ВВЕДЕНИЕ

В настоящее время в условиях жесткой конкуренции на рынке труда растет с каждым днем, все больше людей стремятся выделиться из толпы и найти свою мечту работы. Одним из важнейших инструментов, которые помогают в этом, является профессиональное резюме. Однако, создание качественного резюме может быть очень сложной задачей, особенно для тех, у кого нет достаточного опыта и знаний.

Информационно-справочная система конструктора резюме – это программный продукт, который помогает упрощать и оптимизировать процесс составления качественного резюме. Она основана на использовании информационных технологий и позволяет собирать, обрабатывать и анализировать информацию об опыте работы, профессиональных навыках и достижениях кандидата.

Целью данного проекта является проектирование и разработка информационно-справочной системы конструктора резюме на основе современных информационных технологий. В работе будут рассмотрены основные требования к ИСС конструктора резюме, методы и технологии проектирования, а также способы реализации и интеграции системы в существующую инфраструктуру.

Система конструктора резюме предоставляет пользователям удобный и интуитивно понятный интерфейс для заполнения информации о своем опыте работы, образовании, навыках и достижениях. Она также предлагает пользователю различные шаблоны и рекомендации по оформлению и структуре резюме, что помогает сделать документ более привлекательным для работодателей.

В рамках данного проекта будет рассмотрена информационно-справочная система конструктора резюме, которая позволит людям создавать высококачественные резюме, способствующие привлечению внимания работодателей на рынке труда. В проекте будут рассмотрены основные функциональные требования к системе, описаны ее основные модули и функции, а также проведен анализ и оценка эффективности системы.

Список тест-сьютов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Автор | Приоритет | Заголовок | Список тест-кейсов |
| 1 | user | 2 | Модуль авторизации | 1. Авторизация пользователя |
| 2 | user | 1 | Модуль создания резюме | 1. Создание резюме<br>2. Редактирование резюме |
| 3 | admin | 1 | Модуль администрирования | 1. Управление шаблонами<br>2. Регистрация пользователя<br>3. Редактирование профилей пользователей |

Список тест-кейсов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Описание (Тип) | Предусловия | Шаги | Ожидаемый результат |
| 1 | Авторизация пользователя (позитивный) | Пользователь находится на странице входа | 1. Ввести логин и пароль<br>2. Нажать кнопку «Войти» | Переход в личный кабинет |
| 2 | Создание резюме (позитивный) | Пользователь авторизован и находится в своем профиле | 1. Выбрать создание нового резюме<br>2. Заполнить необходимые поля | Создание и сохранение нового резюме |
| 3 | Редактирование резюме (позитивный) | Пользователь выбрал существующее резюме | 1. Внести изменения в поля<br>2. Сохранить изменения | Резюме обновлено и сохранено |
| 4 | Управление шаблонами (только для администраторов) | Администратор находится в панели управления | 1. Добавить новый шаблон<br>2. Редактировать существующий шаблон | Шаблон добавлен или изменен |
| 5 | Регистрация пользователя (позитивный) | Администратор находится в панели управления | 1. Заполнить форму регистрации нового пользователя<br>2. Подтвердить регистрацию | Новый пользователь создан и может войти в систему |
| 6 | Редактирование профилей пользователей (позитивный) | Администратор выбрал профиль пользователя для изменения | 1. Внести изменения в профиль<br>2. Сохранить изменения | Профиль пользователя обновлен |

**Описание эквивалентных классов**

* Класс "Авторизация и регистрация" - включает в себя тест-кейсы "Авторизация пользователя (позитивный)" и "Регистрация пользователя (позитивный)". Этот класс проверяет возможность входа в систему для существующих пользователей и регистрацию новых пользователей, включая функциональность, доступную только администраторам.
* Класс "Управление резюме" - охватывает тест-кейсы "Создание резюме (позитивный)" и "Редактирование резюме (позитивный)". Здесь проверяется функциональность системы, связанная с созданием и обновлением резюме пользователями.
* Класс "Администрирование" - включает тест-кейсы "Управление шаблонами (только для администраторов)" и "Редактирование профилей пользователей (позитивный)". В этом классе тестируется возможность администратора управлять шаблонами резюме и редактировать профили пользователей.

Расчет количества тестов

* Для класса "Авторизация и регистрация" требуется провести минимум 2 теста.
* Для класса "Управление резюме" необходимо выполнить 2 теста.
* Для класса "Администрирование" предполагается минимум 2 теста.

Итого, для начального тестирования функционала информационно-справочной системы конструктора резюме требуется минимум 6 тестов.

**Ручное тестирование**

ID: 1

- Описание теста: Авторизация (позитивный)

- Предусловия: Пользователь находится на странице входа в личный кабинет

- Шаги:

1. Ввести логин и пароль в соответствующие поля

2. Нажать кнопку «Войти»

- Ожидаемый результат: Пользователь перенаправляется на страницу личного кабинета

ID: 2

- Описание теста: Регистрация нового пользователя (только для администраторов, позитивный)

- Предусловия: Администратор находится в панели управления

- Шаги:

1. Заполнить данные по новому пользователю

2. Выбрать роль нового пользователя (admin/user)

3. Нажать кнопку «Зарегистрировать»

- Ожидаемый результат: В базе данных создается новый пользователь с соответствующими правами

ID: 3

- Описание теста: Создание резюме (позитивный)

- Предусловия: Пользователь авторизован и находится в своем профиле

- Шаги:

1. Нажать кнопку «Создать новое резюме»

2. Заполнить необходимые поля

3. Нажать кнопку «Сохранить»

- Ожидаемый результат: Создание и сохранение нового резюме

ID: 4

- Описание теста: Редактирование резюме (позитивный)

- Предусловия: Пользователь выбрал существующее резюме

- Шаги:

1. Внести изменения в поля резюме

2. Нажать кнопку «Сохранить изменения»

- Ожидаемый результат: Резюме обновлено и сохранено

ID: 5

- Описание теста: Управление шаблонами (только для администраторов, позитивный)

- Предусловия: Администратор находится в панели управления

- Шаги:

1. Нажать кнопку «Добавить новый шаблон»

2. Заполнить необходимые поля

3. Нажать кнопку «Сохранить»

- Ожидаемый результат: Шаблон добавлен или изменен

ID: 6

- Описание теста: Редактирование профилей пользователей (позитивный)

- Предусловия: Администратор выбрал профиль пользователя для изменения

- Шаги:

