среда разработки Visual Studio 2013 из-за следующих ее преимуществ:

1. простота пользовательского интерфейса;
2. расширенный редактор кода и простая отладка;
3. большой набор семантических функций (подсветка синтаксиса кода, автодополнение и т.д.);
4. единственная IDE, поддерживающая .NET.

2.5.2 Выбор операционной системы

Разрабатываемая система будет поддерживаться на операционных системах семейства Windows, которые обладают следующими преимуществами:

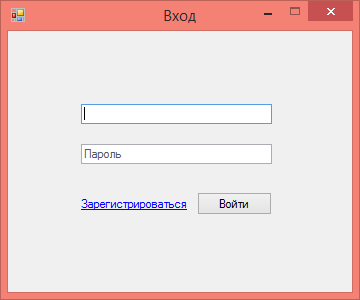
1. распространенность операционной системы;
2. единый пользовательский интерфейс;
3. большое количество поддерживаемых приложений и дополнений;
4. многозадачность;
5. стабильность работы.
6. Реализация системы
   1. Разработка и описание интерфейса пользователя

Интерфейс – это внешняя оболочка приложения вместе с программами управления доступом и другими скрытыми от пользователя механизмами управления, дающая возможность работать с документами, данными и другой информацией, хранящейся в компьютере или за его пределами. При разработке интерфейса необходимо учитывать следующее:

* интерфейс должен быть дружественным, интуитивно понятным;
* доступ ко всем необходимым функциям должен быть быстрым и удобным.
  + 1. Разработка и описание пользовательского меню

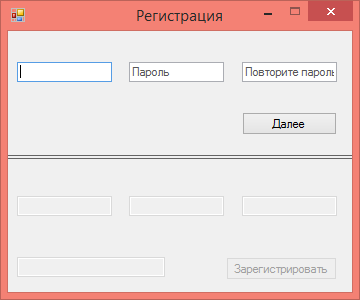
На основе прототипа интерфейса (см. п. 2.3) были разработаны экранные формы, представленные на рисунках 3.1 – 3.14.

После запуска программы открывается форма входа в систему, на которой зарегистрированному пользователю нужно ввести логин и пароль (рисунок 3.1) и нажать на кнопку «Войти».

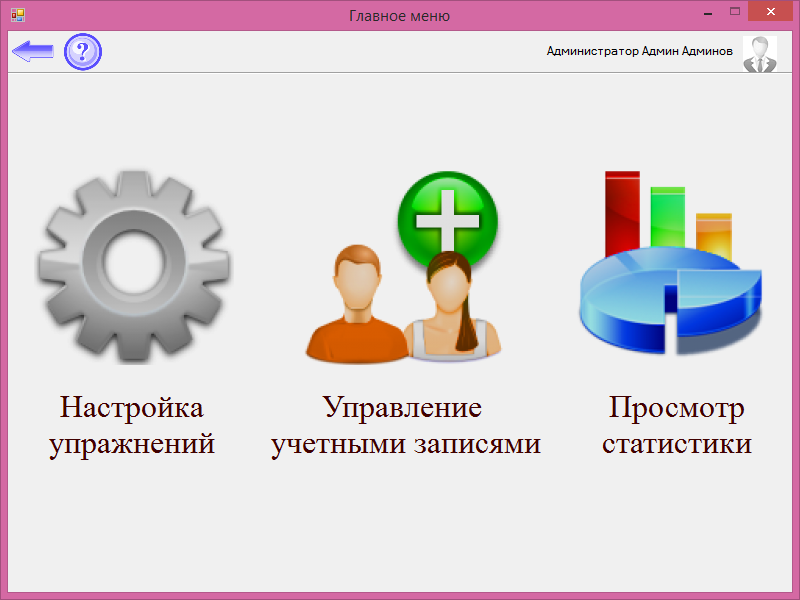
  
Рисунок 3.1 – Форма входа в систему

При нажатии на ссылку «Зарегистрироваться» откроется форма регистрации пользователей (рисунок 3.2).

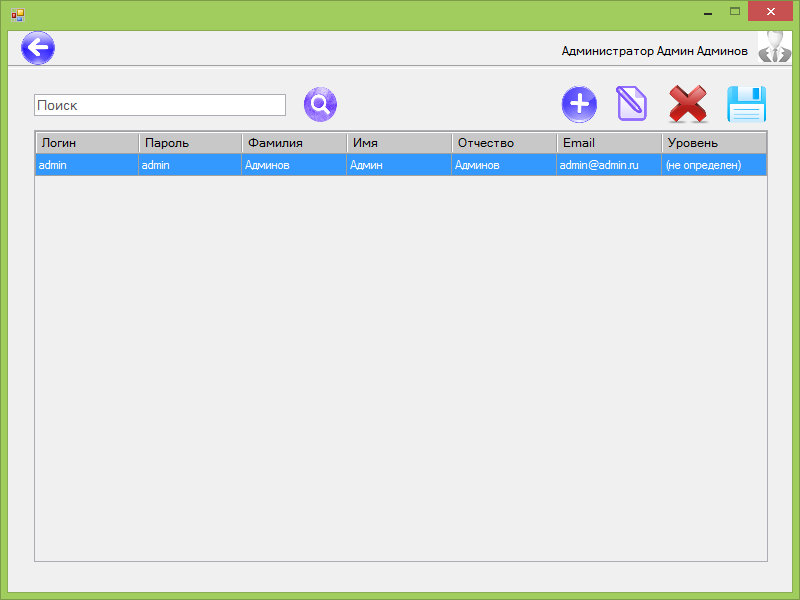
Для регистрации в системе необходимо ввести логин, пароль и подтвердить введенный пароль. После чего при нажатии кнопки «Далее» система проверит существование в системе пользователя с таким логином и, если результат проверки отрицательный, пользователь должен продолжит ввод данных (фамилия, имя, отчество и электронный адрес). При нажатии кнопки «Зарегистрировать» учетная запись будет сохранена, если данные введены верно, и откроется форма входа в систему.

