Министерство науки и высшего образования РФ

Федеральное государственное автономное

образовательное учреждение высшего образования

**«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

Институт космических и информационных технологий

Кафедра вычислительной техники

|  |
| --- |
| **КУРСОВОЙ ПРОЕКТ** |
| Информационная система «Бюро трудоустройства» |
|  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Руководитель |  |  |  |  | В.С. Васильев |
|  |  | подпись, дата |  | инициалы, фамилия |
|  |  |  |  |  |  |
| Студент | КИ22-06б, 032214557 |  |  |  | И.Н. Симанов |
| номер группы, зачетной книжки |  | подпись, дата |  | инициалы, фамилия |

Красноярск 2024

**РЕФЕРАТ**

Курсовой проект состоит из 41 страниц(ы) текста, 38 рисунков, 3 использованных источников и 1 приложения.

Проектирование программного обеспечения, информационная система, ICONIX, объектно- ориентированное программирование

Целью проекта является освоение навыков использования современных инструментальных сред и развитии умений самостоятельной разработки программного обеспечения с графическим интерфейсом.

Суть работы состоит в создании ПО, моделирующую информационную систему, которая позволяет управлять поиском работы и новых кадров. ИС будет обрабатывать данные о пользователях, включая ФИО, существующие вакансии, а также дополнительную информацию. Также система будет фиксировать данные о принятии на работу (соискатель, работодатель, должность, комиссионные).

В первой главе разработана спецификация требований приложения, описаны прецеденты и формы.

Во второй главе спроектированы диаграммы пригодностей и последовательностей.

В третьей главе описана реализация приложения, тестирование и инструкция по эксплуатации.

**СОДЕРЖАНИЕ**

[Введение 3](#_Toc168646063)

[1 Разработка спецификации требований 4](#_Toc168646064)

[1.1 Постановка задачи 4](#_Toc168646065)

[1.2 Выявление ролей и функций, диаграмма прецедентов 4](#_Toc168646066)

[1.3 Макеты интерфейса и текстовое описание прецедентов 7](#_Toc168646067)

[1.4 Описание форматов данных 19](#_Toc168646068)

[2 Объектно-ориентированное проектирование 21](#_Toc168646069)

[2.1 Диаграммы пригодности и последовательности для прецедента «Вход в систему» 21](#_Toc168646070)

[2.2 Диаграммы пригодности и последовательности для прецедентов «Добавить агента», «Добавить соискателя» и «Добавить работодателя» 23](#_Toc168646071)

[2.3 Диаграммы пригодности и последовательности для прецедентов «Удалить агента», «Удалить соискателя» и «Удалить работодателя» 24](#_Toc168646072)

[2.4 Диаграммы пригодности и последовательности для прецедента «Удалить вакансию» 26](#_Toc168646073)

[2.5 Диаграммы пригодности и последовательности для прецедента «Добавить вакансию» 28](#_Toc168646074)

[2.6 Диаграммы пригодности и последовательности для прецедента «Принять на работу» 30](#_Toc168646075)

[2.7 Диаграммы пригодности и последовательности для прецедента «Создание акта» 32](#_Toc168646076)

[2.8 ER-диаграмма 33](#_Toc168646077)

[2.9 Диаграмма классов 34](#_Toc168646078)

[3 Объектно-ориентированное программирование 35](#_Toc168646079)

[3.1 Выбор инструментов 35](#_Toc168646080)

[3.2 Реализация 35](#_Toc168646081)

[3.3 Сборка и запуск 36](#_Toc168646082)

[3.4 Тестирование 36](#_Toc168646083)

[3.5 Инструкция 37](#_Toc168646084)

[Заключение 38](#_Toc168646085)

[Список используемых источников 39](#_Toc168646086)

[Приложение А 40](#_Toc168646087)

# Введение

В современных условиях динамичного рынка труда создание эффективных инструментов для поиска работы и подбора персонала является приоритетной задачей. Одним из таких инструментов является приложение "Бюро трудоустройства", которое направлено на упрощение взаимодействия между соискателями и работодателями.

Цель данной курсовой работы – разработка и анализ функциональных возможностей приложения "Бюро трудоустройства", которое позволит автоматизировать процессы поиска работы и подбора кадров, повысить эффективность и скорость трудоустройст­­­ва, а также обеспечить удобство использования для всех участников рынка труда. В ходе исследования будут рассмотрены основные этапы разработки приложения, включая анализ требований, проектирование интерфейса, реализацию и тестирование.

1 Разработка спецификации требований

1.1 Постановка задачи

Перед формированием требований, были спроектированы задачи информационной системы:

* добавление агентов;
* удаление агентов.
* добавление соискателей;
* удаление соискателей.
* добавление работодателей;
* удаление работодателей.
* отклик соискателя на вакансию;
* принятие на работу;
* составление акта.

1.2 Выявление ролей и функций, диаграмма прецедентов

Были выявлены следующие роли: администратор, агент, соискатель, работодатель.

На рисунке 1 представлена диаграмма прецедентов администратора.

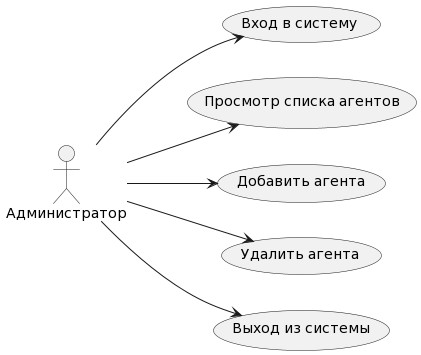


Рисунок 1 – Диаграмма прецедентов администратора

На рисунке 2 представлена диаграмма прецедентов агента.

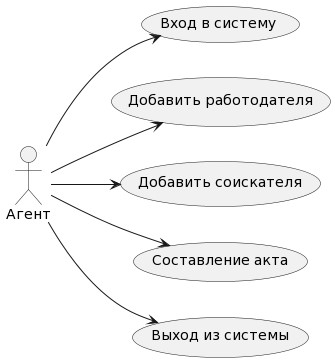


Рисунок 2 – Диаграмма прецедентов агента

На рисунке 3 представлена диаграмма прецедентов соискателя.

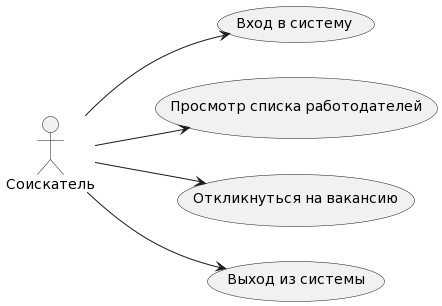


Рисунок 3 – Диаграмма прецедентов соискателя

На рисунке 4 представлена диаграмма прецедентов работодателя.

