Федеральное государственное автономное

образовательное учреждение

высшего образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт Космических и информационных технологий

Кафедра вычислительной техники

**КУРСОВОЙ ПРОЕКТ**

Информационная система “Бюро по трудоустройству”

Вариант № 6

|  |  |  |
| --- | --- | --- |
| Преподаватель | \_\_\_\_\_\_\_\_\_\_ | В.С. Васильев |
|  | подпись |  |
| Студент КИ20-08Б, 032049025 | \_\_\_\_\_\_\_\_\_\_ | А.Н. Головань |
|  | подпись |  |

Красноярск 2021

**РЕФЕРАТ**

Курсовой проект состоит из 6 таблиц и 23 рисунков.

ИНФОРМАЦИОННАЯ СИСТЕМА, ПРОГРАММА, ПОЛЬЗОВАТЕЛЬ, РАБОТНИК, РАБОТОДАТЕЛЬ, ОБЪЕКТНО-ОРИЕНТИРОВАНННОЕ ПРОГРАММИРОВАНИЕ, КЛАСС, МЕТОД.

Целью работы является формирование умений и навыков работы в современных инструментальных средах, приобретение и развитие практических навыков самостоятельной работы при разработке веб-сайта.

Суть работы состоит в создании ПО, моделирующего информационную систему (ИС), которая позволяет ускорить процесс поиска работников и работодателей.

Курсовой проект разделен на четыре части: разработка спецификации требований, проектирование, разработка ПО и заключительный этап — документирование - результатом которого и является настоящая пояснительная записка.

Для решения поставленной задачи, то есть для создания ПО, использующий веб-сайт, был выбран язык программирования Python. Веб-сайт был представлен набором инструментов HTML+JS+CSS и библиотекой для Python - Flask. Данный набор инструментов позволяет создавать переносимое ПО в рамках основных архитектур.

СОДЕРЖАНИЕ

[Введение 5](#_Toc91258267)

[1 Постановка задачи 5](#_Toc91258268)

[2 Разработка спецификаций требований 6](#_Toc91258269)

[2.1 Выявление ролей и функций 6](#_Toc91258270)

[2.2 Разработка макетов интерфейса 6](#_Toc91258271)

[2.3.1 Макет интерфейса входа 7](#_Toc91258275)

[2.3.2 Макет интерфейса регистрации 7](#_Toc91258276)

[2.3.3 Макет профиля пользователя 8](#_Toc91258277)

[2.3.4 Макет просмотра вакансий 8](#_Toc91258278)

[2.3.5 Макет добавления вакансий 8](#_Toc91258279)

[2.3.6 Макет просмотра резюме 9](#_Toc91258280)

[2.3.7 Макет добавления резюме 9](#_Toc91258281)

[2.3 Описание формата данных 10](#_Toc91258285)

[2.3.1 Информация о пользователях 10](#_Toc91258287)

[2.3.2 Информация о вакансиях 11](#_Toc91258294)

[4.4.3 Информация о резюме 11](#_Toc91258302)

[2.4 Описание поведения программы 12](#_Toc91258303)

[4.4.1 Вход в систему 12](#_Toc91258305)

[4.4.2 Регистрация 13](#_Toc91258306)

[4.4.3 Добавление вакансии 13](#_Toc91258307)

[4.4.4 Добавление резюме 14](#_Toc91258308)

[4.4.5 Удаление вакансии 14](#_Toc91258309)

[4.4.6 Удаление резюме 14](#_Toc91258310)

[3 Разработка спецификаций требований 15](#_Toc91258311)

[3.1 Разработка диаграммы пригодности 15](#_Toc91258312)

[3.2 Разработка диаграммы последовательности 19](#_Toc91258313)

[3.3 Разработка диаграммы классов уровня проектирования 20](#_Toc91258314)

[4 Объектно-ориентированное программирование 21](#_Toc91258315)

[4.1 Разработка модульных тестов 21](#_Toc91258316)

[4.2 Реализация программы 23](#_Toc91258317)

[4.3 Рефакторинг 23](#_Toc91258318)

[4.4 Документирование 23](#_Toc91258319)

[4.4.1 Сборка из исходных данных 23](#_Toc91258320)

[4.4.1.1 Зависимости 24](#_Toc91258321)

[4.4.1.2 Windows 24](#_Toc91258322)

[4.4.2 Инструкция по использованию 24](#_Toc91258323)

[Заключение 25](#_Toc91258324)

[Список использованных источников 26](#_Toc91258325)

**ВВЕДЕНИЕ**

На данных момент все больше стали цениться высокие технология, которые могут автоматизировать процесс или ускорить. Такое решение позволяет облегчить работу людей, а также свести многие ошибки к нулю. Поиск работников и работодателей требовало бы больше времени. Поэтому так важно ускорить этот процесс и убедится в удобности.