  
Рисунок 3.2 – Форма регистрации

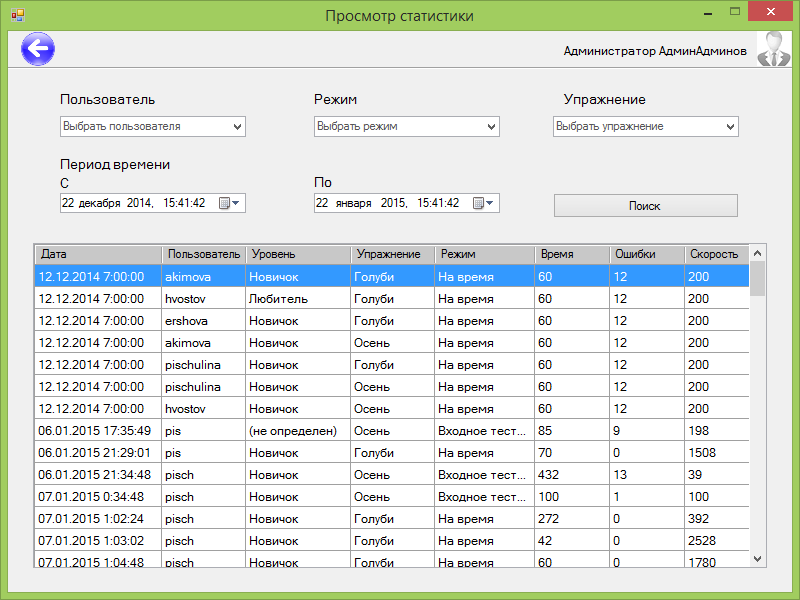
На рисунке 3.3 представлена главная форма администратора, на которой отображены все функции доступные администратору – просмотр справочной информации, управление учетными записями, просмотр статистики и настройка упражнений.

  
Рисунок 3.3 – Главная форма администратора

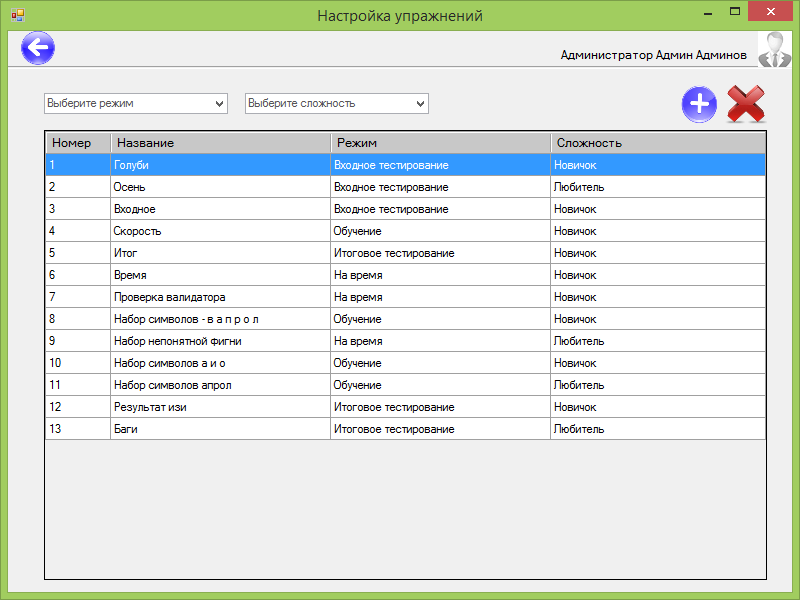
На рисунке 3.4 представлена форма управления учетными записями. Чтобы добавить, изменить или удалить учетную запись нужно нажать кнопки , , , соответственно. Для сохранения всех изменений нужно нажать на кнопку сохранения .

  
Рисунок 3.4 – Форма настройки учетных записей пользователей

На рисунке 3.5 представлена форма просмотра статистики пользователей. Для того чтобы отфильтровать таблицу со статистикой можно выбрать пользователя, режим, упражнение или период времени, в течении которого были получены результаты статистики, и нажать кнопку «Поиск».

  
Рисунок 3.5 – Форма просмотра статистики

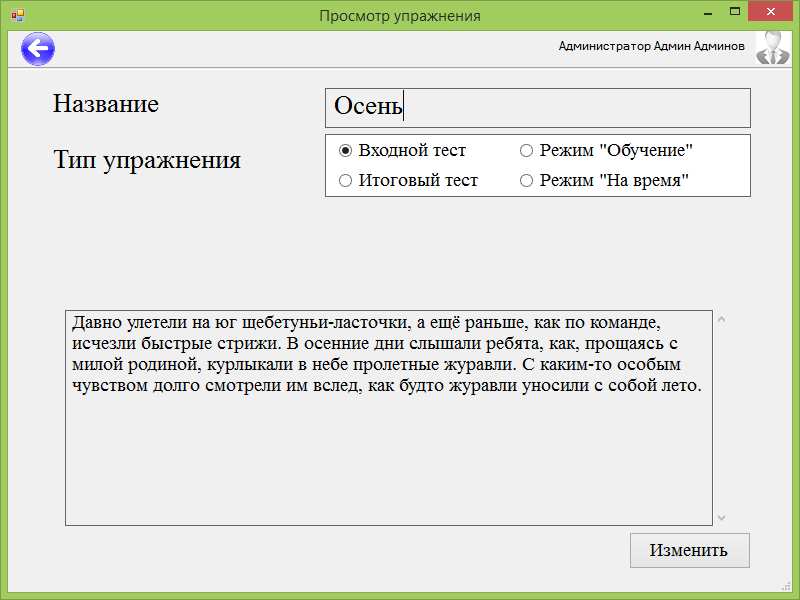
На рисунке 3.6 представлена форма настройки упражнений, на которой отображена таблица с упражнениями.

