**Министерство науки и высшего образования Российской Федерации**

**Федеральное государственное автономное образовательное учреждение**

**высшего образования**

**«КАЗАНСКИЙ (ПРИВОЛЖСКИЙ) ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

ИНСТИТУТ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И

ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

КАФЕДРА ТЕХНОЛОГИЙ ПРОГРАММИРОВАНИЯ

Направление: 09.03.03 – «Прикладная информатика»

КУРСОВАЯ РАБОТА

ПО ДИСЦИПЛИНЕ «ПРОЕКТИРОВАНИЕ ЧЕЛОВЕКО-МАШИННОГО ИНТЕРФЕЙСА»

ПО ТЕМЕ

**«ДЕКАНАТ»**

Студент 2 курса

Группа 09-952

“\_\_\_” \_\_\_\_\_\_\_\_\_\_\_\_\_ 2020 г.    Р.Р. Ахметзянова

Научный руководитель

к.ф.-м.н.,

доцент КТП ИВМиИТ КФУ

“\_\_\_” \_\_\_\_\_\_\_\_\_\_\_\_\_ 2020 г.     И.С. Балафендиева

Казань-2020

Содержание

Введение3

1. Анализ предметной области4

1.1. Назначение системы4

1.2. Основной функционал приложения 4

1.3. Глоссарий5

1.4. Действующие лица6

1.5. Основные функции системы6

1.5.1. Вход в систему7

1.5.2 Просмотреть учет успеваемости8

1.5.3 Отчислить студента9

1.5.4. Принять студента9

1.5.5. Перевести студента с группы в группу и с курса на курс10

2. Описание системы11

2.1. Выбор программных средств реализации11

2.2. Хранение информации11

2.3. Реализация11

3. Описание пользовательского интерфейса14

3.1 Выбор цветовой палитры14

3.2. Выбор шрифта15

3.3. Основные экранные элементы интерфейса16

4. Руководство по использованию системы17

Заключение19

Список использованных источников20

Введение

В наше время, когда информационные системы получили наибольшую популярность, практически невозможно найти предприятие, которое не пользуется этими технологиями. Любое предприятие должно обрабатывать огромные объемы данных в кратчайшие сроки, и чем больше охват аудитории, тем больше информации поступает. Очевидно, что намного выгоднее потратить ресурсы на создание своей информационной системы.

Целью данной курсовой работы является создание системы, требуемой для корректной работы сотрудников деканата.

Процесс создания данного приложения должен сформировать у автора навык практической работы, связанной с проектированием и разработкой человеко-машинного интерфейса, также необходимо выделить основные этапы создания графического интерфейса.

В ходе работы необходимо разделить процесс выполнения на несколько этапов:

1. Проанализировать предметную область;
2. Выявить функциональные требования;
3. Спроектировать интерфейс системы;
4. Реализовать саму систему;

1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

1.1 НАЗНАЧЕНИЕ СИСТЕМЫ.

Деканат работает с кафедрами, преподавателями, старостами групп и студентами. Здесь происходит прием абитуриентов, отчисление студентов, учет успеваемости по итогам сессий, перевод студентов между группами и курсами.

Разрабатываемая модель информационной системы помогает реализовать работу деканата, где осуществляется управление работой факультета. Также она предназначена для ускорения и повышения эффективности функционирования организации.

Пользователям должны быть доступны такие операции, как добавление студентов, отчисление студентов, перевод студентов с группы на группу и с курса на курс.

Исходя из этого, делается вывод, что необходимо хранить информацию о пользователях, студентах и их успеваемости. Критерии к пользователям: ФИО (String), логин (String), пароль (String), должность (String). Критерии к студентам: ФИО (String), курс (int), группа (int), основа обучения (String). Критерии к оценкам: оценка (int), дисциплина (int).

1.2. ОСНОВНОЙ ФУКЦИОНАЛ ПРИЛОЖЕНИЯ

Для того чтобы сотруднику совершить какое-либо действие, ему необходимо зарегистрироваться в системе и войти.

