МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Кафедра Вычислительной Техники

**Курсовой проект**

**«Конструктор IT-резюме. Java Backend разработка»**

По дисциплине

«Управление проектами»

Группа: АВТ-718

Студент: Мереха Е.В. Преподаватель: Якименко А.А.

Новосибирск, 2021

Оглавление

[Введение 4](#_Toc67416257)

[Управление проектом 5](#_Toc67416258)

[Проблематика проекта 5](#_Toc67416259)

[Актуальность 5](#_Toc67416260)

[Рынок 5](#_Toc67416261)

[Команда проекта 6](#_Toc67416262)

[Предварительные риски 6](#_Toc67416263)

[Конечные потребители 6](#_Toc67416264)

[Разработка плана проекта. Разбиение проекта на задачи. 7](#_Toc67416265)

[Календарное планирование 9](#_Toc67416266)

[Определение типов связей предшествования между задачами 12](#_Toc67416267)

[Бюджет проекта 13](#_Toc67416268)

[Ресурсы проекта 15](#_Toc67416269)

[Определение рисков проекта и их оценка 15](#_Toc67416270)

[Описание проекта. Backend разработка 16](#_Toc67416271)

[Сервер 16](#_Toc67416272)

[Технологии Web-приложения 16](#_Toc67416273)

[Архитектура проекта 17](#_Toc67416274)

[Безопасность 18](#_Toc67416275)

[Результат 20](#_Toc67416276)

[Заключение 22](#_Toc67416277)

[Список литературы 23](#_Toc67416278)

# Введение

В настоящее время остро стоит проблема повышения своей конкурентноспособности специалистами, особенно начинающими, которые пытаются найти себе место на рынке труда в сфере информационных технологий. Огромный наплыв людей, решивших попробовать себя в разработке, тестировании, анализе данных и даже информационной безопасности после прохождения посредственных платных курсов, найденных по рекламе в социальных сетях, создаёт гигантскую нагрузку на Human Resources менеджеров компаний-соискателей.

Последние вынуждены фильтровать потенциальных кандидатов по самым незначительным критериям: наличию высшего образования, наличию опыта коммерческой разработки, структурированности резюме и даже, как можно выяснить из митапов HR-специалистов, по «языку тела» на приложенной к резюме фотографии претендента.

Можно придерживаться разных мнений относительно такого подхода, но куда эффективнее было бы адаптироваться под изменяющиеся условия, принимать новые правила игры и учиться с ними ладить.

Данный курсовой проект ставит своей целью создание онлайн-конструктора резюме, оптимального для конкретной области – специалистов в области информационных технологий, которое позволит составить эффективное, полное, приятное на вид и запоминающееся резюме для будущего работодателя.

# Управление проектом

### Проблематика проекта

В настоящее время, чтобы найти работу недостаточно просто найти объявление в газете. Необходимо представить себя с помощью грамотно оформленного резюме, что несколько сложнее, когда по профессии ты программист, ведь существует огромное количество языков и фреймворков.

### Актуальность

Современный мир не перестает удивлять разнообразием резюме, которые можно найти в интернете, однако не лучшего качества. Не секрет, что правильно оформленное резюме – это большая часть успеха при приеме на работу. Конструктор IT-резюме актуален по той причине, что он позволит создавать грамотно оформленное резюме без особых усилий.

### Рынок

Существует не мало готовых решений, которые создают резюме для всех типов специалистов, однако им не хватает качества в плане IT-резюме. Перепутаны места расположения профессии и имени, неверное оформление языков и изученных фреймворков, лишние фотографии или наоборот – отсутствие. Все эти минусы ведут к тому, что работодатель может просто пропустить такое резюме и обратить внимание на резюме оформленное строже.

### Команда проекта

*Рожков Евгений Александрович* – WEB-дизайнер. Разработчик JS, HTML, CSS.

*Мереха Евгений Владимирович –* Enterprise Java-разработчик. Разработка серверной части проекта, создание API для взаимодействия приложения с БД.

*Рубанский Павел Сергеевич* – Бизнес-аналитик. Анализ рынка, конкурентная разведка, системный анализ.