  
Рисунок 3.6 – Форма настройки упражнений

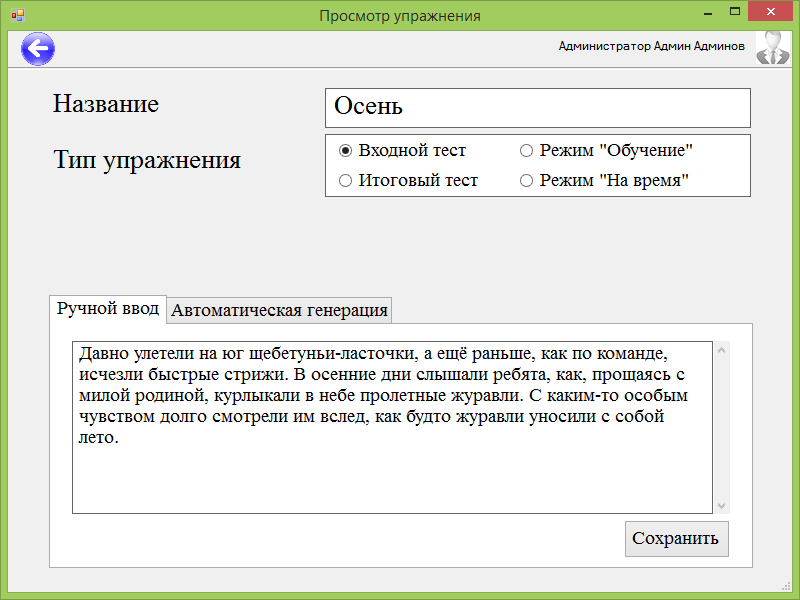
Для добавления и удаления упражнения необходимо нажать кнопки  и , расположенные над правым верхним углом таблицы.

Упражнения в таблице можно отфильтровать по режиму и сложности.

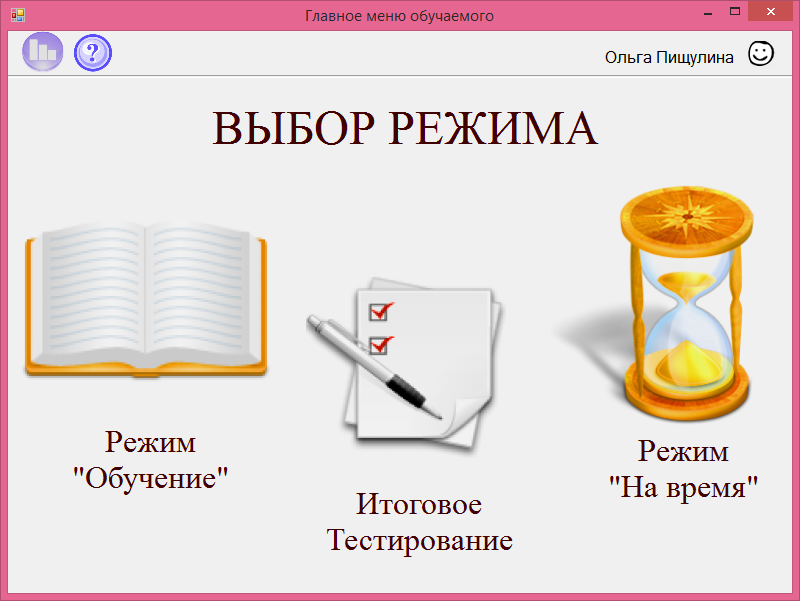
Для просмотра всех параметров упражнения нужно осуществить двойной щелчок левой кнопки мыши по выбранному упражнению. После этого откроется форма просмотра упражнения (рисунок 3.7).

  
Рисунок 3.7 – Форма просмотра упражнения

Чтобы отредактировать упражнение, нужно на форме просмотра нажать кнопку «Изменить», после чего форма будет доступна для редактирования (рисунок 3.8).

  
Рисунок 3.8 – Форма редактирования упражнения

Если пользователь вошел в систему с ролью обучаемого, то после входа будет открыта главное меню обучаемого (рисунок 3.9).

  
Рисунок 3.9 – Главное меню обучаемого

При нажатии на кнопку  будет открыта форма просмотра статистики (рисунок 3.10).

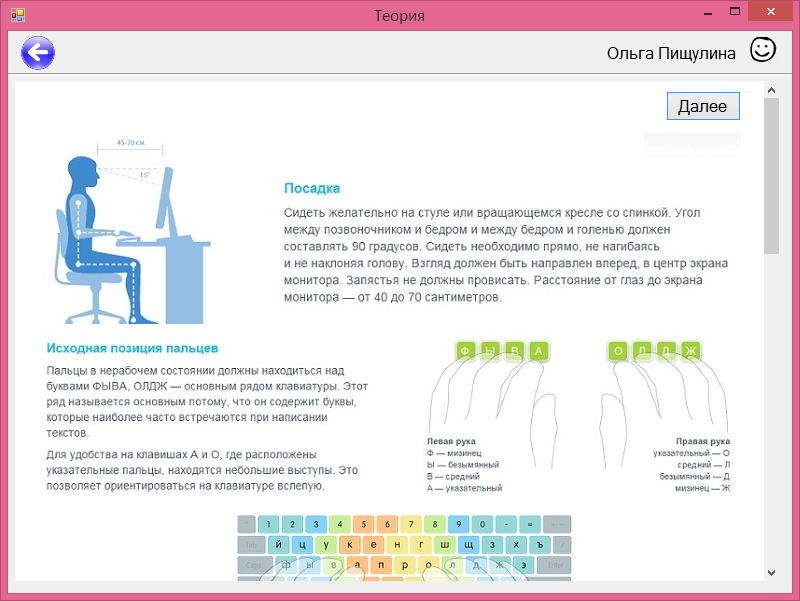
Для просмотра справочной информации необходимо нажать на кнопку .