1. Внести изменения в поля профиля пользователя

2. Нажать кнопку «Сохранить изменения»

- Ожидаемый результат: Профиль пользователя обновлен

**Дата и время проведения теста**

- 23.05.2024 10:00 – ID тест-кейса: 1 – Результат: Успешно

- 23.05.2024 10:10 – ID тест-кейса: 2 – Результат: Успешно

- 23.05.2024 10:20 – ID тест-кейса: 3 – Результат: Успешно

- 23.05.2024 10:30 – ID тест-кейса: 4 – Результат: Успешно

- 23.05.2024 10:40 – ID тест-кейса: 5 – Результат: Успешно

- 23.05.2024 10:50 – ID тест-кейса: 6 – Результат: Успешно

**Список выявленных дефектов**

Дефекты не выявлены

**Отчет по лабораторной работе №6**

**1. Цель работы**

Цель данной лабораторной работы заключается в разработке автотестов для проверки функциональности программной системы, основанной на ранее разработанной тестовой документации. Автотесты должны быть разработаны с учетом свойств сопровождаемости, надежности и структурированности, а также эмулировать поведение пользователя при взаимодействии с элементами пользовательского интерфейса.

**2. Описание реализованных автотестов**

Инструменты и подходы:

- Инструменты: Для реализации автотестов использовались следующие инструменты:

- Selenium WebDriver для эмуляции поведения пользователя при взаимодействии с веб-интерфейсом.

- JUnit (или PyTest) для написания и выполнения модульных тестов.

- Maven (или pip) для управления зависимостями и сборки проекта.

- Подходы:

- Автотесты были структурированы в виде модульных тестов, организованных по функциональным модулям системы (авторизация, создание резюме, администрирование).

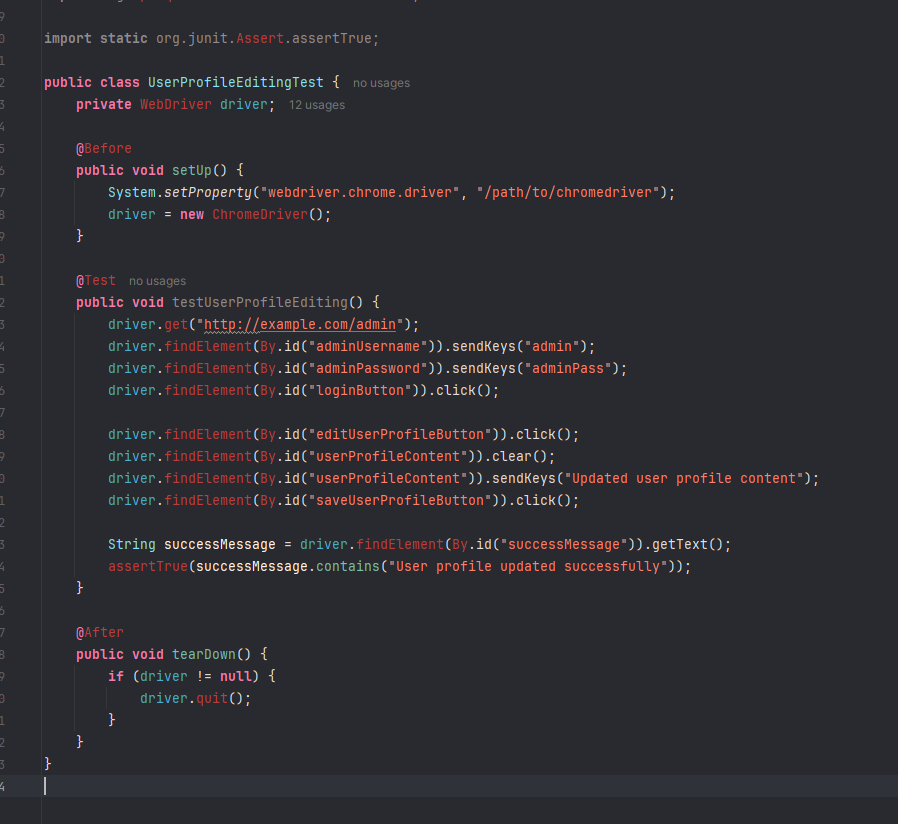
- Каждый автотест включает проверки на корректность выполнения сценариев и контроль результатов выполнения тест-кейсов.

- Автотесты включают позитивные и негативные сценарии для проверки различных аспектов функциональности системы.

**3. Код автотестов**







**4. Отчет о тестировании**

Выполненные тест-кейсы:

- Тест-кейс 1: Авторизация пользователя (позитивный)

- Результат: Успешно

- Тест-кейс 2: Регистрация нового пользователя (только для администраторов, позитивный)

- Результат: Успешно

- Тест-кейс 3: Создание резюме (позитивный)

- Результат: Успешно

- Тест-кейс 4: Редактирование резюме (позитивный)

- Результат: Успешно

- Тест-кейс 5: Управление шаблонами (только для администраторов, позитивный)

- Результат: Успешно

- Тест-кейс 6: Редактирование профилей пользователей (позитивный)

- Результат: Успешно

Выявленные дефекты:

- Дефекты не выявлены

**5. Выводы по работе**

Разработка и реализация автотестов для проверки функциональности программной системы позволили убедиться в корректной работе основных модулей системы. Все тесты прошли успешно, и дефекты не были выявлены, что свидетельствует о высокой степени надежности и сопровождаемости системы. Реализованные автотесты обеспечивают структурированность и позволяют эффективно проверять различные сценарии взаимодействия пользователей с системой.

**6. Список использованных источников**

1. <https://www.selenium.dev/documentation/en/>

2. <https://junit.org/junit5/docs/current/user-guide/>

3. <https://docs.pytest.org/en/stable/>

4. <https://maven.apache.org/guides/index.html>

5. <https://pip.pypa.io/en/stable/user_guide/>

**Отчет по анализу и тестированию документации (Лаб 1 - 2 аттестация)**

**a. Цель работы**

Целью данной работы является анализ и тестирование документации, описывающей требования к проектированию и разработке программного обеспечения. В рамках анализа будут выявлены несоответствия критериям качества документации и предложены рекомендации по их исправлению.

**b. Описание тестируемой документации**

Тестируемая документация включает в себя техническое задание и спецификацию требований к программному обеспечению системы управления резюме. Документация описывает функциональные и нефункциональные требования к системе, включая архитектуру системы, модули и их взаимодействие, а также требования к интерфейсу пользователя.