*Туманов Максим Евгеньевич* – Администратор базы данных, DevOps-инженер.

### Предварительные риски

* Неактуальность проекта на стадии завершения
* Малая конкурентоспособность
* Всего один тип создаваемого резюме

### Конечные потребители

Специалисты в области IT-технологий, которые не знают, как правильно оформить резюме для поиска работы.

Главные недостатки существующих на рынке решений будут нивелированы, так как проект ставит задачей создание именно IT-резюме, которое будет оформлено верно.

В случае успеха и верной реализации проекта, обычный программист, использовав наш сервис, будет уверен в том, что его резюме – максимум, который он может получить.

### Разработка плана проекта. Разбиение проекта на задачи.

Формирование команды может быть непрерывным процессом, особенно на длительных и массивных проектах, в ходе реализации которых сотрудники успевают повысить свою квалификацию, а также открываются новые вакансии, в том числе на аутсорс. В случае нашего проекта, формирование команды завершено и может быть подвергнуто изменениям только в случае некого форс-мажора.

Проведённый анализ рынка позволил определиться со стеком используемых технологий, а также определить недостатки других проектов данной ниши.

Реализация проекта была подразделена на основные задачи, в соответсвии с которыми, а также исходя из имеющихся навыков, участники команды распределились по ролям исполнения.

После распределения ролей между участниками команды, а также уточнения классов задач, решаемых каждой ролью и командой в целом, участники самостоятельно декомпозировали свою задачу внутри проекта на несколько структурных подзадач, что позволит установить зависимости между ролями и более наглядно следить за ходом реализации проекта.

В ходе реализации проекта DevOps-инженер обеспечивает интеграцию инструментов разработки, а также обратной связи всех исполнителей проекта, что позволяет бизнес-аналитику производить оценку хода выполнения задач и их сроки, после чего производить системный анализ – формирование новых требований к участникам разработки.

1. Рожков Евгений Александрович – WEB-разработчик

* *Вёрстка веб-страниц и их элементов.*
* *Разработка скриптов, исполняемых на стороне сервера.*
* *Разработка AJAX.*
* *Дизайн элементов веб-страниц.*

1. Мереха Евгений Владимирович – Enterprise Java-разработчик.

* *Реализация методов обработки пользовательских запросов.*
* *Реализация API взаимодействия с базой данных.*
* *Реализация основного приложения.*

1. Рубанский Павел Сергеевич – Бизнес-аналитик.

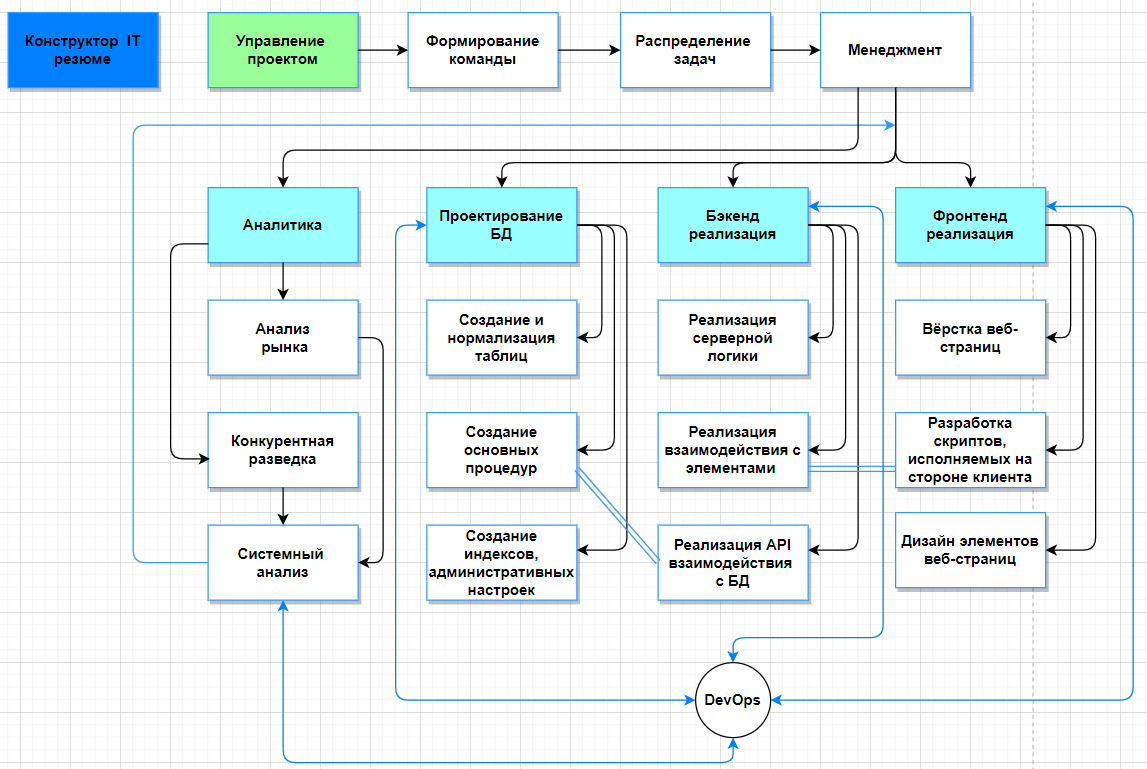
* *Анализ рынка.*
* *Конкурентная разведка.*
* *Системный анализ.*