Целью работы является создание ПО для решения перечисленных проблем.

Информационная система «Бюро по трудоустройству» предназначена для более быстрого и удобного поиска работников и работодателей. С помощью нее можно добавлять/удалять вакансии, а также добавлять/удалять резюме. При этом эти операции не требуют большой ручной работы – достаточно просто ввести пару данных о себе.

1. **Постановка задачи**

Бюро по трудоустройству. Бюро готово искать работников для различных

работодателей и вакансии для ищущих работу специалистов различного профиля. При обращении в бюро работодателя его стандартные данные (название, вид деятельности, адрес, телефон) фиксируются в базе данных. При обращении в бюро соискателя его стандартные данные (фамилия, имя, отчество, квалификация, профессия, иные данные) также фиксируются в базе данных. По каждому факту удовлетворения интересов обеих сторон составляется документ. В документе указываются соискатель, работодатель, должность и комиссионные (доход бюро). В базе должна фиксироваться не только сделка, но и информация по открытым вакансиям. Кроме того для автоматического поиска вариантов необходимо вести справочник «Виды деятельности».

1. **Разработка спецификаций требований**

1. **Выявление ролей и функций**

Диаграмма прецедентов представлена на рисунке 1.

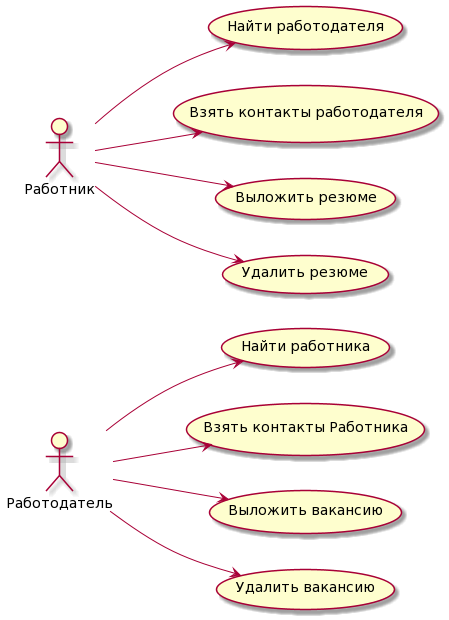


Рисунок 1 - Диаграмма прецедентов

1. **Разработка макетов интерфейса**
2. 2. 1. **Макет интерфейса входа**

Для входа в систему используется макет интерфейса, приведенный на рисунке 2.

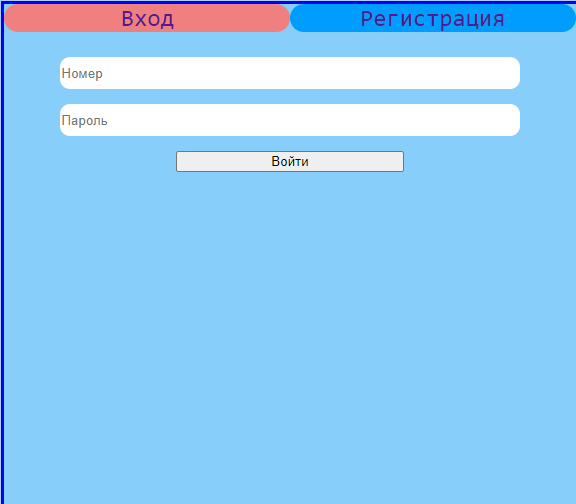


Рисунок 2 - Окно авторизации

* + 1. **Макет интерфейса регистрации**

Для регистрации пользователя используется макет интерфейса, приведенный на рисунке 3.

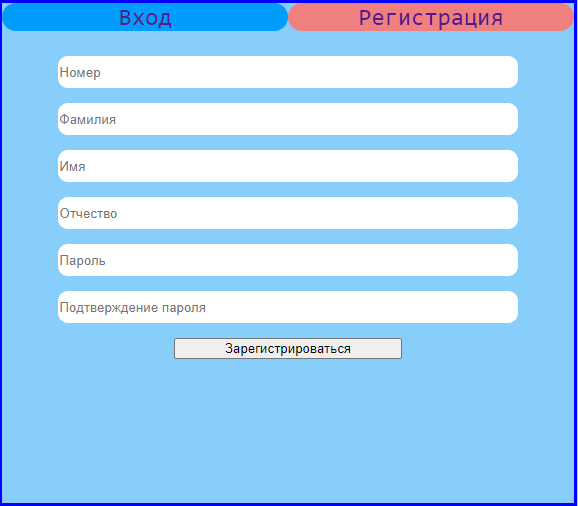


Рисунок 3 - Окно регистрации

* + 1. **Макет профиля пользователя**

Для просмотра информации о пользователе используется макет интерфейса, приведенный на рисунке 4.