**c. Описание критериев качества тестируемой документации**

Для оценки качества тестируемой документации использованы следующие критерии:

1. Полнота: Документация должна содержать все необходимые сведения для понимания и реализации требований.

2. Точность: Все требования должны быть однозначно и четко сформулированы.

3. Актуальность: Документация должна быть актуальной и отражать текущее состояние требований.

4. Структурированность: Информация должна быть логически организована и представлена в удобном для восприятия виде.

5. Прослеживаемость: Требования должны быть связаны с соответствующими элементами дизайна и реализации системы.

**d. Описание и обоснование метода тестирования документации**

Метод тестирования документации включает:

- Чтение и анализ: Внимательное чтение документации для выявления несоответствий критериям качества.

- Сравнение с эталоном: Сравнение документации с эталонными примерами и стандартами.

- Экспертное мнение: Привлечение экспертов для оценки качества и полноты документации.

**e. Список несоответствий в документации критериям качества**

1. Полнота

- Пункт: Раздел 2.2 "Функциональные требования"

- Цитата: "Система должна поддерживать создание и редактирование резюме."

- Рекомендация: Добавить подробное описание всех полей резюме и требований к валидации данных.

2. Точность

- Пункт: Раздел 3.1 "Архитектура системы"

- Цитата: "Система будет использовать клиент-серверную архитектуру."

- Рекомендация: Уточнить, какие конкретно технологии будут использоваться для реализации клиентской и серверной частей системы.

3. Актуальность

- Пункт: Раздел 4.3 "Интерфейс пользователя"

- Цитата: "Пользовательский интерфейс должен быть совместим с Internet Explorer."

- Рекомендация: Обновить информацию, указав актуальные браузеры, поддерживаемые системой (например, Google Chrome, Mozilla Firefox).

4. Структурированность

- Пункт: Весь документ

- Цитата: Документация содержит длинные абзацы без разделения на подразделы.

- Рекомендация: Разделить текст на логически обоснованные подразделы с заголовками для лучшего восприятия информации.

5. Прослеживаемость

- Пункт: Раздел 2.4 "Нефункциональные требования"

- Цитата: "Система должна быть масштабируемой."

- Рекомендация: Добавить ссылки на конкретные проектные решения и архитектурные элементы, обеспечивающие масштабируемость.

**f. Выводы по работе**

Анализ показал, что документация содержит ряд несоответствий критериям качества. Основные проблемы связаны с недостаточной полнотой и точностью описания требований, устаревшей информацией и недостаточной структурированностью текста. Внесение предложенных изменений позволит повысить качество документации и облегчить процесс разработки и тестирования системы.

**Задания к лабораторной работе 2 ( 2 аттестация)**

Отчет по анализу и тестированию программного проекта

a. Цель работы

Целью данной работы является проведение анализа программного проекта с точки зрения тестирования, определение применимых видов, типов, методов и уровней тестирования, а также предоставление примеров характерных дефектов для каждого вида тестирования.

b. Описание программного проекта

Программный проект представляет собой систему управления резюме, включающую модули авторизации, создания и редактирования резюме, а также административный модуль для управления шаблонами и профилями пользователей. Основные функциональные компоненты системы включают:

- Авторизация и регистрация пользователей.

- Создание и редактирование резюме.

- Управление шаблонами (доступно только администраторам).

- Редактирование профилей пользователей (доступно только администраторам).

c. Виды, типы, методы и уровни тестирования

Виды тестирования:

1. Функциональное тестирование

- Проверка соответствия системы функциональным требованиям.

- Примеры: тестирование создания и редактирования резюме, авторизации пользователей.

2. Нефункциональное тестирование

- Проверка нефункциональных аспектов, таких как производительность, безопасность и удобство использования.

- Примеры: тестирование производительности при массовом создании резюме, проверка безопасности при авторизации.

Типы тестирования:

1. Модульное тестирование

- Проверка отдельных компонентов или модулей системы.

- Примеры: тестирование функций авторизации и создания резюме.

2. Интеграционное тестирование

- Проверка взаимодействия между различными модулями системы.

- Примеры: тестирование взаимодействия между модулем авторизации и модулем создания резюме.

3. Системное тестирование

- Полная проверка всей системы на соответствие требованиям.

- Примеры: тестирование всей системы управления резюме как единого целого.

4. Приемочное тестирование

- Проверка системы на соответствие требованиям заказчика и готовности к эксплуатации.

- Примеры: тестирование системы по сценариям, предоставленным заказчиком.

Методы тестирования:

1. Динамическое тестирование

- Тестирование путем выполнения программы.

- Примеры: функциональное и нагрузочное тестирование.

2. Статическое тестирование

- Тестирование без выполнения программы.

- Примеры: ревью кода, анализ требований.

Уровни тестирования:

1. Низкоуровневое тестирование

- Модульное тестирование отдельных функций и методов.

- Примеры: тестирование функций авторизации.

2. Среднеуровневое тестирование

- Интеграционное тестирование взаимодействия модулей.

- Примеры: тестирование взаимодействия модуля авторизации и создания резюме.

3. Высокоуровневое тестирование

- Системное и приемочное тестирование всей системы.

- Примеры: тестирование всей системы управления резюме.

d. Примеры дефектов, характерных для каждого вида тестирования

1. Функциональное тестирование

- Дефект: Невозможность сохранить резюме при заполнении всех обязательных полей.

- Дефект: Неправильное сообщение об ошибке при вводе неправильного пароля.

2. Нефункциональное тестирование

- Дефект: Медленная работа системы при создании большого количества резюме одновременно.

- Дефект: Уязвимость системы к SQL-инъекциям при авторизации.

3. Модульное тестирование

- Дефект: Некорректная работа функции проверки пароля.

- Дефект: Ошибка при добавлении новых полей в резюме.

4. Интеграционное тестирование

- Дефект: Неправильная передача данных между модулем авторизации и модулем создания резюме.

- Дефект: Потеря данных при сохранении изменений в профиле пользователя.

5. Системное тестирование

- Дефект: Невозможность выполнить полную регистрацию нового пользователя.

- Дефект: Ошибка при сохранении шаблонов резюме администратором.

6. Приемочное тестирование

- Дефект: Несоответствие системы требованиям заказчика по функционалу создания резюме.