В приложении должно быть реализовано главное меню, которое позволит пользователю легко ориентироваться по системе.

На основе этой информации выделим действующих лиц, варианты использования и однозначно интерпретируем понятия с помощью глоссария.

Система позволит сотрудникам деканата контролировать прием и отчисление студентов. В деканате методист, исходя из полученных данных от приемной комиссии, формирует академические группы и вносит информацию о студентах в систему.

Деканатом контролируется успеваемость студентов. Cистема предоставит сотрудникам полученные студентами оценки, основываясь на полученных сведениях.

1.3 ГЛОССАРИЙ

|  |  |
| --- | --- |
| **Термин** | **Значение** |
| Деканат | Организационный центр по управлению работой факультета. |
| Студент | Личность, проходящая обучение в университете. |
| Преподаватель | Личность, проводящая занятия у студентов. |
| Секретарь | Личность, занимающаяся делопроизводством и реализацией поручений декана. Помогает в управлении факультетом, решении основных задач, готовит документы, доносит нужные сведения до кафедры и пр. |
| Методист | Личность, отвечающая за формирование групп, оформление и выдачу документов и справок, работают со старостами и т.д. |
| Приемная комиссия | Подразделение университета, осуществляющее приемную кампанию на будущий учебный год. |
| Список курса | Список всех студентов, обучающихся на определенной годичной ступени обучения в университете. |
| Список группы | Список всех студентов, принадлежащих малой учебной группе университета, обусловленной спецификой учебной деятельности. |
| Отчисление | Прекращение обучения студента в вузе. |
| Прием | Начало обучения студента в вузе. |
| Сессия | Период сдачи экзаменов. |
| Табель успеваемости | Все оценки за все предметы, полученные студентом в данном семестре. |

1.4 ДЕЙСТВУЮЩИЕ ЛИЦА

Сотрудники деканата – принимают и отчисляют студентов, ведут учет успеваемости по итогам сессии, переводят студентов из группы в группу и с курса на курс.

Система – хранит информацию, помогает работе сотрудников.

1.5 ОСНОВНЫЕ ФУНКЦИИ СИСТЕМЫ

Исходя из потребностей действующих лиц, выделяются следующие варианты использования:

* Войти в систему.
* Просмотреть учет успеваемости по итогам сессии.
* Отчислить студента.
* Принять студента.
* Перевести студента с группы в группу и с курса на курс.

1.5.1 ВХОД В СИСТЕМУ

Краткое описание:

Данный вариант использования описывает вход пользователя в систему деканата.

Основной поток событий:

Данный вариант использования начинает выполняться, когда пользователь хочет войти в систему.

1. Система запрашивает требуемое действие (войти, зарегистрироваться).
2. Когда пользователь указывает действие, начинает выполняться один из подчиненных потоков (войти, зарегистрироваться).

Войти:

1. Система запрашивает имя пользователя и пароль.

2. Пользователь вводит имя и пароль.

3. Система подтверждает имя и пароль, после чего открывается доступ в систему.

Зарегистрироваться:

1. Система просит ввести данные.

2. Пользователь вводит данные.

3. Система регистрирует пользователя, после чего открывается доступ.

Альтернативный поток:

1. Неправильное имя/пароль (при выполнении потока «Войти»):

Если во время выполнения основного потока обнаружится, что пользователь ввел неправильное имя и/или пароль, система выводит сообщение об ошибке. Пользователь может вернуться к началу основного потока событий, чтобы ввести корректные данные или же отказаться от входа в систему, при этом выполнение варианта использования завершается.

2. Недопустимые символы (при регистрации):

Если во время регистрации пользователь введет недопустимые символы (пробел), система выводит сообщение об ошибке. Пользователь может вернуться к регистрации, при этом выполнение варианта использования завершается.

3. Данные уже существуют (при регистрации):

Если во время регистрации пользователь введет логин, который уже существует в системе, выводится сообщение об ошибке. Пользователь может вернуться к регистрации, при этом выполнение варианта использования завершается.