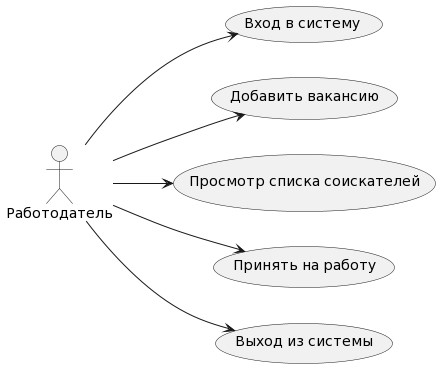


Рисунок 4 – Диаграмма прецедентов работодателя

В приложении А представлены диаграммы потоков экранов.

1.3 Макеты интерфейса и текстовое описание прецедентов

**Роли:** агент, администратор, соискатель, работодатель.

**Цель сценария:** войти в систему.

**Предусловие:** открыто «Окно входа в систему» (рисунок 5).

**Основной сценарий:**

* ввести ФИО;
* нажать кнопку «Войти».

**Постусловие:** в базу данных совершится запрос на поиск данного пользователя. Откроется «Домашнее окно» соответствующего пользователя.

**Альтернативный сценарий**: появится окно об ошибке.

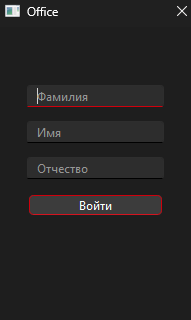


Рисунок 5 – Окно входа в систему

**Роль:** администратор.

**Цель сценария:** добавить агента.

**Предусловие:** выполнен вход в качестве администратора.

**Основной сценарий:**

* нажать на вкладку «Добавить агента»;
* в «Окне добавления агентов» (рисунок 6) ввести ФИО, телефон;
* нажать кнопку «Добавить».

**Постусловие:** в базу данных добавлена запись об агенте. В окне появится надпись об успешной регистрации.

**Альтернативный сценарий:** в окне появится надпись об ошибке.

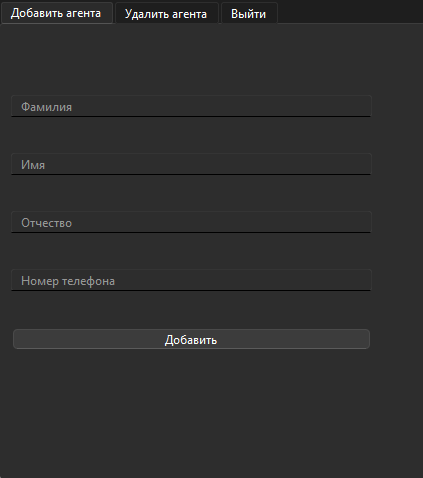


Рисунок 6 – Окно добавления агента

**Роль:** администратор.

**Цель сценария:** удалить агента.

**Предусловие:** выполнен вход в качестве администратора.

**Основной сценарий:**

* + нажать на вкладку «Удалить агента»;
  + в «Окне удаления агентов» (рисунок 7) выбрать строку с информацией о пользователе;
  + нажать кнопку «Удалить».

**Постусловие:** из базы данных удалена запись об агенте. В окне появится надпись об успешной регистрации.

**Альтернативный сценарий:** в окне появится надпись об ошибке.

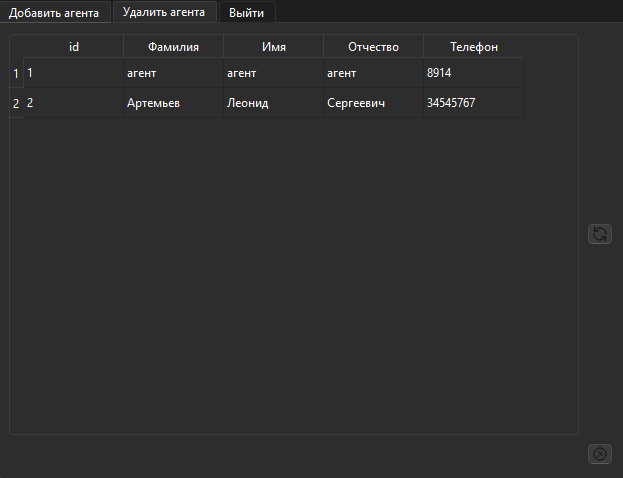


Рисунок 7 – Окно удаления агента

**Роль:** агент.

**Цель сценария:** добавить соискателя.

**Предусловие:** выполнен вход в качестве агента.

**Основной сценарий:**

* нажать на вкладку «Добавить соискателя»;
* в «Окне добавления соискателя» (рисунок 8) ввести ФИО, номер телефона и стаж;
* нажать кнопку «Добавить».

**Постусловие:** в базу данных добавлена запись о соискателе. В окне появится надпись об успешной регистрации.

**Альтернативный сценарий:** в окне появится надпись об ошибке.

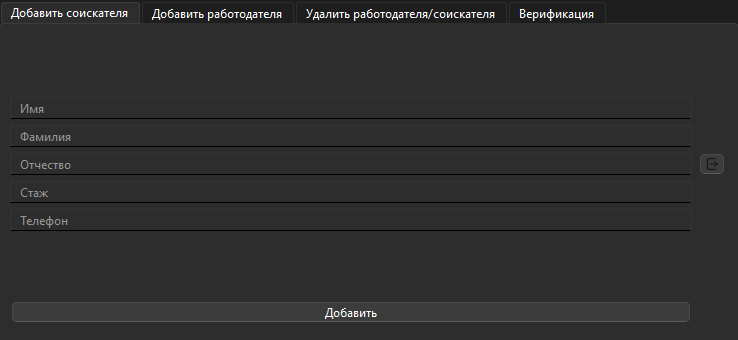


Рисунок 8 – Окно добавления соискателя

**Роль:** агент.

**Цель сценария:** добавить работодателя.

**Предусловие:** выполнен вход в качестве агента.

**Основной сценарий:**

* нажать на вкладку «Добавить работодателя»;
* в «Окне добавления работодателя» (рисунок 9) ввести ФИО, номер телефона и ИНН;
* нажать кнопку «Добавить».