1. Туманов Максим Евгеньевич – Администратор базы данных, DevOps-инженер.

* *Выбор СУБД.*
* *Составление таблиц и их нормализация.*
* *Администрирование БД.*
* *DevOps-обеспечение.*

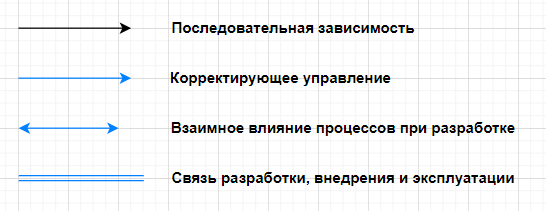
Под тестирование и эксплуатацию не отведено ролей за нехваткой персонала. Эти задачи предполагается отвести будущим пользователям веб-приложения.

Менеджмент проекта производится совместными усилиями всех участников команды в ходе взаимодействия и обсуждения хода реализации проекта.

**

*Рисунок 1. Структура реализации проекта.*

На диаграмме структуры реализации проекта (рис. 1) выделены следующие связи между отдельными задачами и непрерывными процессами проекта:

**

*Рисунок 2. Легенда связей на структурной диаграмме (рис. 1).*

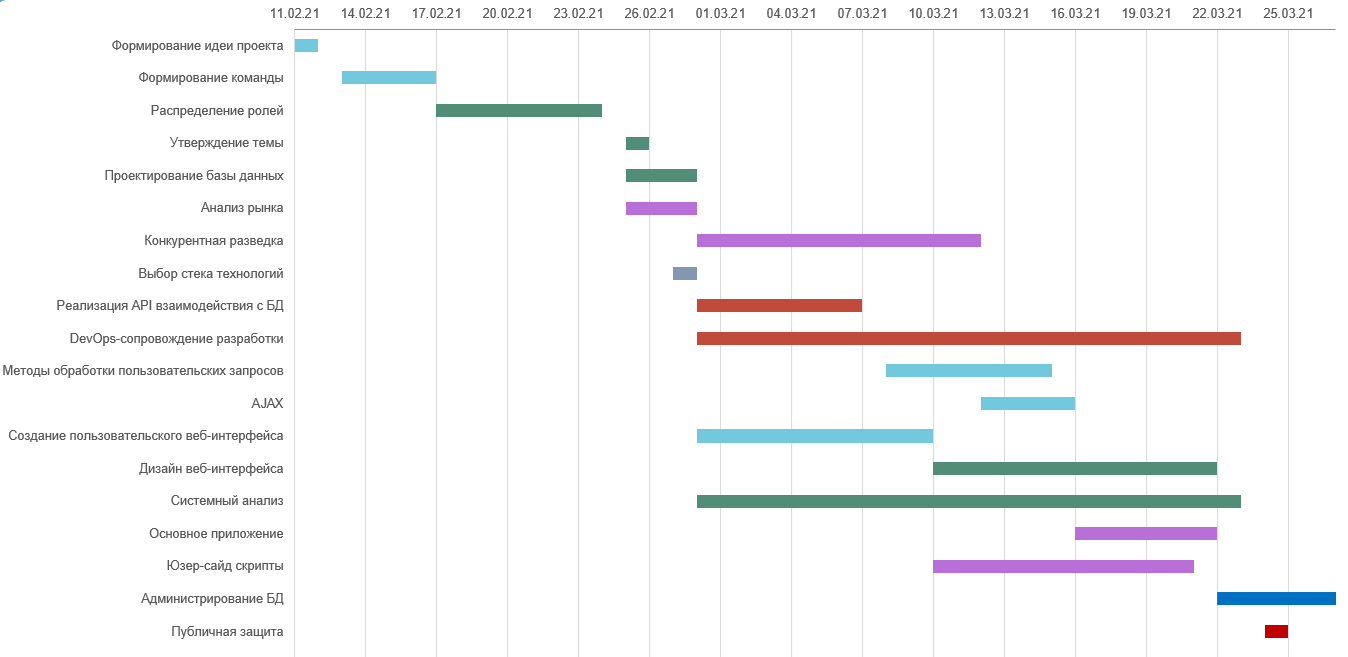
### 

### Календарное планирование

Были выделены следующие основные этапы реализации проекта:

1. **Замысел.** Рождение идеи проекта (05.02.2021-11.02.2021)
2. **Начало.** Формирование команды исполнителей, утверждение темы, распределение ролей (11.02.2020-25.02.2020)
3. **Планирование.** Создание плана проекта (25.02.2020-28.02.2020)
4. **Исполнение.** Выполнение работ по проекту (28.03.2020-23.03.2020)
5. **Завершение.** Срок окончания проекта, день перед презентацией. (23.03.2020)

Выделенные подзадачи реализации проекта, а также процессы управления распределены на временной диаграмме (рис. 3).

**

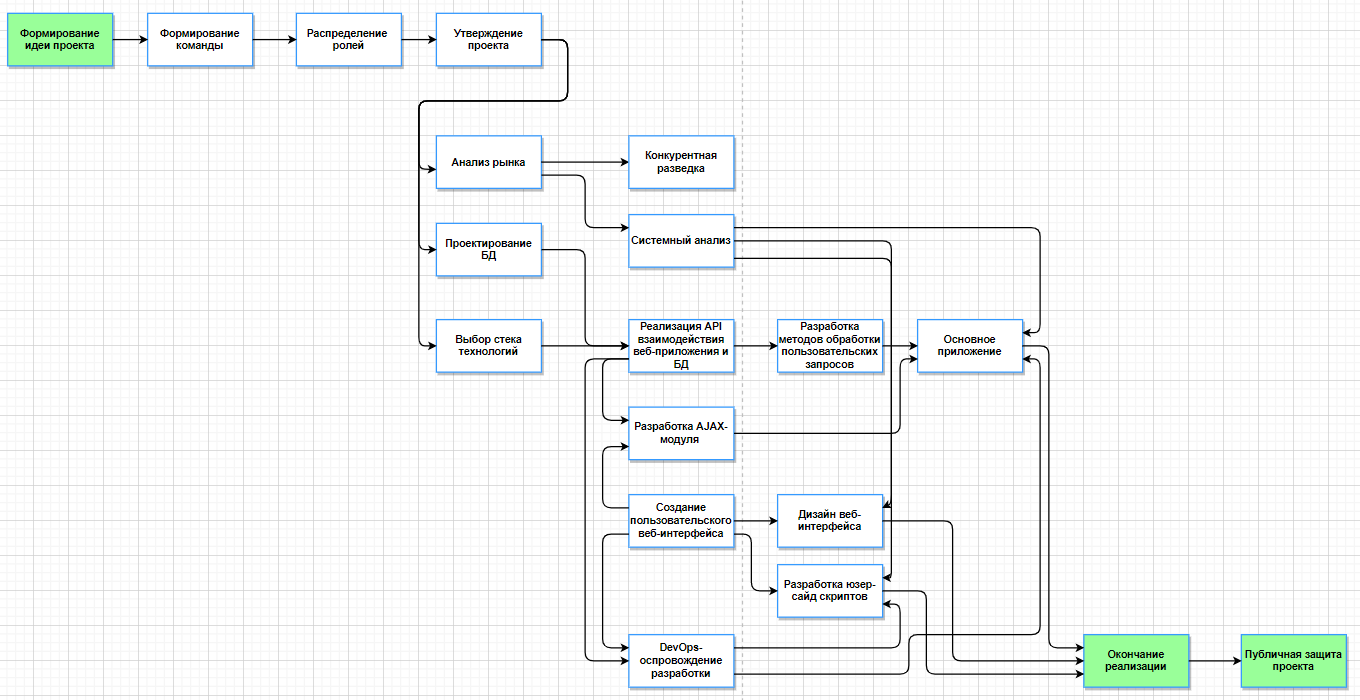
*Рисунок 3. Диаграмма Ганта для реализации проекта «Конструктор IT резюме».*