- Дефект: Некорректное отображение данных в интерфейсе пользователя.

e. Выводы по работе

Анализ программного проекта с точки зрения тестирования показал важность применения различных видов, типов, методов и уровней тестирования для обеспечения качества системы. Функциональное, нефункциональное, модульное, интеграционное, системное и приемочное тестирование помогают выявить широкий спектр дефектов на разных этапах разработки. Регулярное проведение этих тестирований способствует созданию надежного и соответствующего требованиям программного обеспечения.

f. Список использованных источников

1. "Software Testing: Principles and Practices" by Srinivasan Desikan, Gopalaswamy Ramesh

2. "Foundations of Software Testing ISTQB Certification" by Dorothy Graham, Rex Black, Erik Van Veenendaal

3. IEEE Std 829-2008 - IEEE Standard for Software and System Test Documentation

4. "Effective Software Testing: 50 Specific Ways to Improve Your Testing" by Elfriede Dustin

ЗАКЛЮЧЕНИЕ

В данной курсовой работе была разработана информационно-справочная система конструктора резюме с генератором резюме, которая успешно реализована с использованием HTML, реляционной базы данных MySQL и библиотеки Spring Boot. Созданный сервис имеет интуитивно понятный пользовательский интерфейс, что делает его удобным для использования для любых пользователей.

Более того, разработанный сервис имеет большой потенциал для дальнейшей модернизации и расширения. Возможно расширение функционала и усложнение процесса создания резюме. Также можно добавить новые роли для иерархии прав доступа к базе данных и ее изменениям.

В процессе выполнения курсовой работы был получен значительный опыт работы с базами данных, пользовательскими интерфейсами и разметкой веб-страниц. Этот опыт может быть полезен для дальнейшей разработки вебсервисов и приложений, связанных с созданием и хранением информации о пользователе.

Кроме того, при написании данной работы были изучены принципы и практики разработки веб-сервисов с использованием библиотеки Spring Boot и реляционных баз данных, а также применены знания в области HTML и CSS для создания пользовательского интерфейса.

В результате созданной информационно-справочной системы была достигнута цель проекта - создание функционального приложения, позволяющего пользователю хранить, добавлять и изменять данные в базе данных и генерировать резюме.

Полученный опыт и знания будут полезны для дальнейшей работы с вебтехнологиями, а также в различных проектах, связанных с базами данных и разработкой веб-сервисов.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Учебная и научная литература

1. Герберт Шилдт. Java: The Complete Reference, Eleventh Edition. - М.: ООО

«И.Д. Вильямс», 2019. - 1376 с.

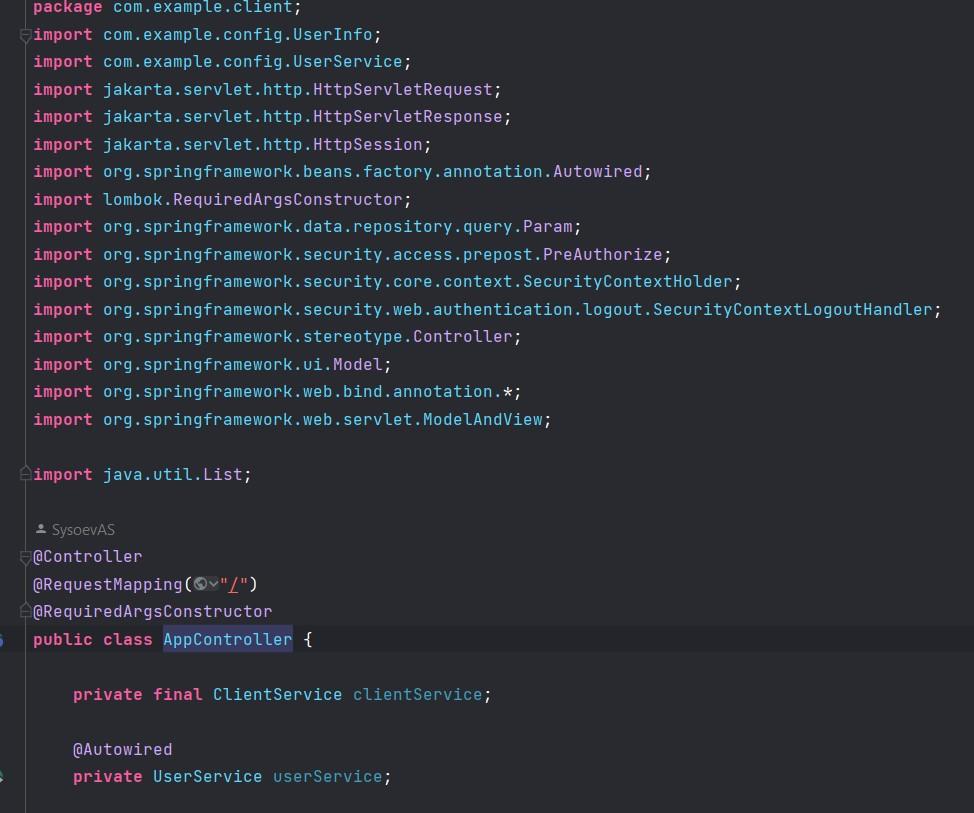
1. Джошуа Блох. Effective Java, Third Edition. - М.: ООО «И.Д. Вильямс», 2019. - 416 с.
2. Роберт Седжвик, Кевин Уэйн. Алгоритмы на Java, части 1-4. - СПб:

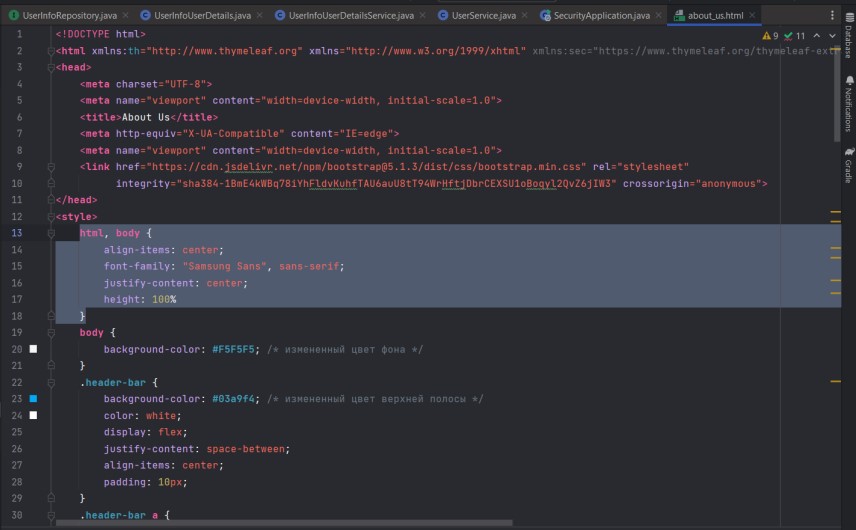
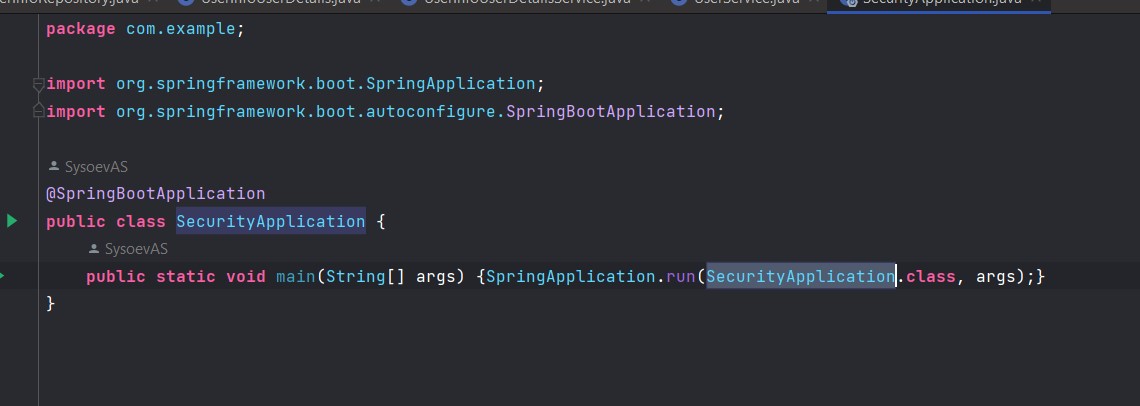
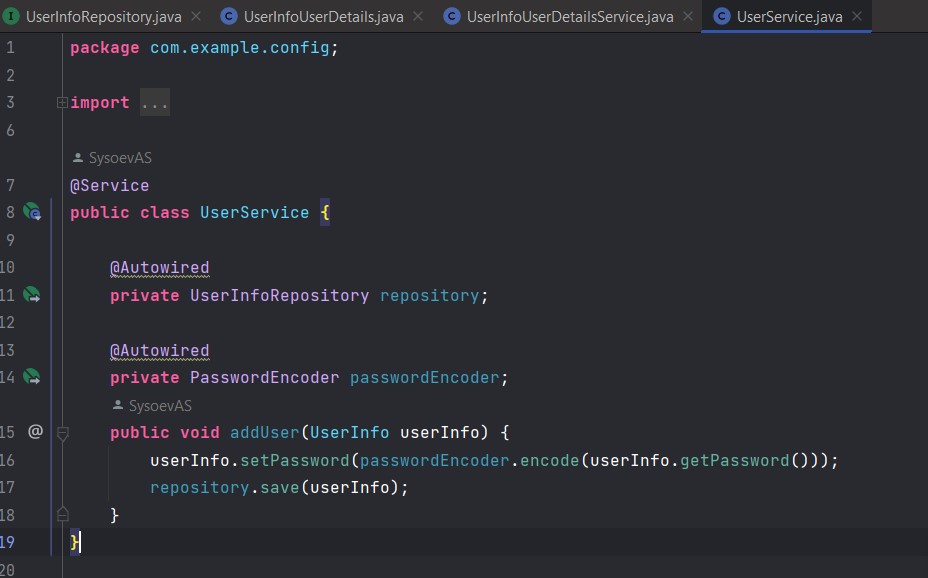
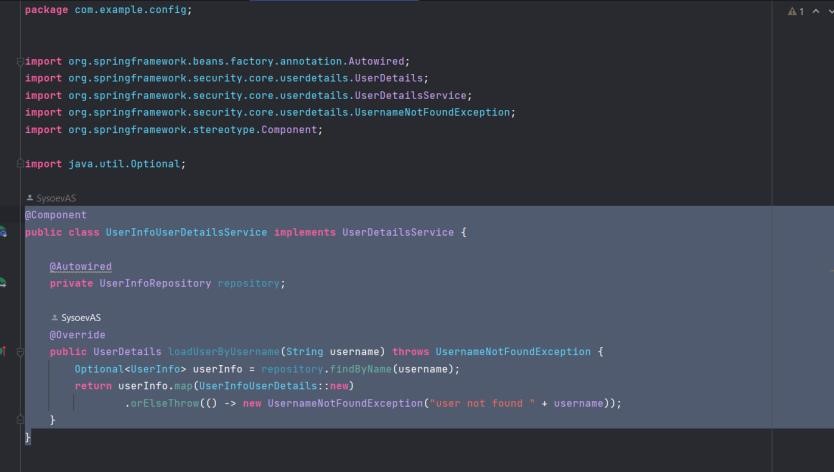
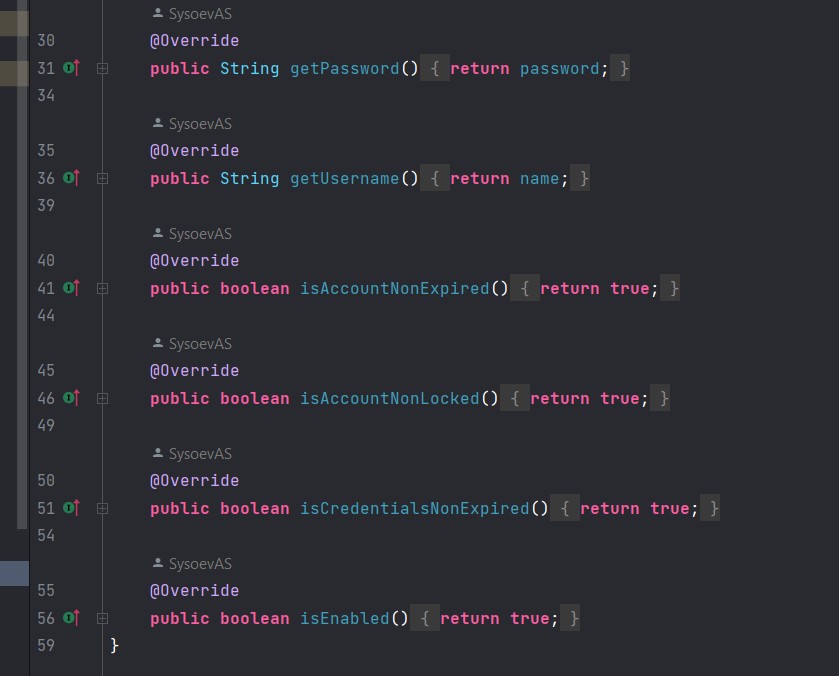
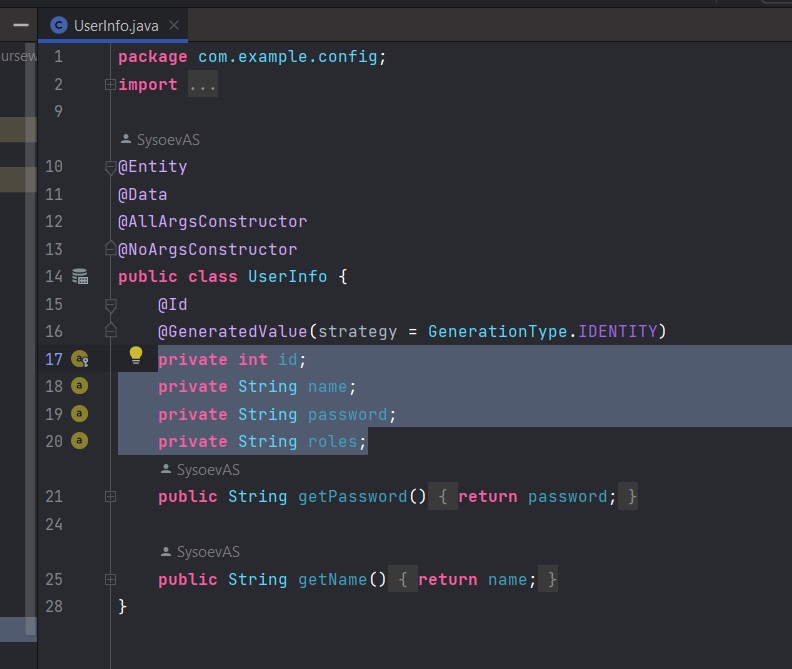
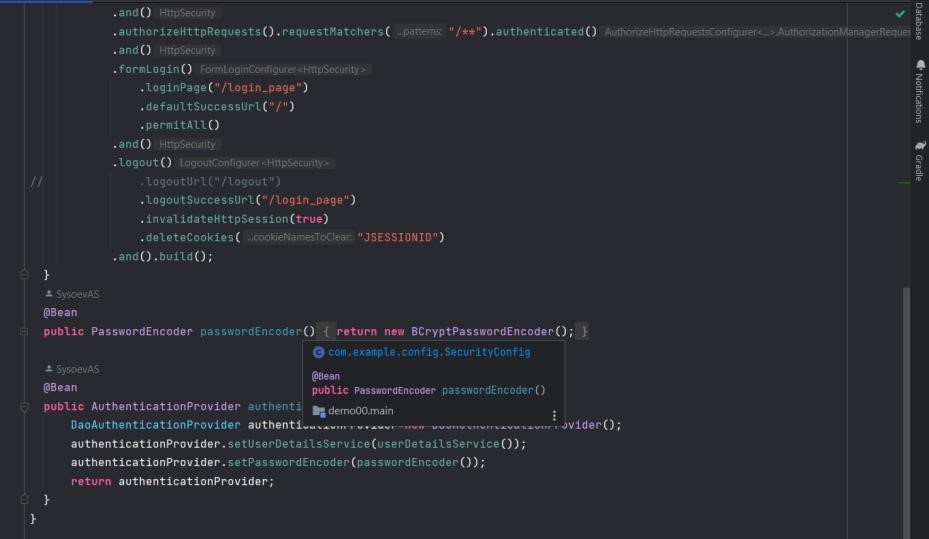