1.5.2 ПРОСМОТРЕТЬ УЧЕТ УСПЕВАЕМОСТИ ПО ИТОГАМ СЕССИИ

Краткое описание:

Данный вариант использования позволяет сотруднику проанализировать успеваемость студентов исходя из результатов каждой сессии. Система каталога групп предоставляет информацию об успеваемости всех групп на каждом курсе.

Основной поток событий:

Данный вариант начинает выполняться, когда деканату необходимо проконтролировать успеваемость.

1. Система запрашивает сделать выбор (выбрать курс, выбрать факультет, выбрать группу, динамика успеваемости).

2. Когда сотрудник делает выбор, открывается диаграмма с данными.

3. Данные выводятся в зависимости от сделанного выбора. В таблице указывается список и кол-во студентов, сдававших экзамены, дисциплины, оценки.

Предусловия:

Перед началом выполнения данного варианта использования студент должен войти в систему.

1.5.3 ОТЧИСЛИТЬ СТУДЕНТА

Краткое описание:

Данный вариант использования позволяет сотрудникам отчислять студентов.

Основной поток событий:

Данный вариант использования начинает выполняться, когда сотрудник запрашивает документацию о нужном студенте/абитуриенте.

Отчисление студентов:

1. Пользователь открывает список студентов, ищет студента, которого необходимо отчислить.

2. Пользователь открывает личную анкету студента и нажимает кнопку «Отчислить».

3. Система удаляет отчислившегося студента.

Предусловия:

Перед началом выполнения данного варианта использования пользователь должен войти в систему.

1.5.4 ПРИНЯТЬ СТУДЕНТА

Краткое описание:

Данный вариант использования позволяет сотрудникам добавить нового студента в систему.

Основной поток событий:

Данный вариант использования начинает выполняться, когда сотрудник переходит во вкладку «Добавить студента».

1. Пользователь вводит данные о новом студенте.

2. Система сохраняет студента и обновляет списки.

Предусловия:

Перед началом выполнения данного варианта использования пользователь должен войти в систему.

1.5.5 ПЕРЕВЕСТИ СТУДЕНТА С ГРУППЫ В ГРУППУ И С КРУСА НА КУРС

Краткое описание:

Данный вариант использования позволяет сотрудникам переводить студентов между группами или курсами. При этом формируются новые списки.

Основной поток событий:

Данный вариант использования начинает выполняться, когда по окончанию курса необходимо перевести студентов на новый курс или же перевести студента в другую группу по его желанию.

1. Система выводит информацию о студенте.

2. Пользователь выбирает куда необходимо перевести студента.

3. Система обновляет списки групп, учитывая внесенные изменения.

Предусловие:

Перед началом выполнения данного варианта использования пользователь должен войти в систему.

2. ОПИСАНИЕ СИСТЕМЫ

2.1 ВЫБОР ПРОГРАММНОГО СРЕДСТВА РЕАЛИЗАЦИИ

Для реализации проекта был использован объектно-ориентированный

язык программирования Java с платформой JavaFX. Эти инструменты позволяют разрабатывать приложение с насыщенным графиком. В качестве среды разработки был использован IntelliJ IDEA. Для создания графического интерфейса была выбрана программа Scene Builder.

2.2. ХРАНЕНИЕ ИНФОРМАЦИИ

Вся информация хранится в текстовых файлах. Например, логины и пароли пользователей хранятся в файле «users.txt» (см. рис. 1).

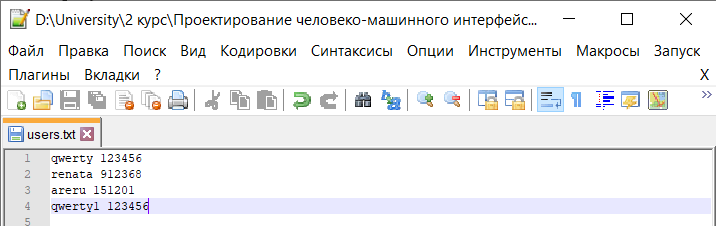


Рис. 1 Пользователи.