*Таблица 1. Ориентировочные сроки выполнения подзадач участниками команды.*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Исполнитель** | **Задача** | **Начало** | **Окончание** | **Дней** | **Часов** |  |
| Мереха, Рожков | Формирование идеи проекта | 11.02.2021 | 12.02.2021 | 1 |  |  |
| Команда | Формирование команды | 13.02.2021 | 17.02.2021 | 4 |  |  |
| Команда | Распределение ролей | 17.02.2021 | 24.02.2021 | 7 |  |  |
| Команда | Утверждение темы | 25.02.2021 | 26.02.2021 | 1 |  |  |
| Туманов | Проектирование базы данных | 25.02.2021 | 28.02.2021 | 3 | 24 |  |
| Рубанский | Анализ рынка | 25.02.2021 | 28.02.2021 | 3 | 24 |  |
| Рубанский | Конкурентная разведка | 28.02.2021 | 12.03.2021 | 12 | 96 |  |
| Туманов, Мереха, Рожков | Выбор стека технологий | 27.02.2021 | 28.02.2021 | 1 | 8 |  |
| Мереха | Реализация API взаимодействия с БД | 28.02.2021 | 07.03.2021 | 7 | 56 |  |
| Туманов | DevOps-сопровождение разработки | 28.02.2021 | 23.03.2021 | 23 | 184 |  |
| Мереха | Методы обработки пользовательских запросов | 08.03.2021 | 15.03.2021 | 7 | 56 |  |
| Рожков | AJAX | 12.03.2021 | 16.03.2021 | 4 | 32 |  |
| Рожков | Создание пользовательского веб-интерфейса | 28.02.2021 | 10.03.2021 | 10 | 80 |  |
| Рожков | Дизайн веб-интерфейса | 10.03.2021 | 22.03.2021 | 12 | 96 |  |
| Рубанский | Системный анализ | 28.02.2021 | 23.03.2021 | 23 | 184 |  |
| Мереха | Основное приложение | 16.03.2021 | 22.03.2021 | 6 | 48 |  |
| Рожков | Юзер-сайд скрипты | 10.03.2021 | 21.03.2021 | 11 | 88 |  |
| Туманов | Администрирование БД | 22.03.2021 | 27.03.2021 | 5 | 40 |  |
| Команда | Публичная защита | 24.03.2021 | 25.03.2021 | 1 |  |  |

### 

### Определение типов связей предшествования между задачами

**

*Рисунок 4. Диаграмма предшествования для реализации проекта «Конструктор IT-резюме».*

### Бюджет проекта

Исходя из плана проекта (Табл. 1) было подсчитано ориентировочное количество времени, необходимое для реализации собственной части проекта каждым участником, обговорены индивидуальные размеры почасовой оплаты труда и высчитан зарплатный бюджет при восьмичасовом рабочем дне.

Под тестирование и эксплуатацию не отведено ролей за нехваткой персонала. Эти задачи предполагается отвести будущим пользователям веб-приложения.

Менеджмент проекта производится совместными усилиями всех участников команды в ходе взаимодействия и обсуждения хода реализации проекта.