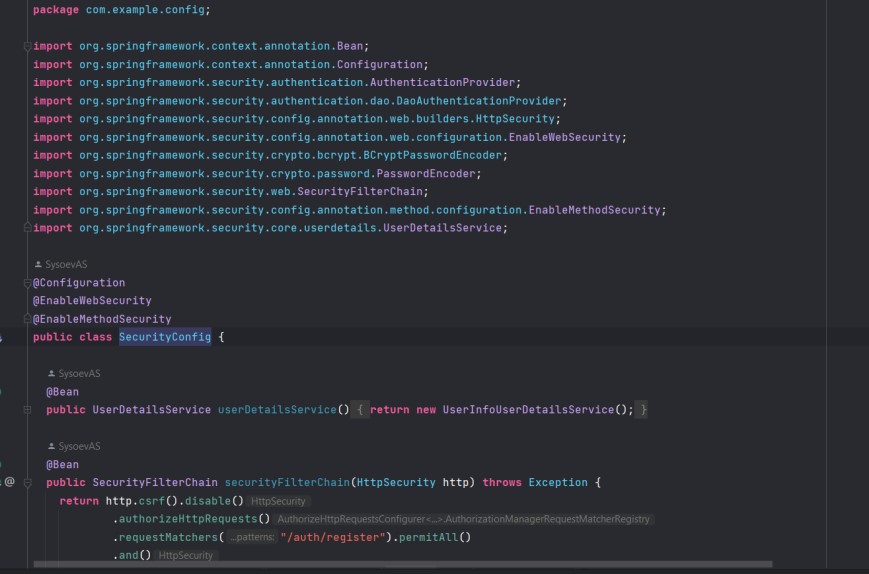
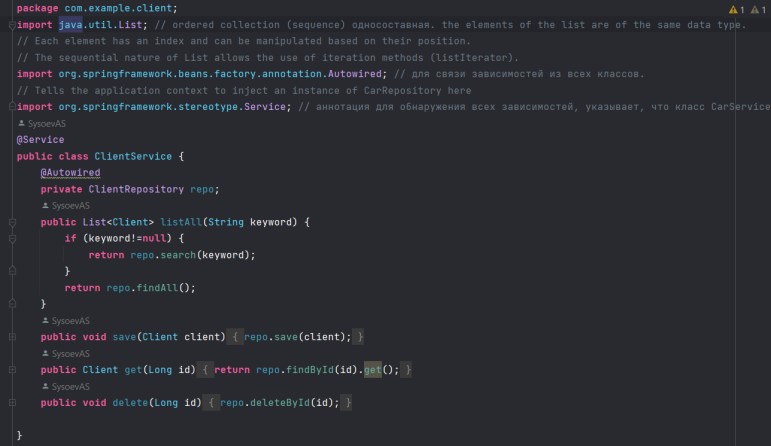
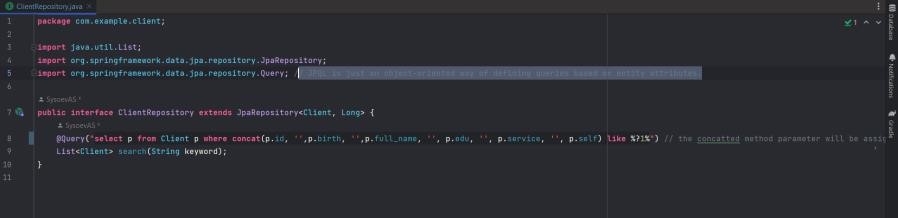
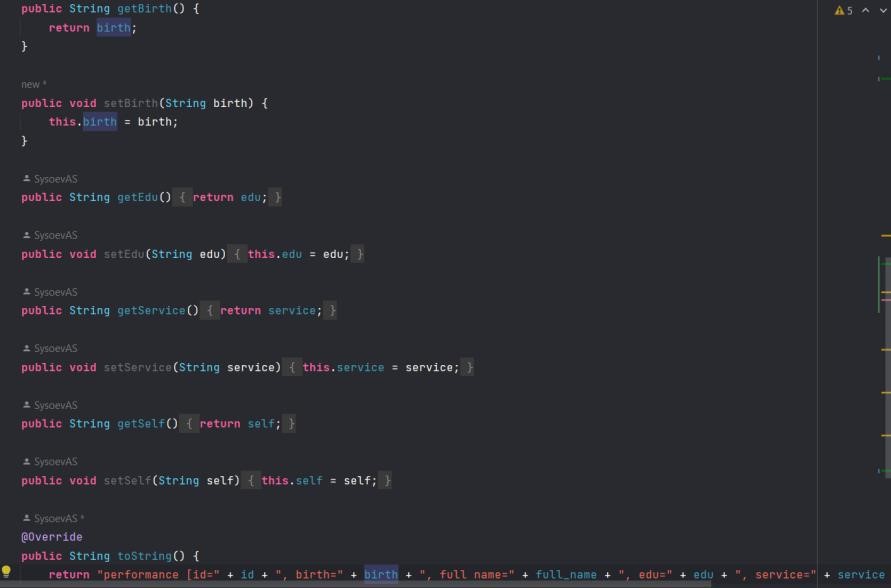
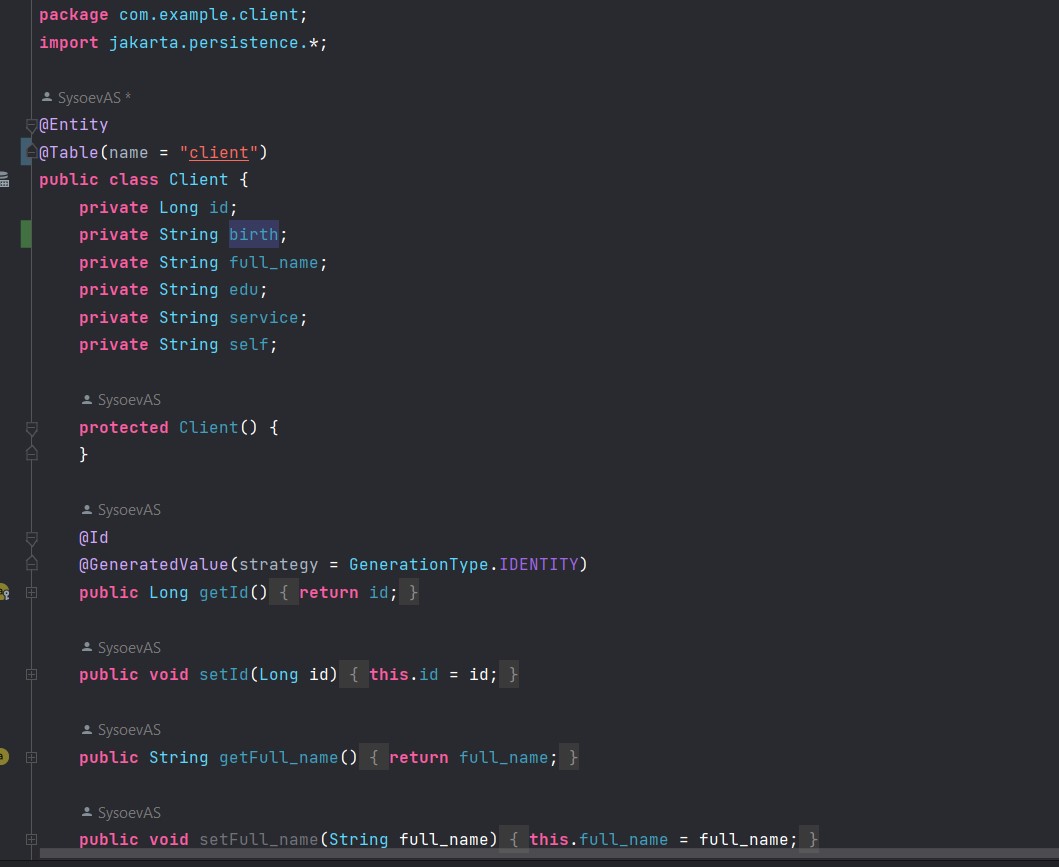
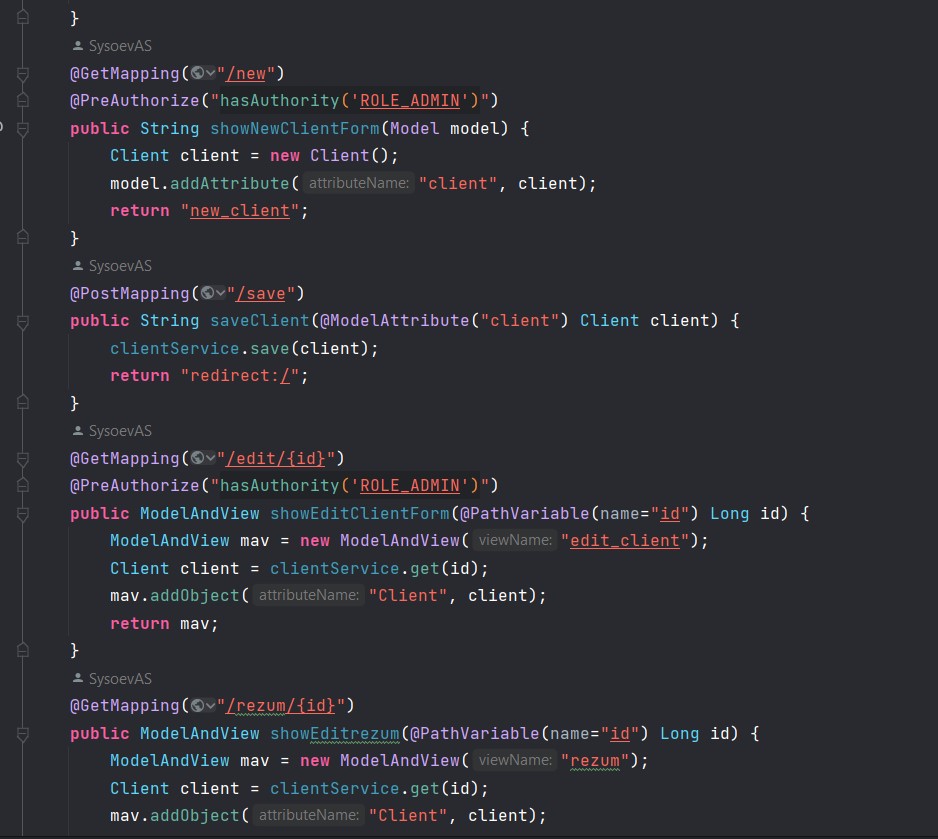
Питер, 2019. - 976 с.