**Постусловие:** в базу данных добавлена запись о работодателе. В окне появится надпись об успешной регистрации.

**Альтернативный сценарий:** в окне появится надпись об ошибке.

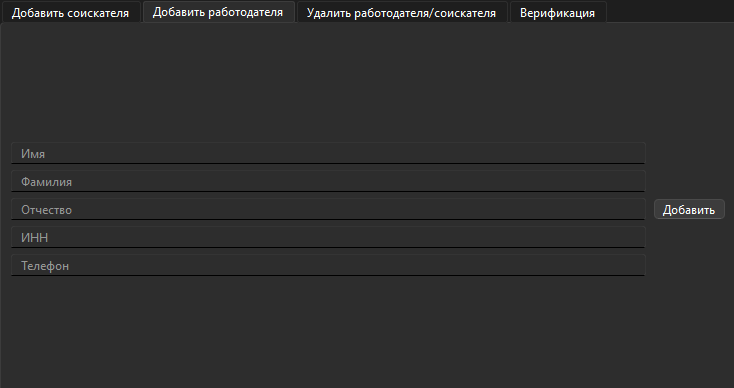


Рисунок 9 – Окно добавления работодателя

**Роль:** агент.

**Цель сценария:** удалить работодателя.

**Предусловие:** выполнен вход в качестве агента.

**Основной сценарий:**

* перейти в раздел «Удалить работодателя/соискателя» (рисунок 10);
* нажать на вкладку «Работодатели»;
* выбрать строку с информацией о работодателе;
* нажать кнопку «Удалить».

**Постусловие:** из базы данных удалена запись о работодателе. В окне появится надпись об успешном удалении.

**Альтернативный сценарий:** в окне появится надпись об ошибке.

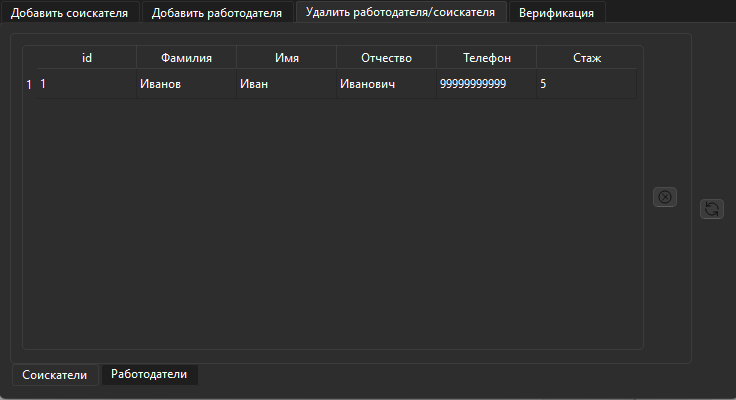


Рисунок 10 – Окно удаления работодателя

**Роль:** агент.

**Цель сценария:** удалить соискателя.

**Предусловие:** выполнен вход в качестве агента.

**Основной сценарий:**

* перейти в раздел «Удалить работодателя/соискателя» (рисунок 11);
* нажать на вкладку «Соискатели»;
* выбрать строку с информацией о соискателе;
* нажать кнопку «Удалить».

**Постусловие:** из базы данных удалена запись о соискателе. В окне появится надпись об успешном удалении.

**Альтернативный сценарий:** в окне появится надпись об ошибке.

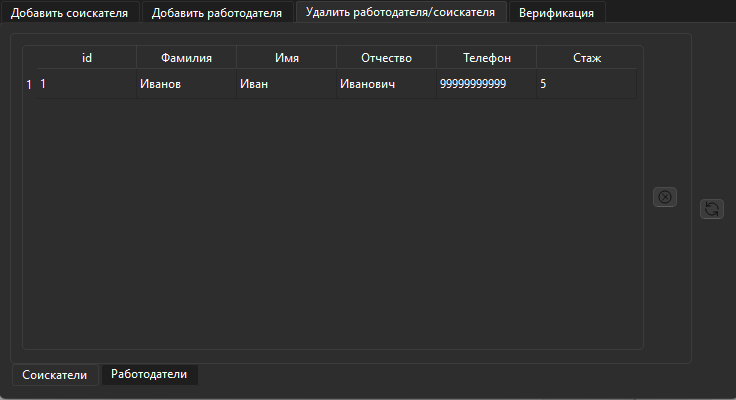


Рисунок 11 – Окно удаления соискателя

**Роль:** агент.

**Цель сценария:** составление акта.

**Предусловие:** выполнен вход в качестве агента.

**Основной сценарий:**

* перейти в раздел «Верификация» (рисунок 12);
* выбрать строку из существующих заявок;
* нажать кнопку «Верифицировать»;
* в новом окне ввести сумму комиссионных и нажать кнопку «ОК».

**Постусловие:** в базе данных добавляется запись об акте. В окне появляется запись об успешной верификации акта.

**Альтернативный сценарий:** при невозможности верифицировать акт, появляется запись об ошибке.

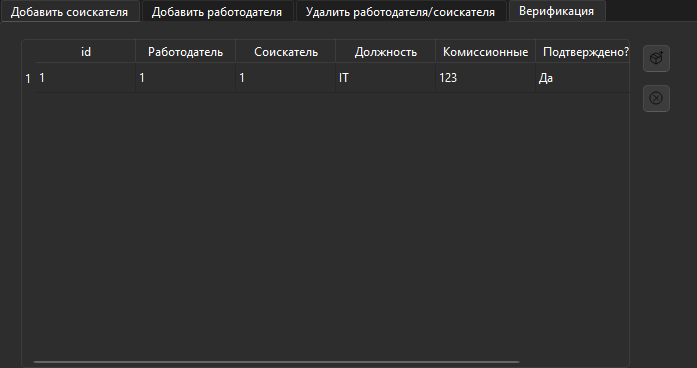


Рисунок 12 – Окно составления акта

**Роль:** соискатель.

**Цель сценария:** откликнуться на вакансию.

**Предусловие:** выполнен вход в качестве соискателя.

**Основной сценарий:**

* выбрать из первого выпадающего списка работодателя (рисунок 13);
* выбрать из второго выпадающего списка вакансию;
* нажать кнопку «Откликнуться».

**Постусловие:** создана запись о заявке в базе данных.

**Альтернативный сценарий:** при невозможности откликнуться на вакансию, появляется запись об ошибке.

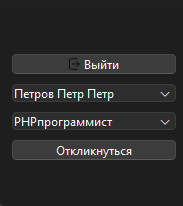


Рисунок 13 – Окно отклика на вакансию

**Роль:** работодатель.

**Цель сценария:** создать вакансию.

**Предусловие:** выполнен вход в качестве работодателя.

**Основной сценарий:**

* перейти во вкладку «Добавить/удалить вакансии» (рисунок 14);
* ввести название вакансии в текстовое поле;
* нажать на кнопку «Добавить».