Для комфортной работы всех участников команды было решено арендовать комнату на 4 рабочих места сроком 1 месяц в коворкинг-центре BusinessLab в связи с высокими оценками на сервисах отзывов и удобных расположением для всех участников команды. Месячный абонемент резидента включает в себя:

* Закреплённое рабочее место с доступом 24/7
* Безлимитный доступ в интернет через WiFi
* МФУ-принтер
* Личный шкафчик
* Доступ в общую кухню.

1. Рожков Евгений Александрович – WEB-разработчик **(450р/час), 304 часа**

* *Вёрстка веб-страниц и их элементов.*
* *Разработка скриптов, исполняемых на стороне сервера.*
* *Разработка AJAX.*
* *Дизайн элементов веб-страниц.*

1. Мереха Евгений Владимирович – Enterprise Java-разработчик. **(750р/час), 168 часов**

* *Реализация методов обработки пользовательских запросов.*
* *Реализация API взаимодействия с базой данных.*
* *Реализация основного приложения.*

1. Рубанский Павел Сергеевич – Бизнес-аналитик. **(120р/час), 304 часа**

* *Анализ рынка.*
* *Конкурентная разведка.*
* *Системный анализ.*

1. Туманов Максим Евгеньевич – Администратор базы данных, DevOps-инженер. **(420р/час), 256 часов**

* *Выбор СУБД.*
* *Составление таблиц и их нормализация.*
* *Администрирование БД.*
* *DevOps-обеспечение.*

*Таблица 2. Статьи расходов*

|  |  |  |
| --- | --- | --- |
| **Наименование** | **Количество** | **Стоимость (руб.)** |
| Расходы на разработку | | |
| Работа бэкенд-программиста | 168ч | 126000р |
| Работа веб-программиста | 304ч | 136800р |
| Работа аналитика | 304ч | 36480р |
| Работа DevOps-инженера | 256ч | 107520р |
| **Итого** | | 406800р |
| Техническая поддержка | | |
| Аренда сервера | 1 месяц | 5000р |
| Аренда почтового сервера | 1 месяц | 4000р |
| Аренда дискового пространства | 1 месяц | 7000р |
| **Итого** | | 16000р |
| Дополнительные траты | | |
| Рабочее место | 4 резидента/1 месяц | 30000р |
| Продвижение продукта | 1 месяц | 120000р |
| **Итого** | | 150000р |
| **Итого всего** | | **572800р** |

### Ресурсы проекта

Исходя из состава команды проекта, плана его реализации и затребованных технических нужд определим имеющиеся трудовые и материальные ресурсы проекта:

*Таблица 3. Ресурсы проекта*

|  |  |  |
| --- | --- | --- |
| **Ресурс** | **Описание** | **Срок** |
| *Трудовые ресурсы* | | |
| Руководство | Децентрализовано | 25.02-25.03 |
| Аналитик | Рубанский П.С. | 25.02-25.03 |
| Разработчики | Мереха Е.В., Туманов М.Е., Рожков Е.А. | 25.02-25.03 |
| Администратор | Туманов М.Е. | 25.02-25.03 |
| Дизайнер | Рожков Е.А. | 25.02-25.03 |
| *Материальные ресурсы* | | |
| Рабочий офис | Комната на 4 человек в коворкинг-центре | 25.02-25.03 с продлением |
| Серверы | Арендованный VDS + почтовый сервер + облачное хранилище | 25.02-25.03 с продлением |

### Определение рисков проекта и их оценка

*Таблица 4. Риски проекта*

|  |  |  |
| --- | --- | --- |
| **Факторы риска** | **Угрозы** | **Мероприятия по снижению риска** |
| *Технические риски* | | |
| Ошибочный выбор стека технологий для реализации проекта | Возникновение сложностей при разработке | Сравнение нескольких стеков технологий, выбор оптимального и обоснование выбора |
| Нехватка вычислительной мощности сервера | Падение быстродействия | Расчет возможного максимального и минимального трафика и подбор подходящей конфигурации сервера |
| *Финансовые риски* | | |
| Задержка разработки проекта | Ежемесячные расходы при задержке проекта приведут к переплате | Расписание проекта, определение сроков этапов разработки проекта, выбор компетентных разработчиков |
| *Организационные риски* | | |
| Недостаточная поддержка со стороны руководства заказчика | Увеличение сроков исполнения работ | Выделение ответственного, контролирующего сроки и качество работ |
| *Форс-мажорные риски* | | |
| Атака на сервер | Утеря БД | Регулярные бэкапы |