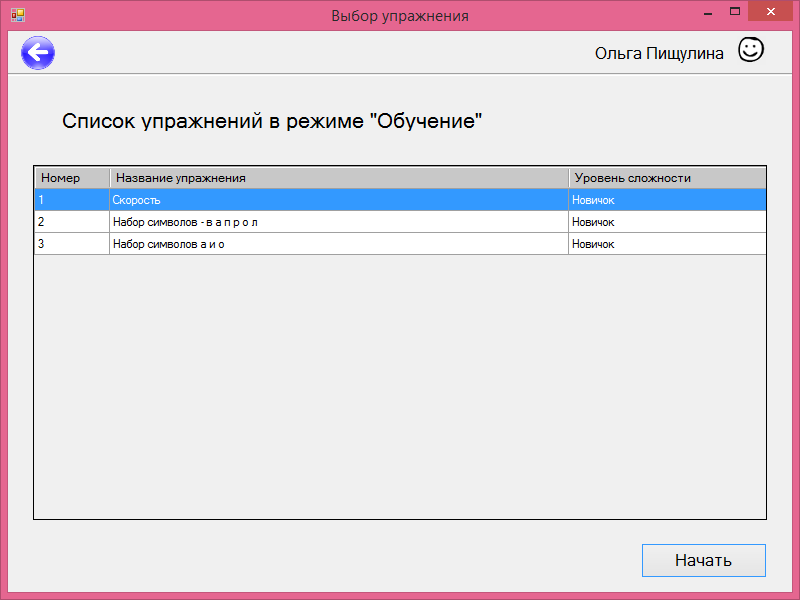
  
Рисунок 3.10 – Форма просмотра статистики

В главном меню обучаемый может выбрать режим упражнения. При выборе итогового тестирования будет открыта форма прохождения упражнения (рисунок 3.11).

  
Рисунок 3.11 – Форма прохождения упражнений

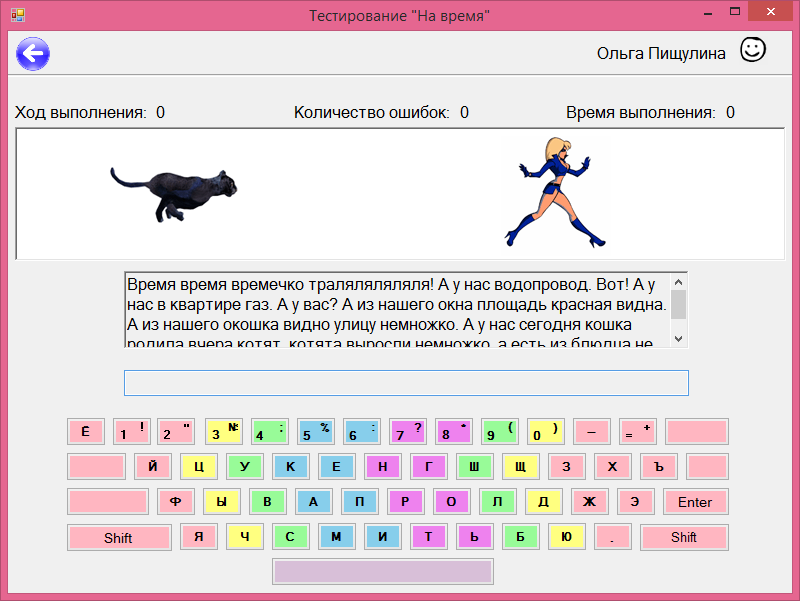
При выборе режима «Обучение» будет открыта форма теории (рисунок 3.12), нажав кнопку далее на которой можно перейти в форму выбора упражнения (рисунок 3.13).

  
Рисунок 3.12 – Форма теории

  
Рисунок 3.13 – Форма выбора упражнения

Начать прохождение упражнения можно при нажатии соответствующей кнопки или при двойном щелчке по выбранному упражнению.

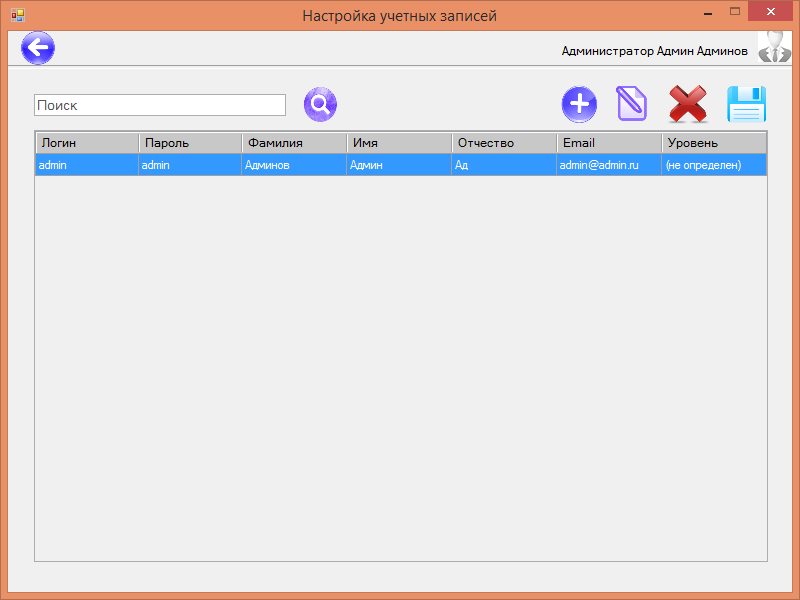
На рисунке 3.14 представлена форма прохождения упражнений в режиме «На время». Если скорость набора символов пользователя меньше заданной минимальной скорости для этого упражнения, то пантера достигнет девушку и упражнение будет прервано.

  
Рисунок 3.14 – Форма прохождения упражнений в режиме «На время»

* + 1. Описание контрольного примера в режиме администратора

Рассмотрим в качестве контрольного примера создание учетной записи пользователя в режиме администратора.