**Постусловие:** в базу данных добавлена запись о вакансии.

**Альтернативный сценарий:** в окне появится надпись об ошибке.

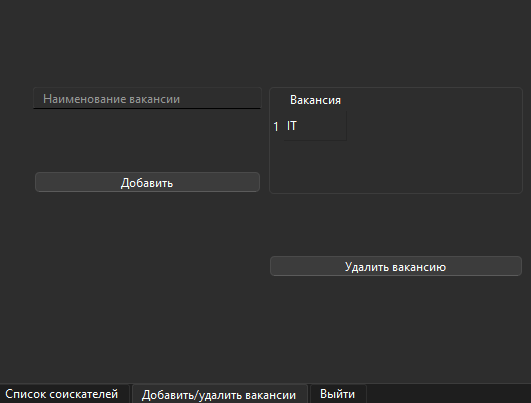


Рисунок 14 – Окно добавления/удаления вакансии

**Роль:** работодатель.

**Цель сценария:** удалить вакансию.

**Предусловие:** выполнен вход в качестве работодателя.

**Основной сценарий:**

* перейти во вкладку «Добавить/удалить вакансии» (рисунок 14);
* выбрать вакансию из списка;
* нажать на кнопку «Удалить вакансию».

**Постусловие:** из базы данных удаляется запись о вакансии.

**Альтернативный сценарий:** если вакансия используется в других записях, появляется окно об ошибке.

**Роль:** работодатель.

**Цель сценария:** принятие заявки на работу.

**Предусловие:** выполнен вход в качестве работодателя.

**Основной сценарий:**

* Перейти во вкладку «Список соискателей» (рисунок 15);
* Выбрать соискателя из списка откликов;
* Нажать на кнопку «Отправить на верификацию».

**Постусловие:** в базе данных перемещается запись о заявке в сторону агента.

**Альтернативный сценарий:** если запись не удаётся вставить в новую таблицу или удалить из старой, появляется окно об ошибке.

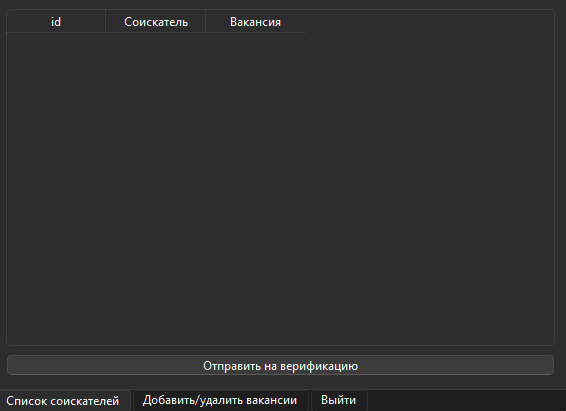
****

Рисунок 15 – Окно принятия заявки соискателя

1.4 Описание форматов данных

Важной частью функционирования системы является хранение и обработка данных. На рисунке 16 представлен пример заполнения администраторов в базу данных.



Рисунок 16 – Пример заполнения администраторов в базу данных

На рисунке 17 представлен пример заполнения агентов в базу данных.

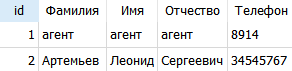


Рисунок 17 – Пример заполнения агентов в базу данных

На рисунке 18 представлен пример заполнения соискателей в базу данных.



Рисунок 18– Пример заполнения соискателей в базу данных

На рисунке 19 представлен пример заполнения работодателей в базу данных.

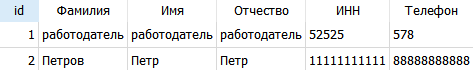


Рисунок 19 – Пример заполнения работодателей в базу данных

На рисунке 20 представлен пример заполнения вакансий в базу данных.

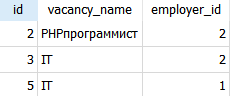


Рисунок 20 – Пример заполнения вакансий в базу данных

На рисунке 21 представлен пример заполнения заявок в базу данных.



Рисунок 21 – Пример заполнения заявок в базу данных

На рисунке 22 представлен пример актов в базу данных.



Рисунок 22 – Пример заполнения актов в базу данных

**2 Объектно-ориентированное проектирование**

2.1 Диаграммы пригодности и последовательности для прецедента «Вход в систему»

На рисунке 23 представлена диаграмма пригодности для данного прецедента.

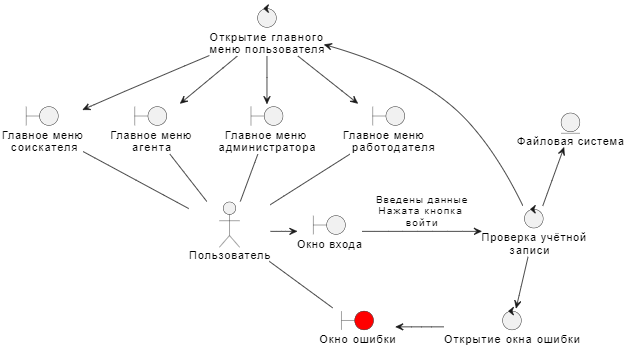


Рисунок 23 – Диаграмма пригодности прецедента «Вход в систему»

На рисунке 24 представлена диаграмма последовательности для этого прецедента.

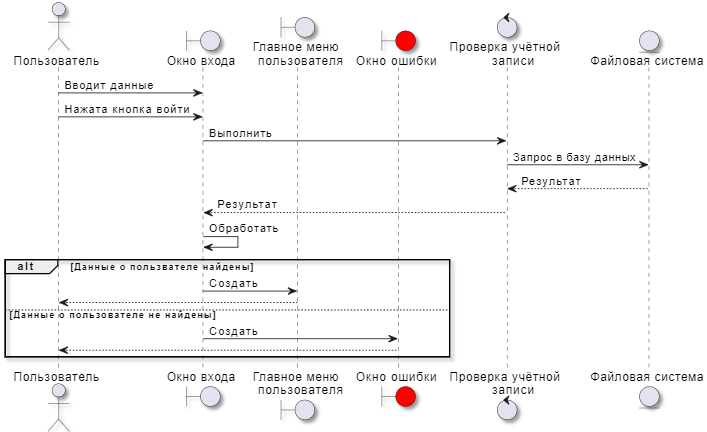


Рисунок 24 – Диаграмма последовательности прецедента «Вход в систему»

2.2 Диаграммы пригодности и последовательности для прецедентов «Добавить агента», «Добавить соискателя» и «Добавить работодателя»

В связи с тем, что прецеденты «Добавить агента», «Добавить соискателя», и «Добавить работодателя» имеют единый принцип работы, они будут описаны единой диаграммой пригодности, общей для каждого из прецедентов.