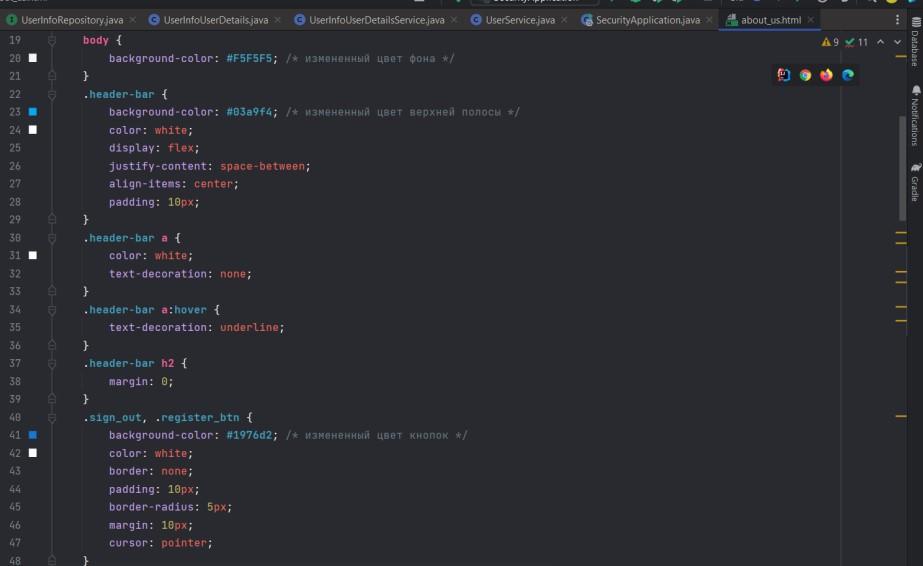
1. Крейг Уоллс. Spring in Action, Fifth Edition. - М.: ДМК Пресс, 2020. - 624 с.
2. Мартин Фаулер. Refactoring: Improving the Design of Existing Code, Second Edition. - СПб: Питер, 2018. - 464 с.
3. Джош Лондон, Натаниэль Т. Шейфер. Java 11 and 12 - New Features: Learn about Project Amber, Records, Local-Variable Syntaxt for Lambda Parameters, Switch Expressions, and More. - США: O'Reilly Media, 2019. - 352 с.
4. Кей С. Хорстманн. Core Java Volume I--Fundamentals, Eleventh Edition. - США: Prentice Hall, 2018. - 928 с.