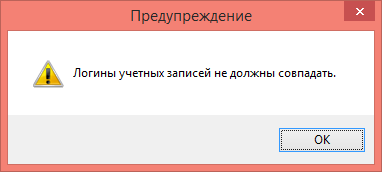
Для работы с учетными записями необходимо выбрать пункт «Управление учетными записями» в главном меню администратора, после чего отображается форма, изображенная на рисунке 3.15.

  
Рисунок 3.15 – Форма управления учетными записями

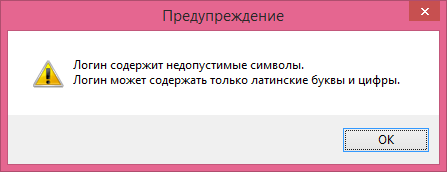
Для того чтобы добавить учетную запись нужно нажать на кнопку добавления . После этого в таблице добавится пустая строка, предназначенная для новой учетной записи.

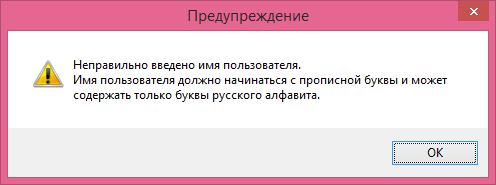
Для сохранения учетной записи нужно нажать кнопку сохранения .

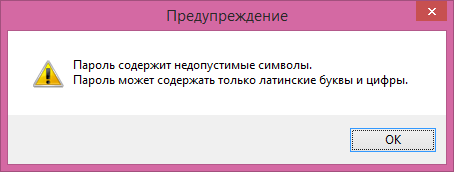
Если при создании новой учетной записи ввести такой логин, с которым уже зарегистрирован другой пользователь, то при нажатии кнопки сохранения появится предупреждение (рисунок 3.17) и учетная запись не будет сохранена.

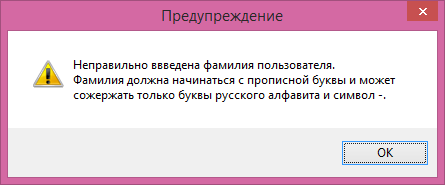
  
Рисунок 3.16 – Сообщение о совпадении логина

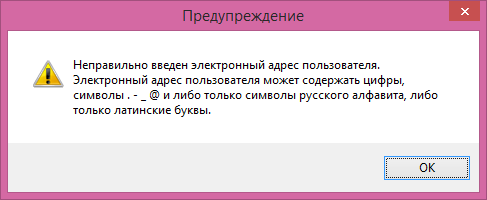
На рисунках 3.17-3.22 представлены возможные ошибки в учетных записях.

  
Рисунок 3.18 – Сообщение об ошибке логина

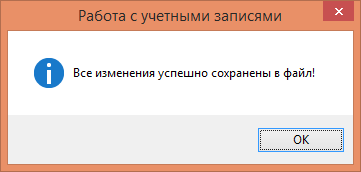
  
Рисунок 3.19 – Сообщение об ошибке имени пользователя

  
Рисунок 3.20 – Сообщение об ошибке пароля

  
Рисунок 3.21 – Сообщение об ошибке фамилии

  
Рисунок 3.22 – Сообщение об ошибке электронного адреса

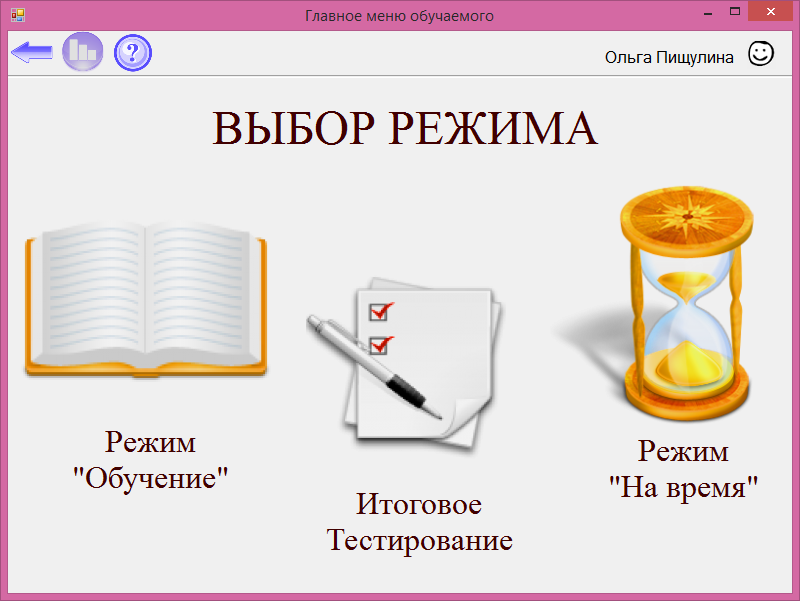
В случае ввода учетной записи без ошибок, будет выведено сообщение об успешном сохранении (рисунок 3.23).

  
Рисунок 3.23 – Сообщение об успешном сохранении

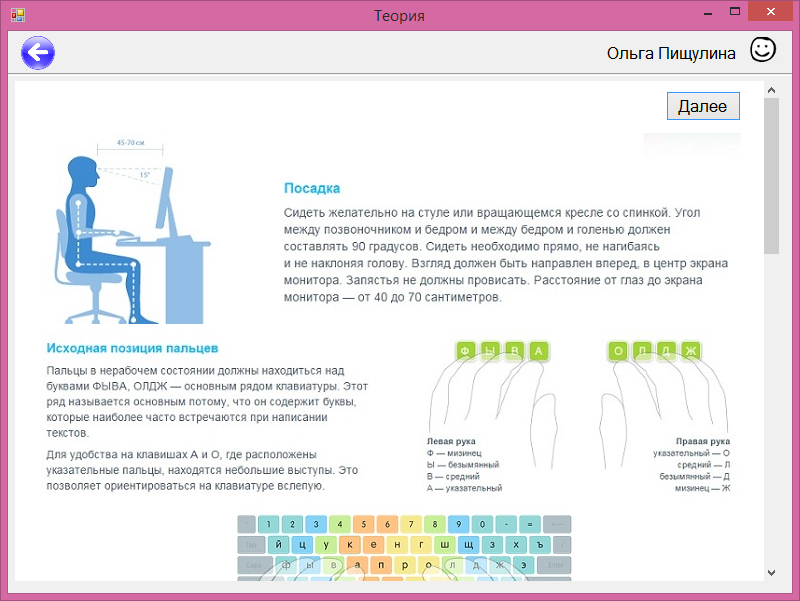
* + 1. Описание контрольного примера в режиме обучаемого