На рисунке 25 представлена диаграмма пригодности для данных прецедентов.

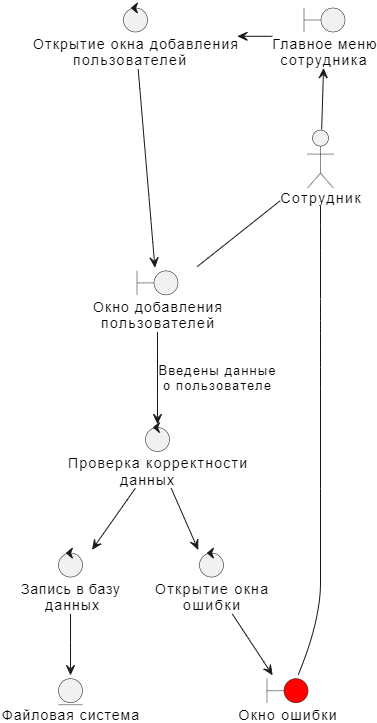


Рисунок 25 – Диаграмма пригодности прецедента «Добавление пользователя»

На рисунке 26 представлена диаграмма последовательности данного прецедента.

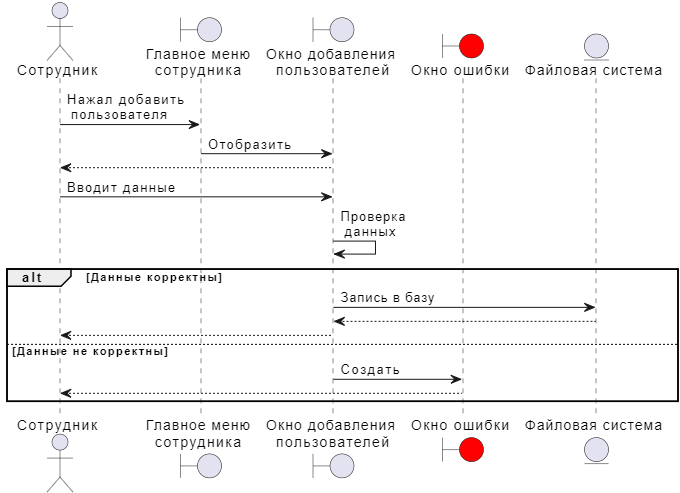


Рисунок 26 – Диаграмма последовательности прецедента «Добавление пользователя»

2.3 Диаграммы пригодности и последовательности для прецедентов «Удалить агента», «Удалить соискателя» и «Удалить работодателя»

В связи с тем, что прецеденты «Удалить агента», «Удалить соискателя», и «Удалить работодателя» имеют единый принцип работы, они будут описаны единой диаграммой пригодности, общей для каждого из прецедентов.

На рисунке 27 представлена диаграмма пригодности для данного прецедента.

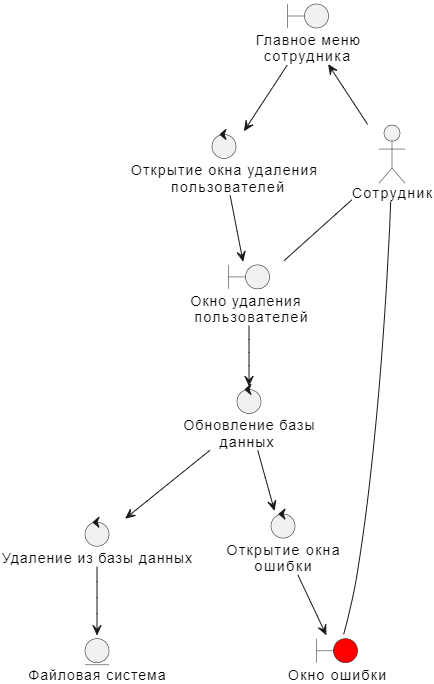


Рисунок 27 – Диаграмма пригодности прецедента «Удаление пользователя»

На рисунке 28 представлена диаграмма последовательности для данного прецедента.

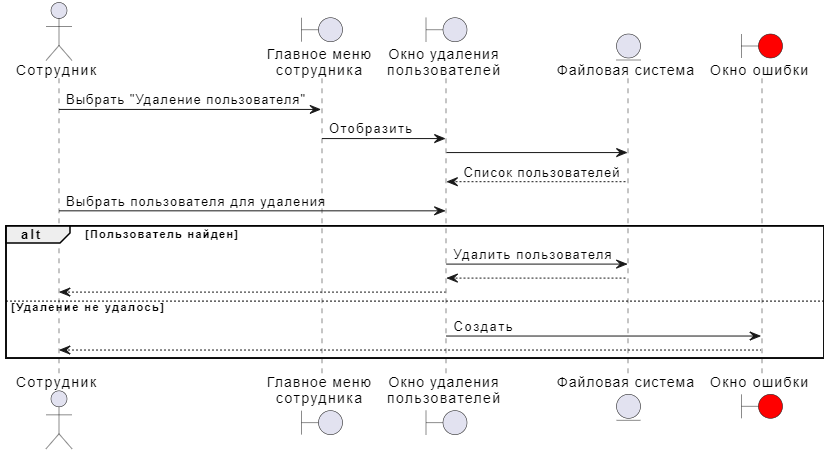


Рисунок 28 – Диаграмма последовательности прецедента «Удаление пользователя»

2.4 Диаграммы пригодности и последовательности для прецедента «Удалить вакансию»

На рисунке 29 представлена диаграмма пригодности для данного прецедента.



Рисунок 29 – Диаграмма пригодности прецедента «Удалить вакансию»

На рисунке 30 представлена диаграмма последовательности для данного прецедента.

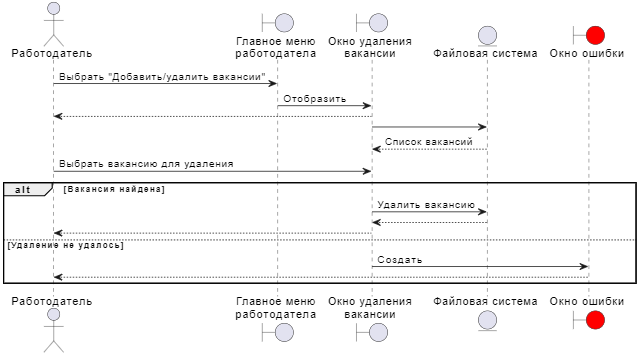


Рисунок 30 – Диаграмма последовательности прецедента «Удалить вакансию»

2.5 Диаграммы пригодности и последовательности для прецедента «Добавить вакансию»

На рисунке 31 представлена диаграмма пригодности для данного прецедента.