# Описание проекта. Backend разработка

## ***Сервер***

Для развертывания приложения в первую очередь решалась задача по выбору Web-сервера. Поскольку приложение будет не высоконагруженным, то выбор пал на сервер Tomcat.

Tomcat — это контейнер сервлетов с открытым исходным кодом, который также выполняет функцию веб-сервера. Веб-приложение рассчитано на взаимодействие с клиентом. Если есть запрос от клиента, он обрабатывается, и пользователю отправляется ответ. Если нет, приложение простаивает.

Tomcat фактически представляет собой Java-приложение, которое заботится об открытии порта для взаимодействия с клиентом, настройке сессий, количестве запросов, длине заголовка и еще многих операциях.

## ***Технологии Web-приложения***

Все приложение будет строиться вокруг технологии Spring Framework. Spring Framework представляет собой просто контейнер внедрения зависимостей, с несколькими удобными слоями (например: доступ к базе данных, прокси, аспектно-ориентированное программирование, RPC, веб-инфраструктура MVC). Это все позволяет быстрее и удобнее создавать Java-приложения.

Для взаимодействия с базой данных использовалось framework Hiberante. Hibernate — это библиотека, которая предназначена для задач объектно-реляционного отображения. Библиотека позволяет разработчику работать с базой данных не напрямую, как это делается с помощью библиотеки JDBC, а с помощью представления таблиц баз данных в виде классов Java.

Чтобы оптимизировать работу с большим объемом информации, используют базы данных. В свою очередь, сами БД тоже постоянно меняются и дополняются новыми данными. Для упрощения процесса администрирования, добавления и редактирования информации, были разработаны специальные системы управления (СУБД). MySQL — это одна из множества СУБД, которую используют для упрощения работы с базами данных. Она включает в себя библиотеку внутреннего сервера, с помощью которой можно использовать MySQL в отдельных программах.

Maven — инструмент для автоматизации сборки проектов. Чем сложнее разрабатываемое ПО и чем больше оно использует сторонних библиотек и ресурсов, тем сложнее будет команда для сборки. Maven разработан для облегчения этой работы. Одна из главных особенностей фреймворка — декларативное описание проекта. Это значит, что разработчику не нужно уделять внимание каждому аспекту сборки — все необходимые параметры настроены по умолчанию. Изменения нужно вносить лишь в том объёме, в котором программист хочет отклониться от стандартных настроек.

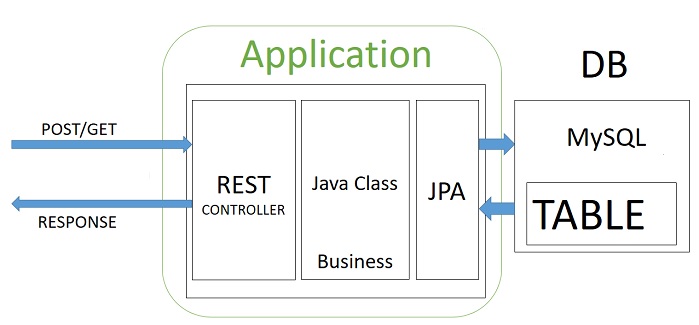
Thymeleaf — современный серверный механизм Java-шаблонов для веб и автономных сред, способный обрабатывать HTML, XML, JavaScript, CSS и даже простой текст. Основной целью Thymeleaf является создание элегантного и удобного способа шаблонизации. Чтобы достичь этого, Thymeleaf основывается на концепции Natural Templates, чтобы внедрить свою логику в файлы шаблонов таким образом, чтобы этот шаблон не влиял на отображение прототипа дизайна. Это улучшает коммуникацию в команде и уменьшает разрыв между дизайнерско-программистскими группами.