Рассмотрим в качестве контрольного примера прохождение пользователем упражнения в режиме обучения.

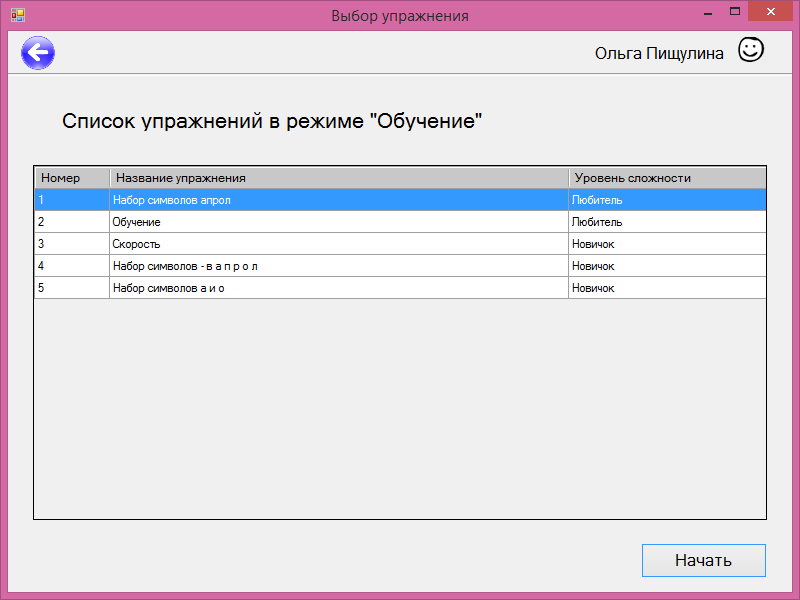
Для того чтобы пройти упражнение в режиме «Обучение», пользователь должен выбрать пункт меню «Режим «Обучение»» в главном меню обучаемого (рисунок 3.24).

  
Рисунок 3.24 – Главное меню обучаемого

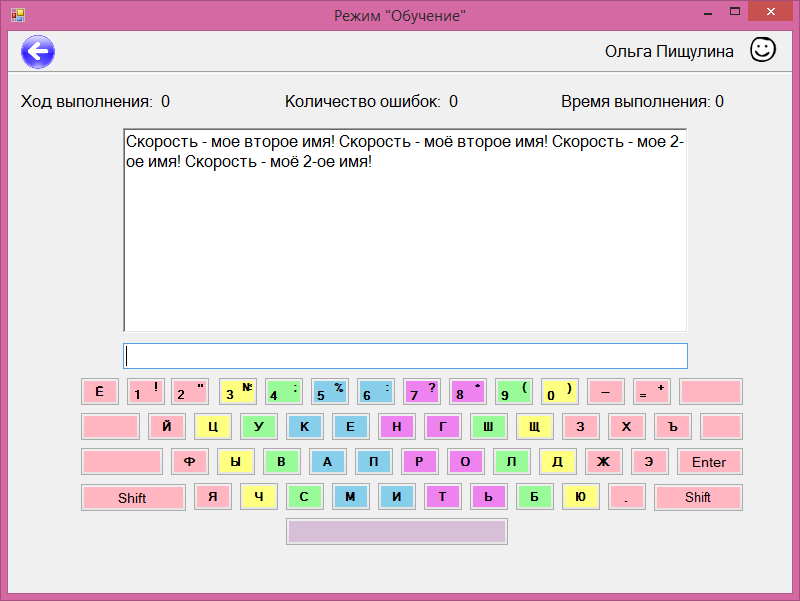
После выбора обучения будет открыта форма теории по основам метода слепой печати (рисунок 3.25).

  
Рисунок 3.25 – Форма теории

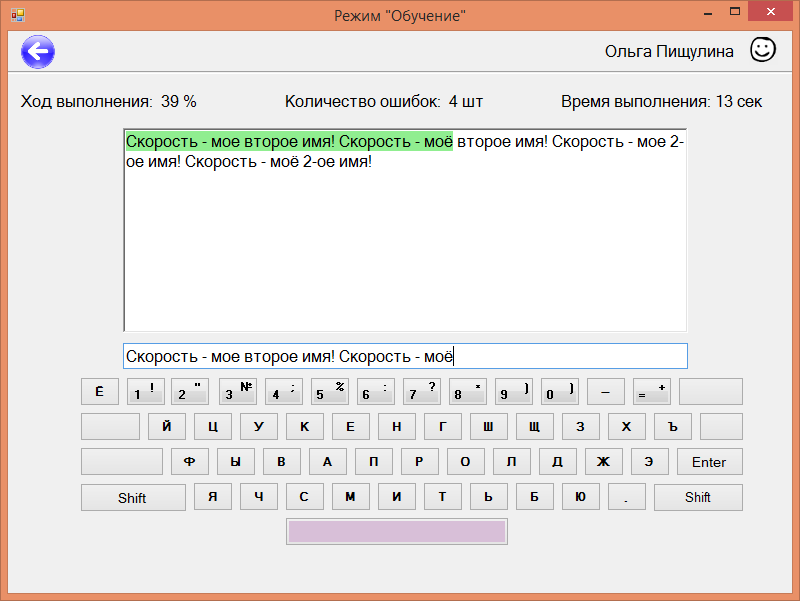
При нажатии на кнопке далее откроется форма выбора упражнения (рисунок 3.26).