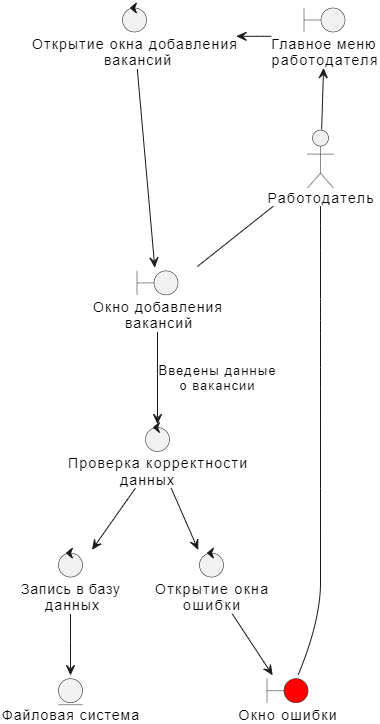


Рисунок 31 – Диаграмма пригодности прецедента «Добавить вакансию»

На рисунке 32 представлена диаграмма последовательности для данного прецедента.

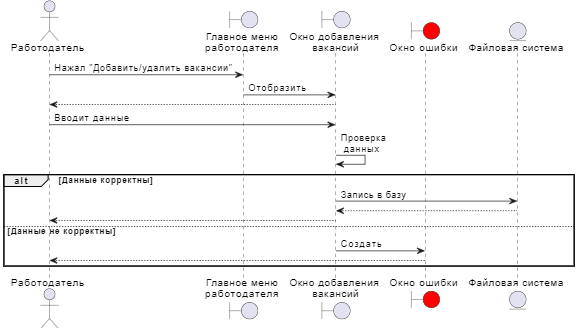


Рисунок 32 – Диаграмма последовательности прецедента «Добавить вакансию»

2.6 Диаграммы пригодности и последовательности для прецедента «Принять на работу»

На рисунке 33 представлена диаграмма пригодности для данного прецедента.

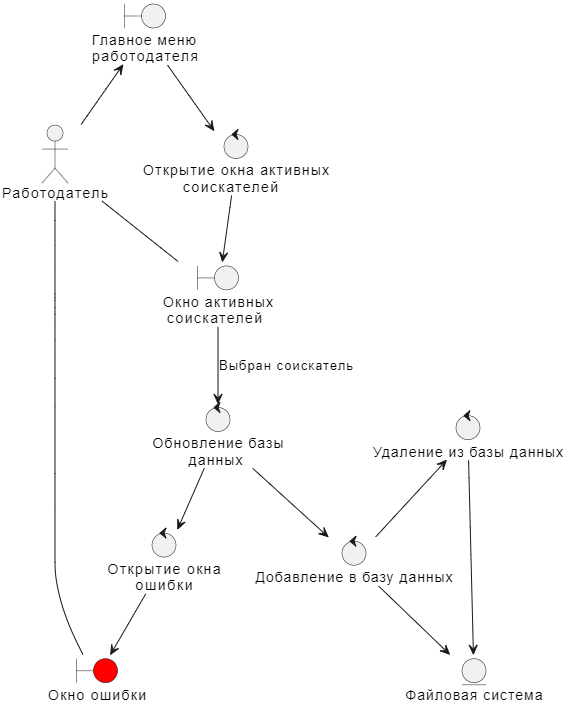


Рисунок 33 – Диаграмма пригодности прецедента «Принять на работу»

На рисунке 34 представлена диаграмма последовательности для данного прецедента.

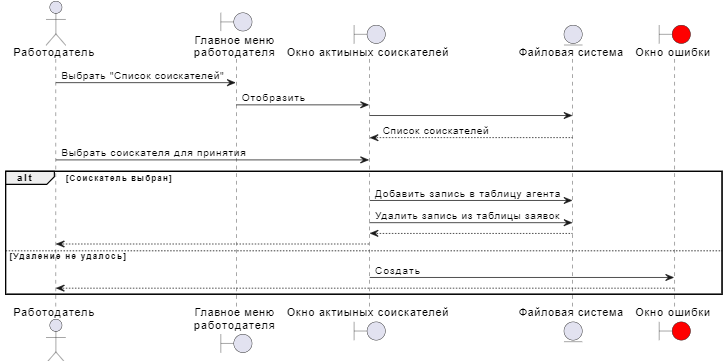


Рисунок 34 – Диаграмма последовательности прецедента «Принять на работу»

2.7 Диаграммы пригодности и последовательности для прецедента «Создание акта»

На рисунке 35 представлена диаграмма пригодности для данного прецедента.

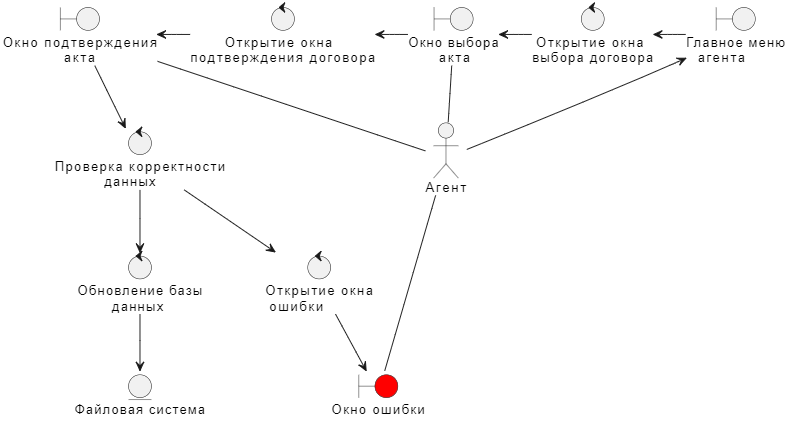


Рисунок 35 – Диаграмма пригодности прецедента «Создание акта»

На рисунке 36 представлена диаграмма последовательности для данного прецедента.

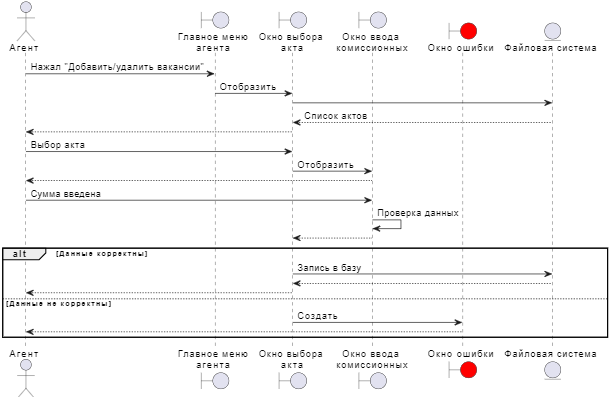


Рисунок 36 – Диаграмма последовательности прецедента «Создание акта»

2.8 ER-диаграмма

На рисунке 37 представлена ER-диаграмма в нотации Crow's Foot.