2.3 РЕАЛИЗАЦИЯ

В процессе реализации проекта были созданы такие классы, как: Main, LoginController, MainMenuController, RegisterController, Student и т.д.

Для удобства перемещения между файлами, они были разделены на пакеты:

assets – для хранения изображений,

sample – для хранения классов и fxml файлов.

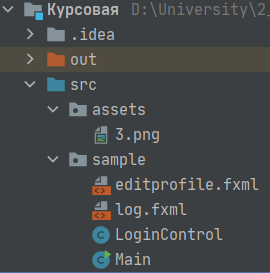


Рис. 2.

Примеры реализации:

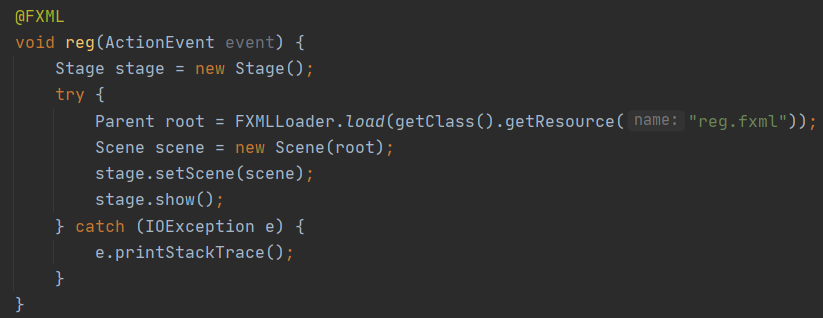


Рис. 3. Переход между сценами.



Рис. 4. Очистка полей.

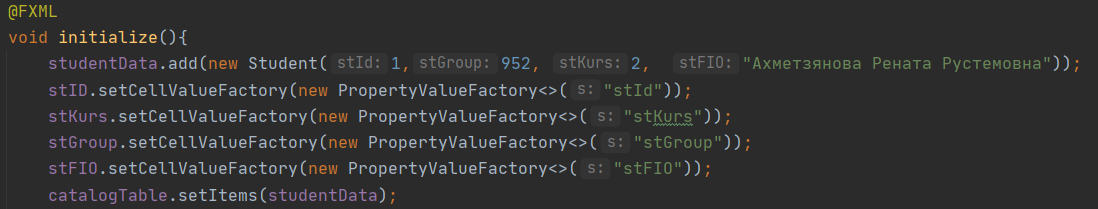


Рис. 5. Пример заполнения полей таблицы «Список студентов».



Рис. 6. Сохранение логина и пароля нового пользователя.

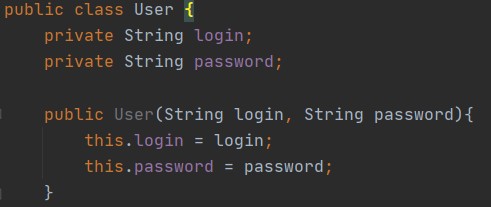


Рис. 7. Конструктор для формы регистрации.