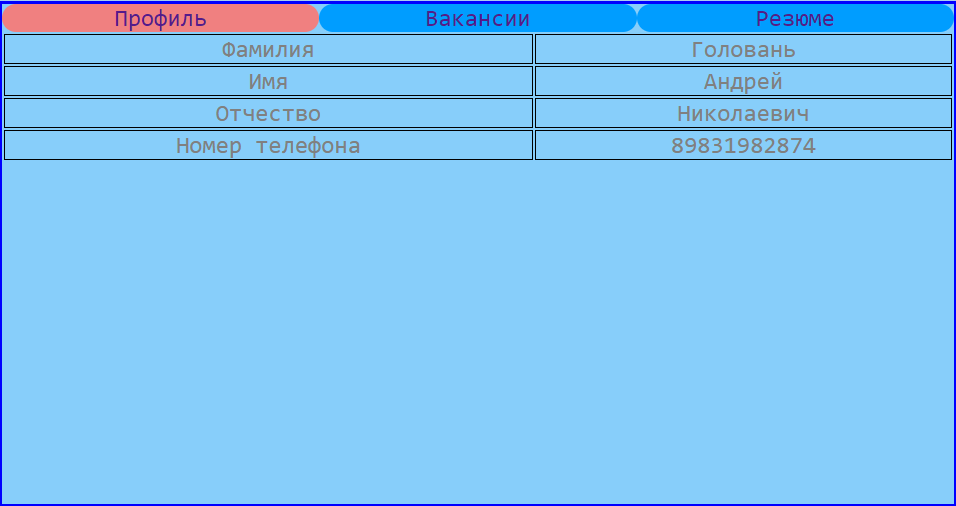


Рисунок 4 - Окно профиля

* + 1. **Макет просмотра вакансий**

Для просмотра вакансий используется макет интерфейса, приведенный на рисунке 5.

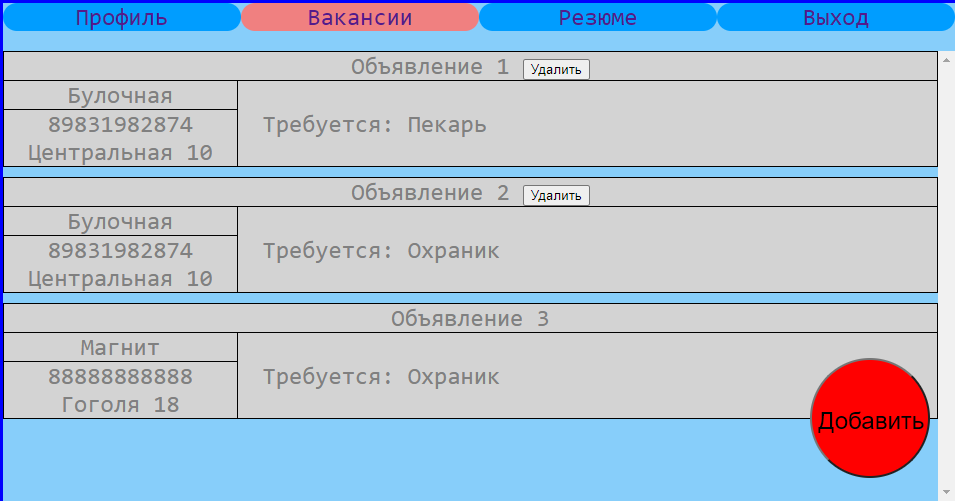


Рисунок 5 - Окно вакансий

* + 1. **Макет добавления вакансий**

Для добавления вакансий используется макет интерфейса, приведенный на рисунке 6.

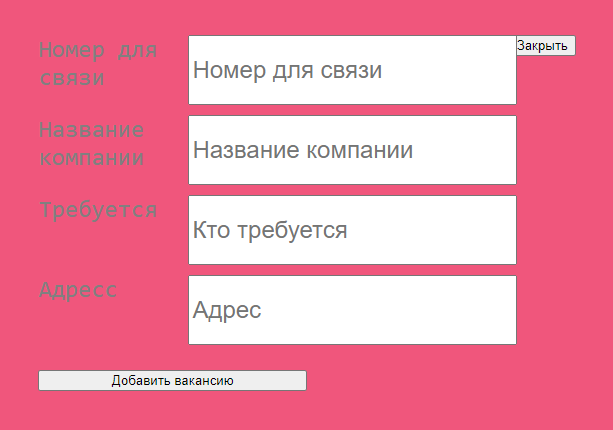


Рисунок 6 - Окно добавления вакансий

* + 1. **Макет просмотра резюме**

Для просмотра резюме используется макет интерфейса, приведенный на рисунке 7.