  
Рисунок 3.26 – Форма выбора упражнения в режиме «Обучение»

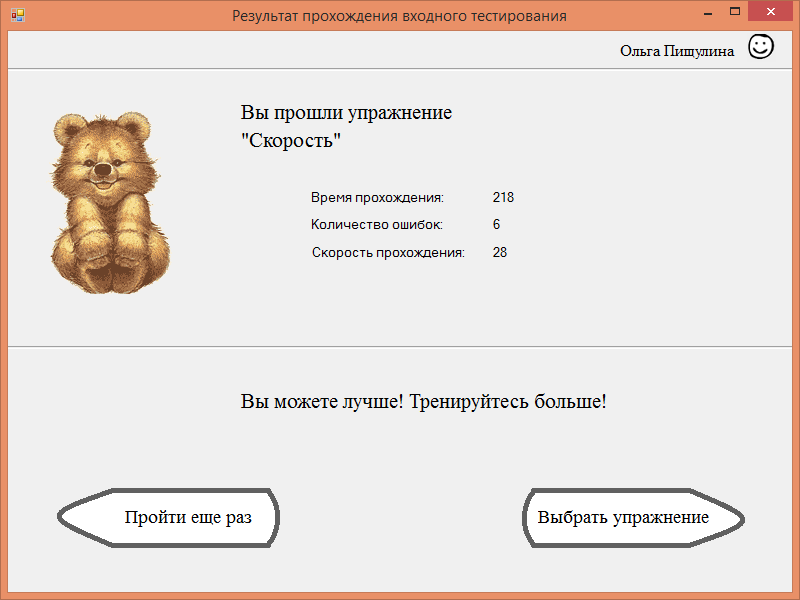
Чтобы начать упражнение нужно нажать кнопку «Начать» или два раза щелкнуть левой кнопкой мыши по выбранному упражнению. После чего откроется форма прохождения упражнения (рисунок 3.27).

  
Рисунок 3.27 – Форма прохождения упражнения в режиме «Обучение»

При прохождении упражнения введенный текст подсвечивается зеленым цветом. В случае ошибки при наборе символ в упражнении будет выделен красным цветом и количество ошибок увеличится (рисунок 3.28).

  
Рисунок 3.28 – Форма прохождения упражнения в процессе набора символов

После того как последний символ упражнения будет введен правильно откроется форма результата (рисунок 3.29). Упражнение можно пройти повторно, либо вернуться к форме выбора упражнений.

  
Рисунок 3.29 – Форма результата прохождения упражнения

* 1. Диаграммы реализации

Диаграмма компонентов и диаграмма развертывания применяются при моделировании физических аспектов объектно-ориентированной системы

* + 1. Диаграмма компонентов

Диаграммы компонентов применяются для моделирования статического вида системы с точки зрения реализации. Сюда относится моделирование физических сущностей, развернутых в узле, например, исполняемых программ, библиотек, таблиц, файлов и документов.

Диаграмма компонентов показывает набор компонентов и отношения между ними, обычно включает в себя компоненты, интерфейсы, отношения зависимости, обобщения, ассоциации и реализации [16].

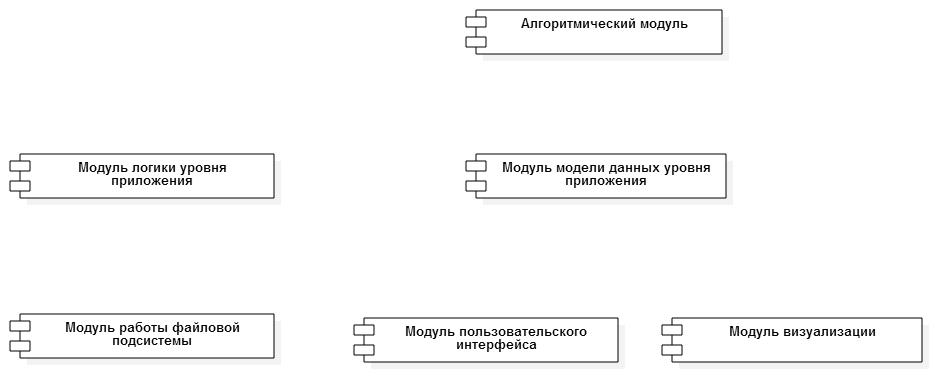
Диаграмма компонентов изображена на рисунке 3.30. Описание компонент диаграммы приведено в таблице 8.

Таблица 8 – Описание диаграммы компонент

|  |  |
| --- | --- |
| Название компонента | Назначение компоненты |
| Алгоритмический модуль | Объединение реализаций всей алгоритмической базы, используемой в проекте. |
| Модуль логики уровня приложения | Формирует реакцию системы на пользовательский ввод. |

Продолжение таблицы 8

|  |  |
| --- | --- |
| Модуль модели данных уровня приложения | Обеспечивает обмен данными между частями подсистемы логики уровня приложения. |
| Модуль работы файловой подсистемы | Обеспечивает возможность проведения операций, связанных с файловой подсистемой. |
| Модуль визуализации | Формирует графическое представление результатов работы системы. |
| Модуль пользовательского интерфейса | Формирует графическую оболочку системы для её дальнейшего взаимодействия с пользователем. |

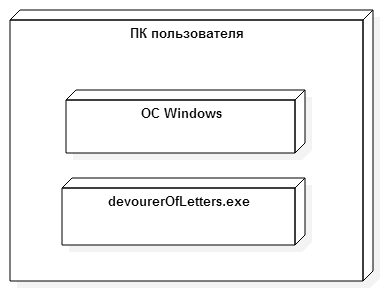
  
Рисунок 3.30 – Диаграмма компонентов системы

* + 1. Диаграмма развертывания

На диаграмме развертывания, или применения, показана конфигурация обрабатывающих узлов, на которых выполняется система, и компонентов, размещенных в этих узлах.