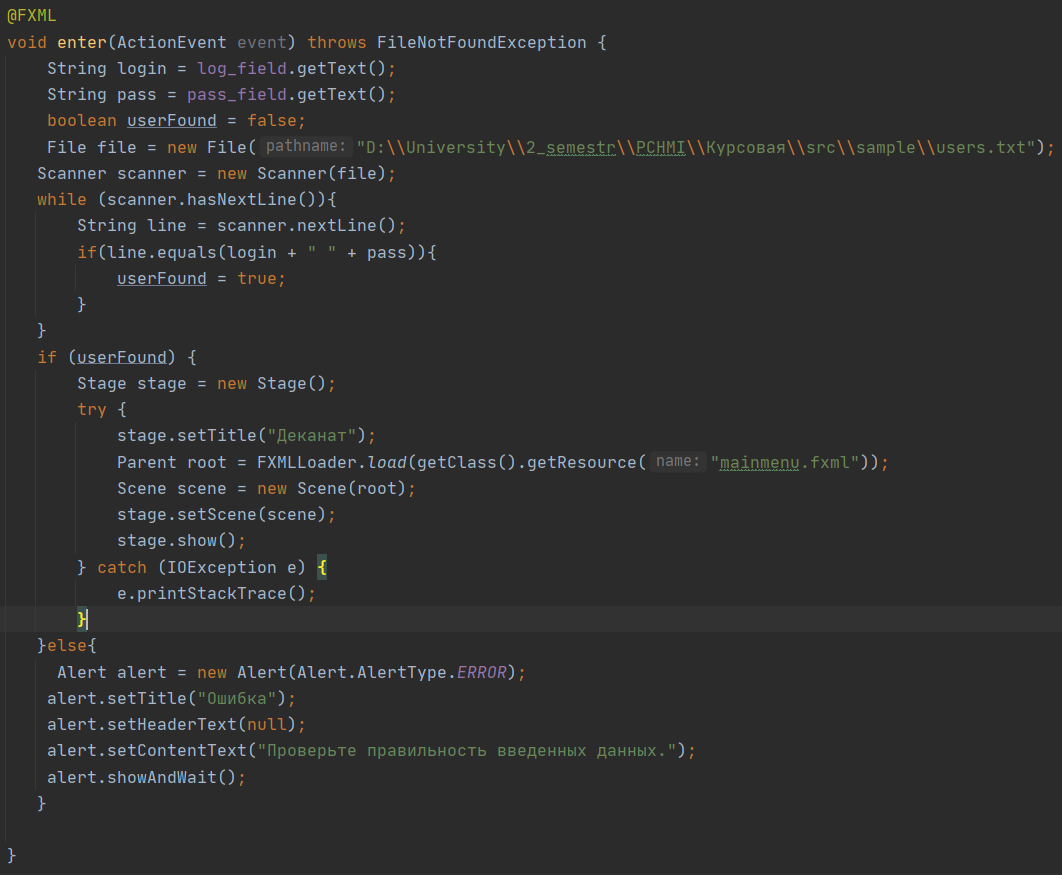


Рис. 8. Вход в систему.

3. ОПИСАНИЕ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА

Разработка пользовательского интерфейса важна не меньше написания кода программы. Необходимо правильно выбрать набор визуальных атрибутов, так как он сильно влияет на эффективность работы пользователя и его восприятие.

3.1 ВЫБОР ЦВЕТОВОЙ ПАЛИТРЫ

Для информационной системы «Деканат» решено было использовать синий и белый цвет (см. рис. 8). Такой выбор обоснован тем, что синий цвет, как и его оттенки, ассоциируются с надежностью и респектабельностью. А белый не отвлекает на себя внимание, на нем текст читабелен, так как черный цвет текста контрастирует с белым. Это поможет сотрудникам сконцентрироваться на работе и не будет напрягать глаза, что также повысит эффективность системы.