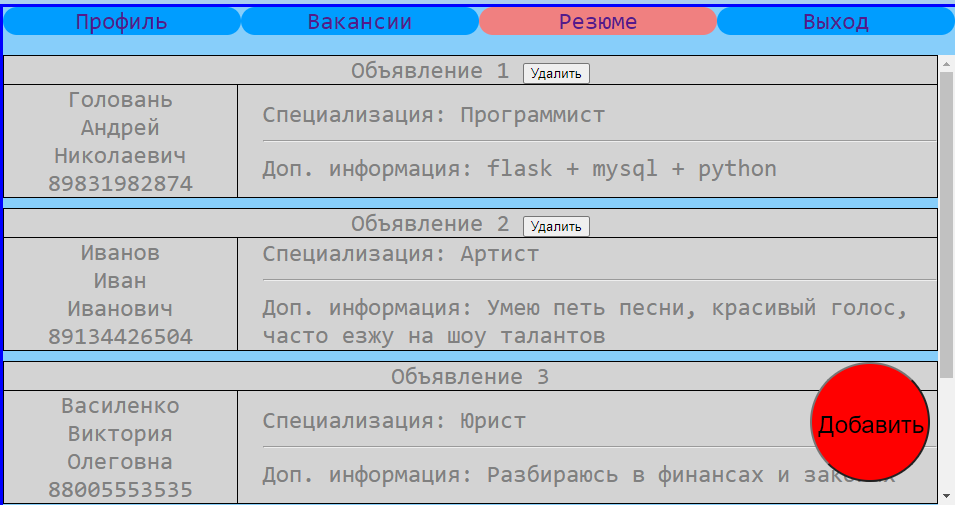


Рисунок 7 - Окно резюме

* + 1. **Макет добавления резюме**

Для добавления резюме используется макет интерфейса, приведенный на рисунке 8.

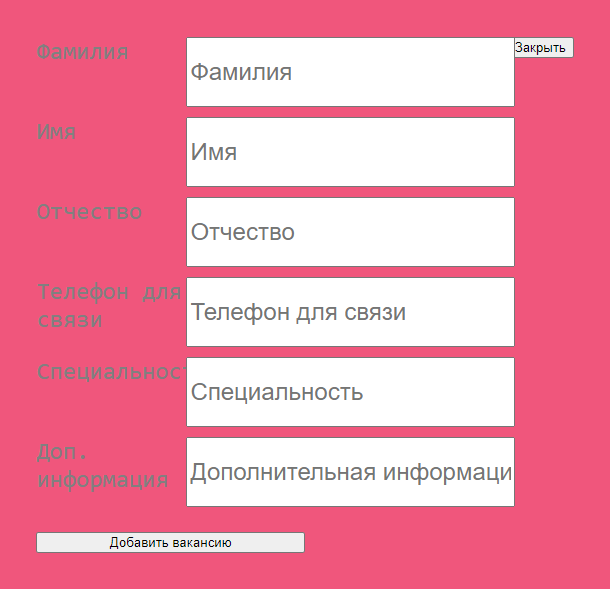


Рисунок 8 - Окно добавления резюме

2. **Описание формата данных**

Программа хранит информацию в БД sqlite3.

* 1. 1. **Информация о пользователях**

Программа хранит информацию о пользователях в таблице profiles.

В таблице 1 приведено описание типов полей, в таблицe 2 - пример данных**.**

Таблица 1 - Описание типов полей таблицы profiles

|  |  |
| --- | --- |
| Поле | тип |
| phone | TEXT |
| first\_name | TEXT |
| last\_name | TEXT |
| patronymic | TEXT |
| password | TEXT |

Таблица 2 - Пример содержимого таблицы profiles

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| phone | first\_name | last\_name | patronymic | password |
| 89831982874 | Головань | Андрей | Николаевич | 2301 |
| 89134426504 | Серженко | Петр | Петрович | 777333 |

2. 3. 2. **Информация о вакансиях**

Программа хранит информацию о вакансиях в таблице vacancies.

В таблице 3 приведено описание типов полей, в таблицe 4 - пример данных.