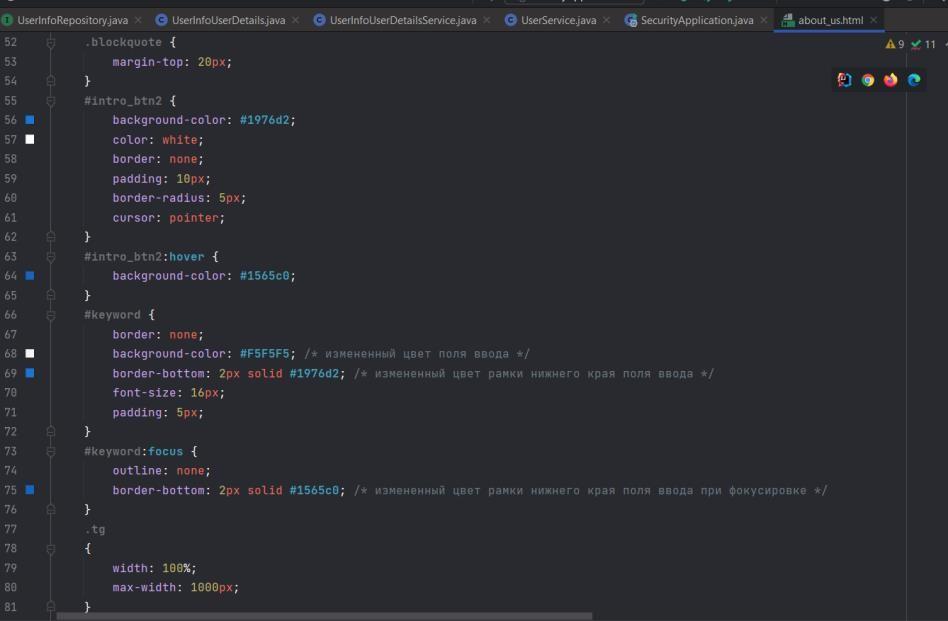
ПРИЛОЖЕНИЯ

Приложение A Исходный код программы









Приложение Б. Исходный код веб-страниц