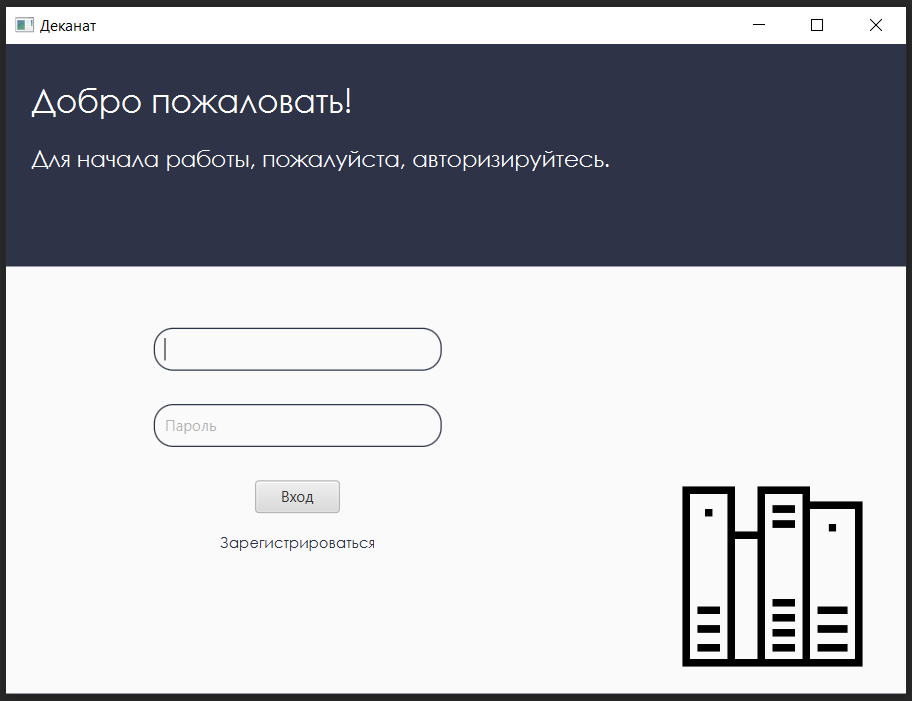


Рис. 9. Пример интерфейса.

3.2 ВЫБОР ШРИФТА

Выбор шрифта не менее важен, так как то, как написан текст, влияет на его восприятие пользователем. В создании данной системы использовался шрифт «Century Gothic». Выбор обусловлен тем, что данный шрифт минималистичный и неброский (см. рис. 9).



Рис. 10. Пример шрифта.

3.3 ОСНОВНЫЕ ЭКРАННЫЕ ЭЛЕМЕНТЫ ИНТЕРФЕЙСА

|  |  |  |
| --- | --- | --- |
| Элемент | Изображение | Назначение |
| Button |  | Перемещение пользователя с одной сцены на другую. |
| Hyperlink |  | При нажатии происходит переходит на другую страницу. Окрашен в синий цвет для сочетания с основной темой. |
| Label |  | Отображает информацию. |
| TextField |  | Используется для ввода данных |
| TableView |  | Отображает таблицу. Содержит n-ое количество колонок. |
| ImageView |  | Отображает изображение. |
| ChoiceBox |  | Используется для выбора из нескольких вариантов. |
| BarChart |  | Отображает статистику. |

4. РУКОВОДСТВО ПО ИСПОЛЬЗОВАНИЮ СИСТЕМЫ

При запуске программы открывается окно входа в систему (см. рис. 10). Новому пользователю необходимо сначала зарегистрироваться, для этого надо нажать на соответствующий Hyperlink (см. рис. 11).