Таблица 3 - Описание типов полей таблицы vacancies

|  |  |
| --- | --- |
| Поле | Тип |
| hash | TEXT |
| creator | TEXT |
| phone | TEXT |
| company | TEXT |
| activity | TEXT |
| address | TEXT |

Таблица 4 - Пример содержимого таблицы vacancies

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| hash | creator | phone | company | activity | address |
| 269346 | 89831982874 | 89831982874 | Булочная | Пекарь | Центральная 10 |
| 662372 | 89134426504 | 83917463452 | IT-вышка | Программист | Гоголя 17 |

2. 3. 3. **Информация о резюме**

Программа хранит информацию о резюме в таблице summaries.

В таблице 5 приведено описание типов полей, в таблицe 6 - пример данных.

Таблица 5 - Описание типов полей таблицы summaries

|  |  |
| --- | --- |
| Поле | Тип |
| hash | TEXT |
| creator | TEXT |
| first\_name | TEXT |
| last\_name | TEXT |
| patronymic | TEXT |
| phone | TEXT |
| speciality | TEXT |
| diff | TEXT |

Таблица 6 - Пример содержимого таблицы summaries

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| hash | creator | first\_name | last\_name | patronymic | phone | speciality | diff |
| 654753 | 89831982874 | Головань | Андрей | Николаевич | 89831982874 | Прогер | - |
| 126357 | 89134426504 | Серженко | Петр | Петрович | 89134457689 | Уборщик | - |

1. **Описание поведения программы**

Схема поведения интерфейса программы представлена на рисунке 9.

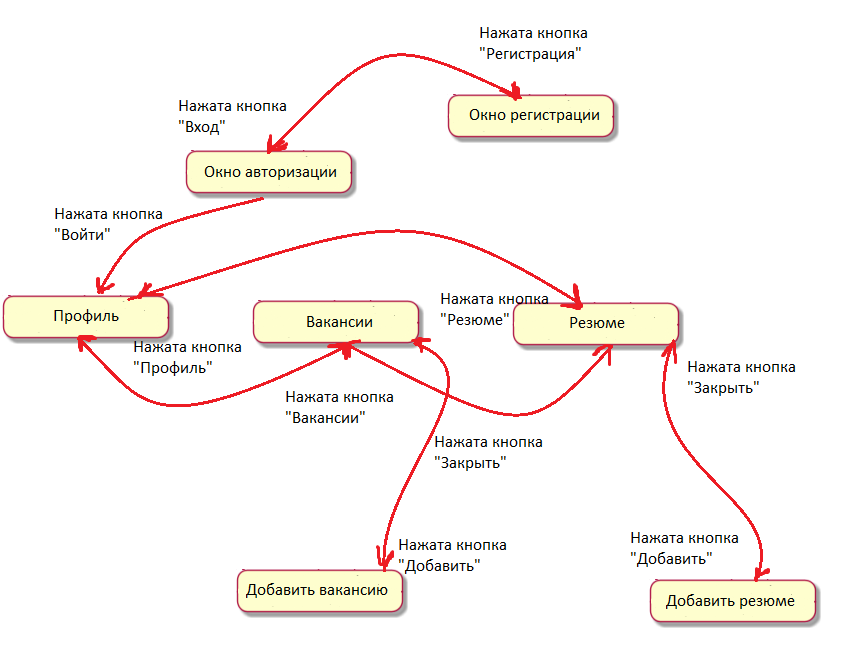


Рисунок 9 - Схема поведения интерфейса программы

* 1. 1. **Вход в систему**

Основная последовательность:

1. При запуске сайта открывается окно авторизации.
2. Пользователь вводит данные учетной записи.
3. При нажатии кнопки “войти” открывается “Профиль”.

Альтернативная последовательность (если были введены некорректные данные):

1. При запуске сайта открывается окно авторизации.
2. Пользователь вводит данные учетной записи.
3. При нажатии кнопки “войти” появляется уведомление о некорректных данных учетной записи.
   * 1. **Регистрация**

Основная последовательность:

1. При запуске сайта открывается окно регистрации.
2. Пользователь вводит данные для регистрации.
3. При нажатии кнопки “Зарегистрироваться” открывается “Окно авторизации”.

Постусловие: информация о пользователе добавляется в таблицу profiles.

Альтернативная последовательность (если были введены некорректные данные):

1. При запуске сайта открывается окно регистрации.
2. Пользователь вводит данные для регистрации.
3. При нажатии кнопки “Зарегистрироваться” появляется уведомление о том, что этот пользователь уже зарегистрирован.
   * 1. **Добавление вакансии**

Основная последовательность:

1. При запуске сайта открывается окно вакансий.
2. Пользователь нажимает кнопку “Добавить”.
3. Пользователь вводит данные для добавления вакансии.
4. При нажатии кнопки “Добавить вакансию” открывается “окно вакансий”.

Постусловие: информация о вакансии добавляется в таблицу vacancies.