Диаграммы развертывания используются для моделирования статического вида системы с точки зрения развертывания. В основном под этим понимается моделирование топологии аппаратных средств, на которых выполняется система [16].

Диаграмма развертывания системы представлена на рисунке 3.31. Приложение взаимодействует с базой данных, которая представлена в виде файловой подсистемы. База данных располагается на той же ЭВМ, что и приложение.

  
Рисунок 3.31 – Диаграмма развертывания системы

* 1. Выбор и обоснование комплекса технических средств
     1. Расчет объема занимаемой памяти

Расчет объема занимаемой памяти включает в себя расчет необходимого объема оперативной памяти, а также памяти жесткого диска.

Необходимый объем внешней памяти рассчитывается по следующей формуле [13]:

VЖД = VОС + VПР +,

где VОС – размер системных файлов операционной системы;

VПР – объем памяти, занимаемый файлами приложения;

Vф .

Операционная система Windows 7 требует VОС 9,43 Гб памяти. Файлы приложения занимают VПР ~5 Мб.

Исходя из этих данных, рассчитаем требуемый объем памяти жесткого диска:

VЖД = 9,43 Гб + 5 Мб = 9,435 Гб.

Для расчета необходимого объема ОЗУ используется следующая формула [13]:

VОЗУ = VОС + VПР.

Операционная система занимает 1024 Мб, приложение – 13 Мб. Исходя из этих значений, рассчитаем требуемый объем оперативной памяти:

VОЗУ = 1024 Мб + 13 Мб + 0 Мб = 1037 Мб.

* + 1. Минимальные требования, предъявляемые к системе

Для функционирования системы необходимо:

1. IBM PC – совместимый компьютер;
2. не менее 1,1 Гб оперативной памяти;
3. 9,435 Гб на жестком диске;
4. дисплей с разрешением не менее 800 x 600 пикселей;
5. клавиатура или иное устройство ввода;
6. мышь или иное манипулирующее устройство.

Заключение

В соответствии с техническим заданием был произведен анализ предметной области, реализовано программное и информационное обеспечение, подготовлены контрольные примеры и произведены тестирование и отладка автоматизированной системы «Клавиатурный тренажер».

При выполнении работы использовалась методология объектно-ориентированного анализа и проектирования, кроме того применялась технология быстрой разработки приложений RAD.

В первом разделе были проведены описание и анализ предметной области, обзор существующих систем аналогов, используемых для обучения методу слепой печати. На основании этого была сформулирована задача определения функций, которые должны быть выполнены в системе.

Во втором разделе описаны этапы проектирования системы: была разработана структурная схема системы, описана спецификация требований, проведено прототипирование интерфейса пользователя системы и построены соответствующие проекту диаграммы: вариантов использования, классов и другие. Также произведен выбор и обоснование комплекса программных средств.

В третьей части описана реализация системы: разработка и описание интерфейса пользователя, описание контрольных примеров.

Разработанная в ходе курсового проектирования система отвечает всем поставленным требованиям: имеет удобный интерфейс для пользователя, легок и надежен в использовании, не требует серьезных аппаратных затрат.

Список использованной литературы

1. RAD (программирование) [Электронный ресурс] – https://ru.wikipedia.org/wiki/RAD (дата обращения 25.09.2014 г.).
2. UML [Электронный ресурс] – <https://ru.wikipedia.org/wiki/UML>.
3. Соло на клавиатуре [Электронный ресурс] – https://ergosolo.ru/ (дата обращения 25.09.2014 г.).
4. Список клавиатурных тренажеров [Электронный ресурс] – https://ru.wikipedia.org/wiki/Список\_клавиатурных\_тренажеров (дата обращения 27.09.2014 г.).
5. Слепой метод печати [Электронный ресурс] – https://ru.wikipedia.org/wiki/Слепой\_метод\_печати (дата обращения 27.09.2014).
6. Зрячий метод печати [Электронный ресурс] – https://klavogonki.ru/wiki/Зрячий\_метод\_печати (дата обращения 25.09.2014 г.).
7. Фыва-олдж [Электронный ресурс] – https://klavogonki.ru/wiki/Фыва\_олдж (дата обращения 25.06.2014 г.).
8. Ывам-толд [Электронный ресурс] – https://klavogonki.ru/wiki/Ывам-толд (дата обращения 25.09.2014 г.).
9. Клавиатурный тренажер Stamina [Электронный ресурс] – http://stamina-online.ru/ (дата обращения 27.09.2014 г.).
10. Клавиатурный тренажер «Все 10» [Электронный ресурс] – http://vse10.ru/ (дата обращения 27.09.2014 г.).
11. Клавиатурный тренажер Verseq [Электронный ресурс] – http://www.verseq.ru/ (дата обращения 27.09.2014 г.).
12. Клавиатурный тренажер RapidTyping [Электронный ресурс] – http://www.rapidtyping.com/ (дата обращения 27.09.2014 г.).
13. Зеленко, Л.С. Методические указания к лабораторному практикуму по дисциплине «Программная инженерия» [Текст]/ Л.С. Зеленко. – Самара: СГАУ, 2012. – 67 с.
14. Диаграммы вариантов использования [Электронный ресурс] – http://www.informicus.ru/ (дата обращения 20.11.2014 г.).
15. Диаграмма деятельности [Электронный ресурс] – https://ru.wikipedia.org/wiki/Диаграмма\_деятельности (дата обращения 20.11.2014 г.).
16. Язык UML Руководство пользователя [Электронный ресурс] – http://dit/isuct.ru (дата обращения 20.12.2014 г.).

Приложение А  
Руководство пользователя

Приложение Б  
Листинг программы