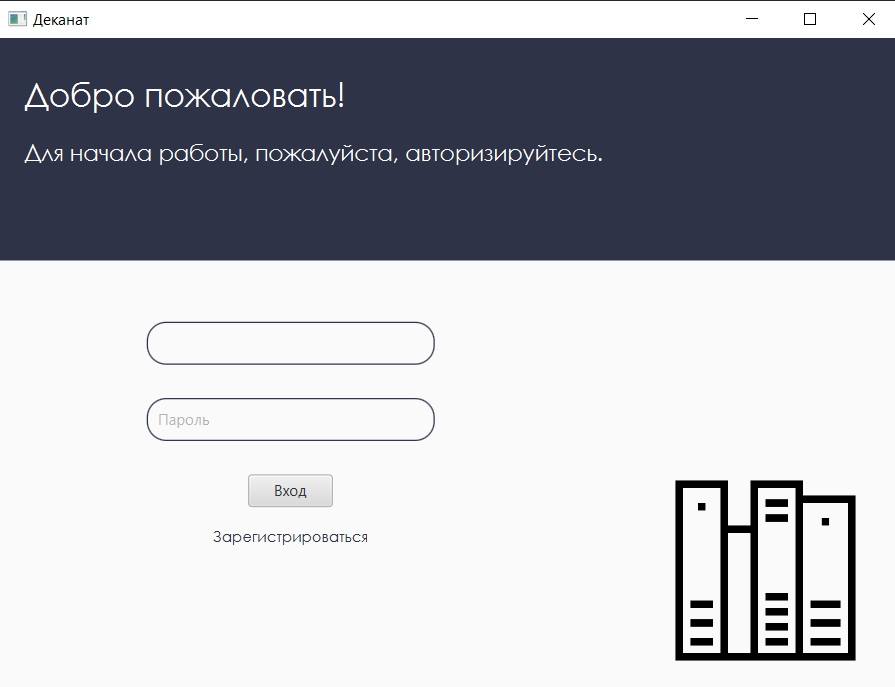


Рис. 11. Вход в систему.

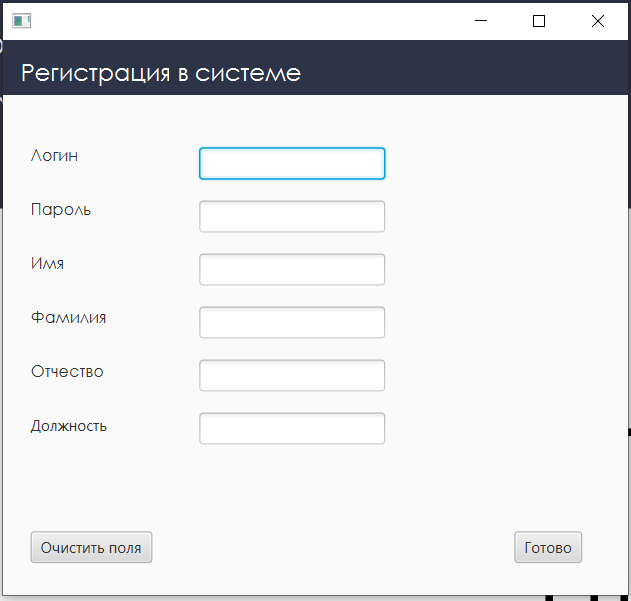


Рис. 12. Форма регистрации.

После ввода данных открывается главная сцена. Сверху можно увидеть вкладки меню. В зависимости от выбора пользователя осуществляется переход на другие вкладки.

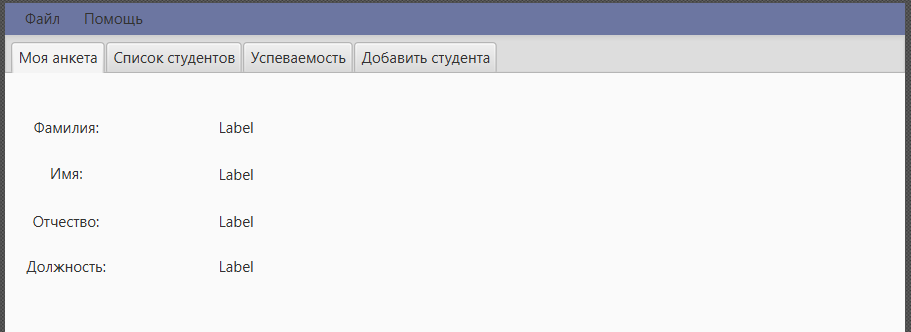


Рис. 13. Главное меню.

Заключение

В ходе выполнения данного проекта были изучены основы проектирования человеко-машинных интерфейсов. Было освоено умение работы с программой Scene Builder. Данная работа помогла сформировать представление о том, как нужно создавать простой, удобный и понятный для пользователя интерфейс, при этом придавать особое значение разработке дизайна, так как пользователь смотрит только на внешний вид приложения, и ему неинтересно то, что у него внутри.

Список использованных источников

1. Руководство по Java [Электронный ресурс] – Режим доступа: <https://metanit.com/java>
2. Руководство по JavaFX [Электронный ресурс] – Режим доступа: <https://metanit.com/java/javafx/>
3. Инструмент для подбора цветов [Электронный ресурс] – Режим доступа: <https://colorscheme.ru>
4. Сервис поиска иконок [Электронный ресурс] – Режим доступа: https://www.iconfinder.com