Альтернативная последовательность (если были введены некорректные данные):

1. При запуске сайта открывается окно вакансий.
2. Пользователь нажимает кнопку “Добавить”.
3. Пользователь вводит данные для добавления вакансии.
4. При нажатии кнопки “Добавить вакансию” появляется уведомление о том, что какое-то поле ввода было не введено.
   * 1. **Добавление резюме**

Основная последовательность:

1. При запуске сайта открывается окно резюме.
2. Пользователь нажимает кнопку “Добавить”.
3. Пользователь вводит данные для добавления резюме.
4. При нажатии кнопки “Добавить Резюме” открывается “окно резюме”.

Постусловие: информация о вакансии добавляется в таблицу vacancies.

Альтернативная последовательность (если были введены некорректные данные):

1. При запуске сайта открывается окно резюме.
2. Пользователь нажимает кнопку “Добавить”.
3. Пользователь вводит данные для добавления резюме.
4. При нажатии кнопки “Добавить Резюме” появляется уведомление о том, что какое-то поле ввода было не введено.
   * 1. **Удаление вакансии**

Основная последовательность:

1. При запуске сайта открывается окно вакансий.
2. Пользователь нажимает кнопку “Удалить”, напротив той вакансии, какую хочет удалить.
3. Отобразиться новая кнопка “Подтвердить”.
4. Пользователь нажимает кнопку “Подтвердить”.
5. Обновляется страница вакансий.

Постусловие: информация о вакансии удаляется из таблицы vacancies.

* + 1. **Удаление резюме**

Основная последовательность:

1. При запуске сайта открывается окно резюме.
2. Пользователь нажимает кнопку “Удалить”, напротив того резюме, какое хочет удалить.
3. Отобразиться новая кнопка “Подтвердить”.
4. Пользователь нажимает кнопку “Подтвердить”.
5. Обновляется страница резюме.

Постусловие: информация о резюме удаляется из таблицы summaries.

1. **Разработка спецификаций требований**
   * 1. **Разработка диаграммы пригодности**

На рисунках 10-13 приведены диаграммы пригодности для описанных прецедентов.

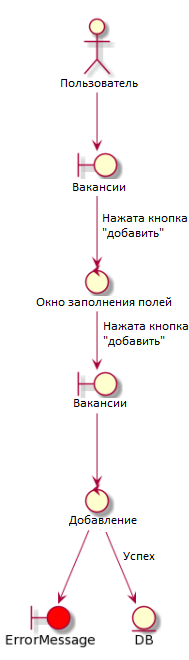


Рисунок 10 — Диаграмма пригодности прецедента добавление вакансии

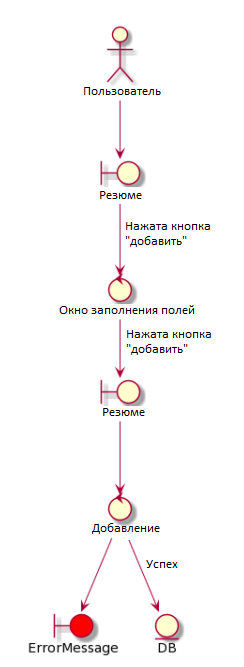


Рисунок 11 — Диаграмма пригодности прецедента добавление резюме

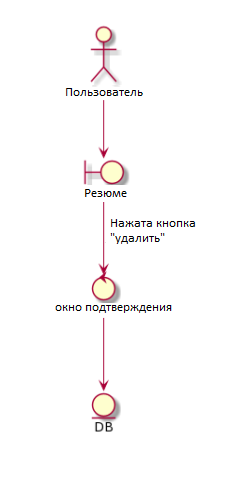


Рисунок 12 — Диаграмма пригодности прецедента удаления резюме

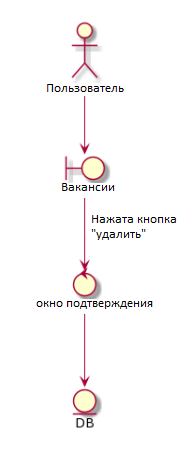


Рисунок 13 — Диаграмма пригодности прецедента удаления вакансии

* + 1. **Разработка диаграммы последовательности**

На рисунках 14-17 приведены диаграммы последовательности.