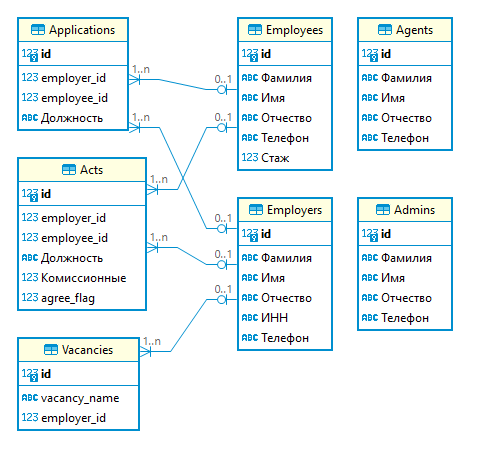


Рисунок 37 – ER-диаграмма

2.9 Диаграмма классов

На рисунке 38 представлена диаграмма классов.

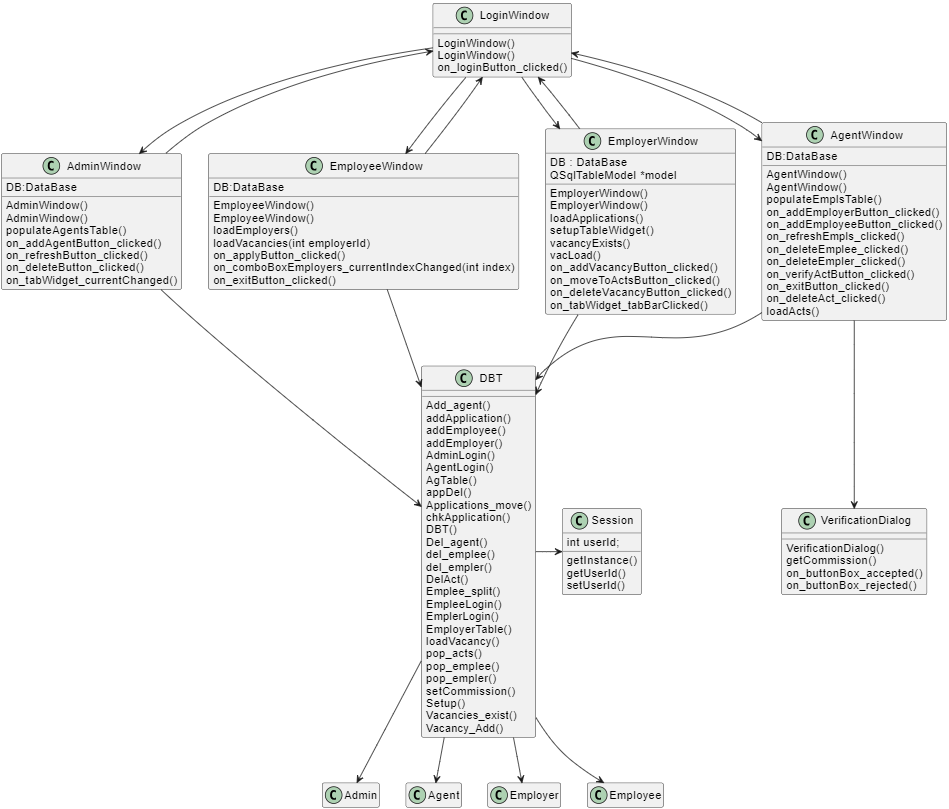


Рисунок 38 – Диаграмма классов

**3 Объектно-ориентированное программирование**

3.1 Выбор инструментов

Для разработки ПК-приложения необходимы библиотеки для создания графического интерфейса, работы с базами данных. Для этих целей отлично подходит язык программирования C++ вместе с фреймворком Qt. Qt предоставляет широкий выбор инструментов для графического интерфейса, включая стандартные Qt Widgets, и содержит всё необходимое для работы с моделями, списками и представлениями.

3.2 Реализация

В приложении используется достаточно большое количество окон, в которых необходимо взаимодействовать с различными данными. Для организации работы с данными была выбрана библиотека QSqlQuery, потому что для неё достаточно лишь указать SQL запрос, и заполнить таблицу результатом. В качестве средства отображения данных используется QTableWidget.

Для хранения данных используется база данных SQLite. Она была выбрана, потому что является встраиваемой и пользователю не придётся дополнительно устанавливать серверную часть СУБД. Все функции для работы с БД собраны в классе Database. При первом запуске программы в базе данных создаются учётные записи для следующих пользователей:

* администратор: Симанов Иван Николаевич;
* агент: Еремеев Олег Львович;
* соискатель: Петров Илья Максимович;
* работодатель: Серов Герман Миронович.

3.3 Сборка и запуск

Для сборки программы для использования на устройствах, не имеющих ПО Qt, необходимо провести сборку exe файла, состоящую из следующих действий на ОС Windows:

* в Qt Creator необходимо сменить режим сборки с «Debug» до «Release»;
* перенести exe файл из папки сборки в отдельную директорию;
* по пути «C:\Qt\версия\_Qt\текущий\_компилятор\bin» найти файл windeployqt6.exe или windeployqt.exe;
* перенести файл собранного проекта на данный файл или открыть папку bin в терминале и вызвать команду «windeployqt6 \*», где \* - путь к exe файлу проекта.

Данные действия создадут в папке с exe файлом необходимые для запуска файлы формата .dll и позволят запускать программу на тех системах, где не установлен Qt.

3.4 Тестирование

Тестирование проводилось вручную. Для проверки корректности прецедента «Добавить агента», выполнялись следующие действия:

* + ввод данных, уже существующих в таблице;
  + одно или несколько полей оставались пустыми;
  + попытка ввода в поля цифр или знаков пунктуации;

Для проверки корректности прецедента «Удалить соискателя», выполнялись следующие действия:

* + попытка удаления данных при наличии зависимых записей;
  + удаление без выбора пользователя.

Аналогичное тестирование было проведено для всех прецедентов, описанных в первом разделе.

По окончанию тестирования были выявлены и устранены следующие ошибки:

* при удалении соискателя или работодателя, запись удалялась несмотря на то, что в БД существовали записи, содержащие ID пользователей;
* при добавлении вакансии работодателем была возможность добавить несколько идентичных записей.