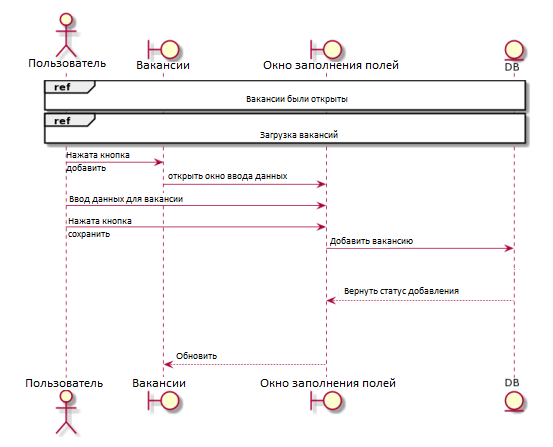


Рисунок 14 — Диаграмма последовательности прецедента добавление вакансии

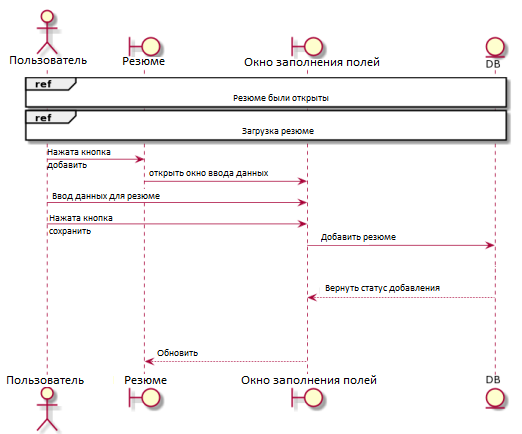


Рисунок 15 — Диаграмма последовательности прецедента добавление резюме

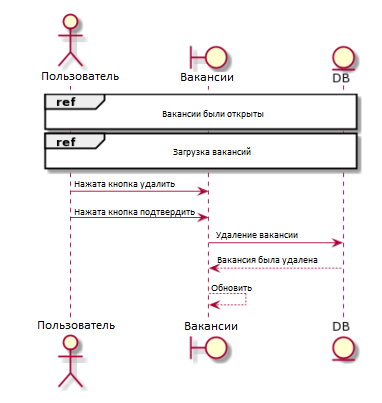


Рисунок 16 — Диаграмма последовательности прецедента удаление вакансии

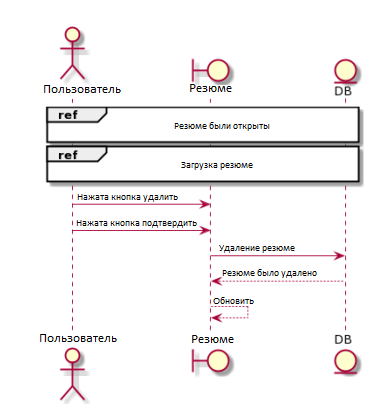


Рисунок 17 — Диаграмма последовательности прецедента удаление резюме

* + 1. **Разработка диаграммы классов уровня проектирования**

Диаграмма классов представлена на рисунке 18

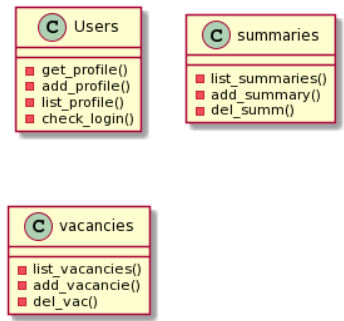


Рисунок 18 — Диаграмма классов

1. **Объектно-ориентированное программирование**
2. **Разработка модульных тестов**

Программа тестировалась с помощью библиотеки pytest. Для этого был

создан дополнительный файл test\_mod.py. Для проверки программы

используется тестирование на основе данных (Data Driven Testing). Тесты проверяют только фасад, т.к. в нем реализованы все функции необходимые для корректного выполнения программы. Ниже будут приведены примеры данных и слотов тестирования некоторых функций.

Данные для тестирования функции проверки номера на корректность представлена на рисунке 19.

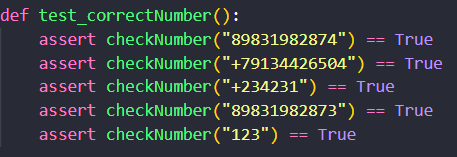


Рисунок 19 - Данные для тестирования функции проверки номера

Данные для тестирования функции удаления вакансии представлены на рисунке 20.

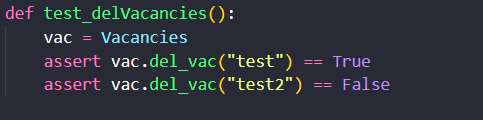


Рисунок 20 - Данные для тестирования функции удаления вакансии

Данные для тестирования функции удаления резюме представлены на рисунке 21.

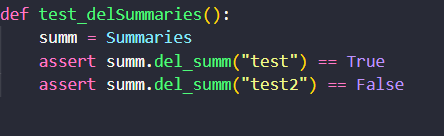


Рисунок 21 - Данные для тестирования функции удаления резюме

Данные для тестирования функции добавления пользователя представлены на рисунке 22.