3.5 Инструкция

В качестве инструкций для пользователя можно использовать прецеденты, описанные в первом разделе и диаграммы последовательности, разработанные во втором разделе.

Заключение

В результате выполнения курсового проекта была спроектирована и реализована информационная система, которая полностью соответствует данному техническому заданию. Получен опыт проектирования информационных систем с использованием диаграмм UML, а также опыт работы с инструментами разработки ПО.

# Список используемых источников

1. СТУ 7.5–07–2021. Стандарт университета «Система менеджмента качества. Общие требования к построению, изложению и оформлению документов учебной деятельности».
2. Документация Qt // QT | Tools for Each Stage of Software Development Lifecycle. – URL: <https://doc.qt.io/> (дата обращения 12.05.2024).
3. Диаграммы классов UML // Блог программиста – программирование и алгоритмы. – URL: <https://pro-prof.com/> (дата обращения 25.05.2024).

# Приложение А

**Диаграмма потоков экранов**

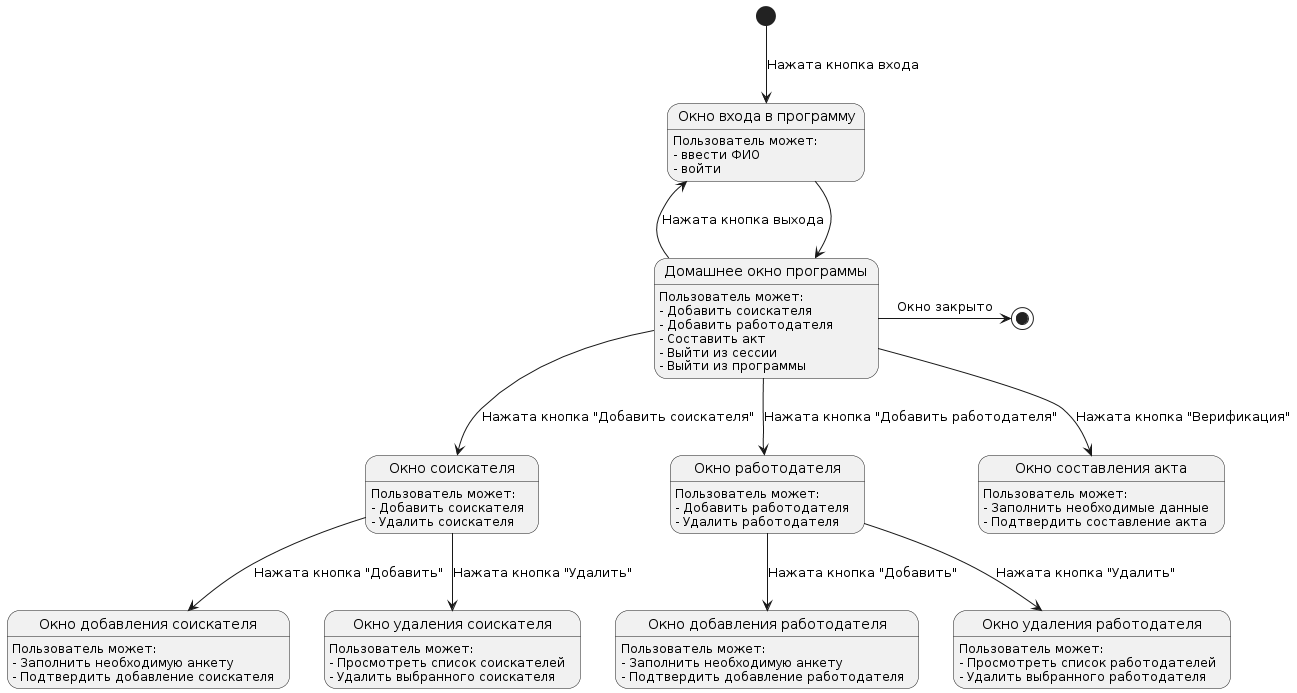


Рисунок А.1 – Диаграмма потоков экранов агента



Рисунок А.2 – Диаграмма потоков экранов администратора

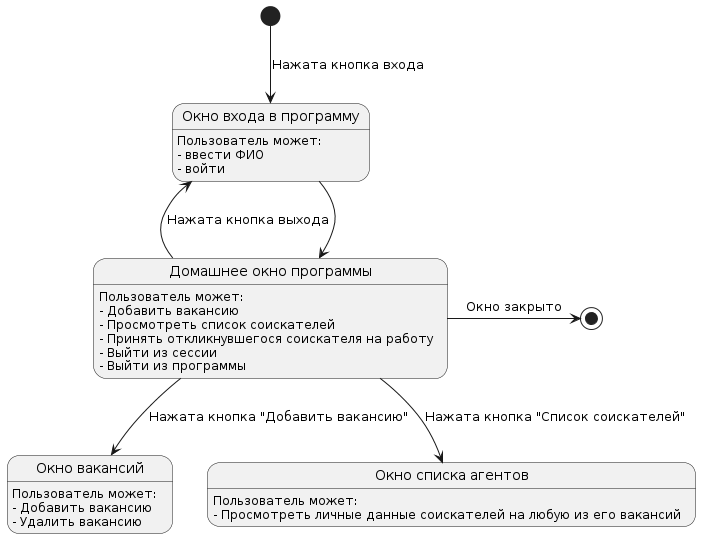


Рисунок А.3 – Диаграмма потоков экрана работодателя

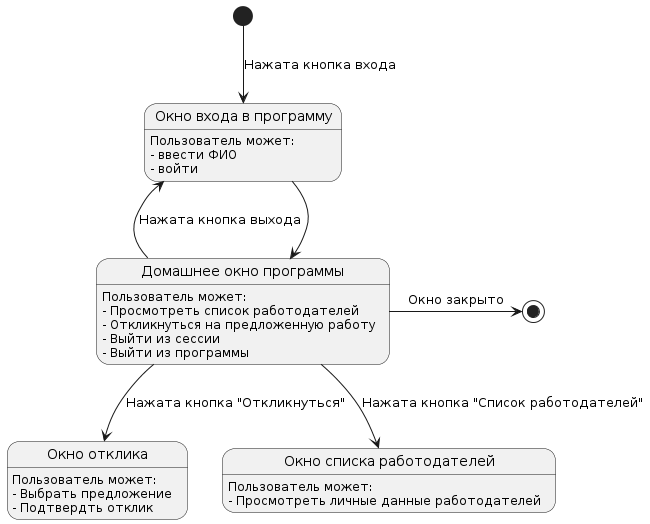


Рисунок А.4 – Диаграмма потоков экрана соискателя