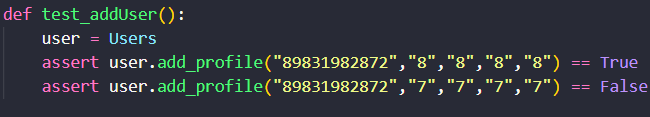


Рисунок 22 - Данные для тестирования функции добавления пользователя

Данные для тестирования функции добавления пользователя представлены на рисунке 23.

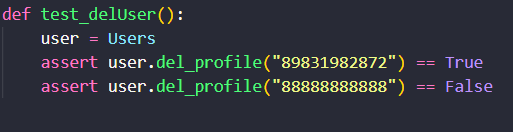


Рисунок 23 - Данные для тестирования функции удаления пользователя

1. **Реализация программы**

Код программы, тесты и документация загружены в удаленный репозиторий на GitHab. Ссылка на репозиторий: <https://github.com/AndPewka/CourseProject/tree/master>

1. **Рефакторинг**

В процессе рефакторинга было сделано несколько изменений:

* Была добавлена функция checkNumber()
* Удалены комментарии

Степень покрытия тестами не составляет 100%, потому что нет никакого

смысла тестировать:

* методы, которые возвращают значения из полей или базы данных;
* веб-сайт.

Тестирование базы данных ограничено логикой программы, т.е. не проверяется корректность данных, находящихся в базе данных.

1. **Документирование**

В работе используется текстовый процессор Microsoft word 2016.

* + 1. **Сборка из исходных данных**

Этот проект был разработан на Windows 10, но его можно будет собрать в macOS и Linux. Все, что вам нужно, это среда разработки Python и установка библиотеки Flask.

1. **Зависимости**
2. Python 3.10.0
3. Flask 2.0.2
4. **Windows**
5. Clone repository

*git clone https://github.com/AndPewka/CourseProject.git*

1. Запустите программу:

*python main.py*

1. Откройте ссылку http://localhost

*start http://localhost*

* + 1. **Инструкция по использованию**

При первом запуске вам необходимо зарегистрироваться и войти. Теперь вы можете добавлять резюме или вакансии.

# ЗАКЛЮЧЕНИЕ

В ходе выполнения проекта были решены следующие поставленные задачи:

* Разработка спецификации требований;
* Проектирование ПО;
* Реализация информационной системы на тему «Страховая компания»;
* Ее документирование.

В процессе работы над проектом, я улучшил знания Python и базы данных

sqlite.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. СТО 4.2–07–2014. Стандарт организации «Общие требования к построению, изложению и оформлению документов учебной деятельности». - Красноярск, 2014. — 60 c.
2. Открытый кроссплатформенный инструмент для построения UMLдиаграмм из текстового описания [Электронный ресурс]. - Режим доступа: PlantUML.
3. А. Н. Чаплыгин. Учимся программировать вместе с Питоном. Учебник. – ревизия 226. – 135 с.
4. Flask (A Python Microframework) [Электронный ресурс] / Armin Ronacher.

Режим доступа: <http://flask.pocoo.org/>.

1. Маркин, А.К. Bianalyzer is a Python package for bicluster analysis over unstructured text data [Электронный ресурс] / GitHub. Режим доступа: <https://github.com/luntos/bianalyzer>.
2. Лутц, М. Программирование на Python, в 2 т. Т. 2, 4-е издание / Марк Лутц. - Санкт-Петербург: Символ-Плюс, 2011. - 992 с.
3. Веб-сервисы в теории и на практике для начинающих / Хабр [Электронный ресурс]. – https://habr.com/ru/post/46374
4. Гринберг, М. Разработка веб-приложений с использованием Flask на языке Python / М. Гринберг. - М.: ДМК, 2014. - 272 c.
5. Гринберг, М. Разработка веб-приложений с использованием Flask на языке Python / М. Гринберг. - М.: ДМК, 2016. - 272 c.
6. Современный учебник JavaScript [Электронный ресурс]: Введение JavaScript – Режим доступа: <https://learn.javascript.ru/intro/>
7. Официальная сайт Git [Электронный ресурс]: - Режим доступа: <https://git-scm.com/>
8. Официальная сайт Python [Электронный ресурс]: About Python Python.org Режим доступа: <https://www.python.org/about/>
9. Официальный сайт Flask [Электрооный ресурс]: Welcome to Flask – Режим доступа: <http://flask.pocoo.org/docs/1.0/>
10. Пирогов Л.Г., HTML и CSS для создания Web-страниц - СПб.:Петербург, 1999. - 332 с.
11. Репин В.А., Языки HTML и CSS. - СПб.: Питер, 2002. - 688 с.