Git — распределённая система контроля версий, которая даёт возможность разработчикам отслеживать изменения в файлах и работать над одним проектом совместно с коллегами. Git известен своей скоростью, простым дизайном, поддержкой нелинейной разработки, полной децентрализацией и возможностью эффективно работать с большими проектами.

## ***Архитектура проекта***

Фреймворк Spring MVC обеспечивает архитектуру паттерна Model — View — Controller (Модель — Отображение (Вид) — Контроллер) при помощи слабо связанных готовых компонентов. Паттерн MVC разделяет аспекты приложения (логику ввода, бизнес-логику и логику UI), обеспечивая при этом свободную связь между ними.

* Model (Модель) инкапсулирует (объединяет) данные приложения, в целом они будут состоять из POJO («Старых добрых Java-объектов», или бинов).
* View (Отображение, Вид) отвечает за отображение данных Модели, — как правило, генерируя HTML, которые мы видим в своём браузере.
* Controller (Контроллер) обрабатывает запрос пользователя, создаёт соответствующую Модель и передаёт её для отображения в Вид.

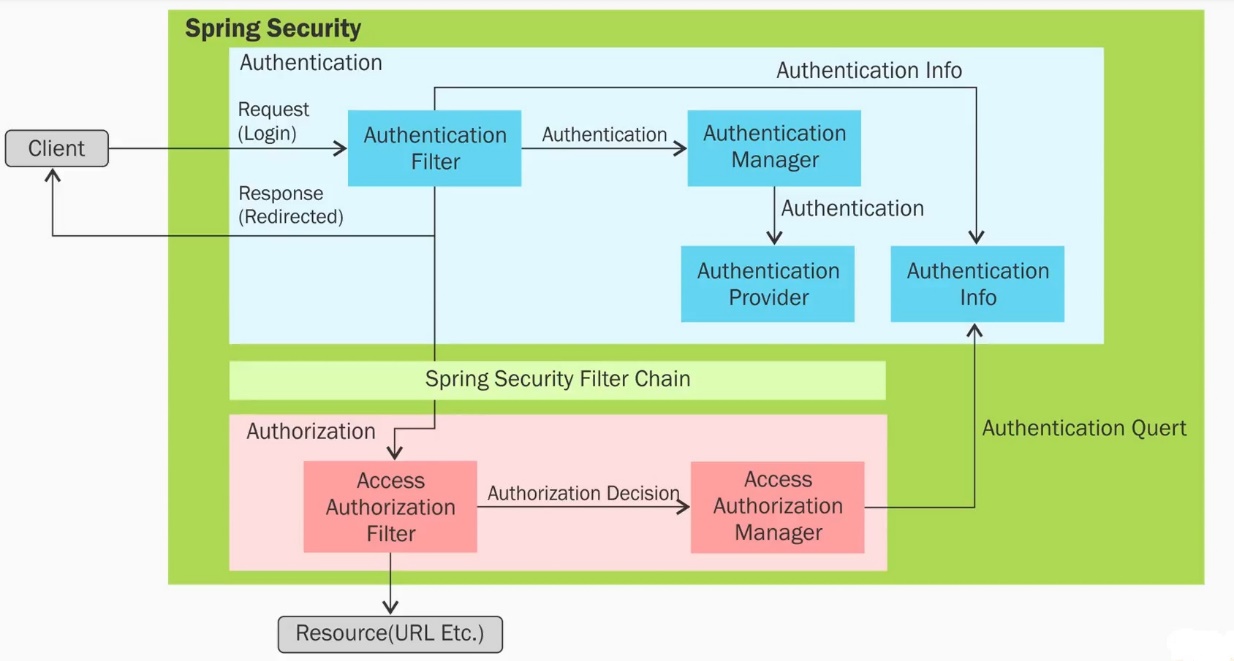


*Рисунок 5. Архитектура приложения.*

Как видно из рисунка 5, запрос поступает на наше приложение. Запрос обрабатывается controller’ом и, при необходимости, обращается за данными к Model (в нашем случаи JPA). После завершения обработки, controller возвращает пользователю ответ в виде странички (View).

## ***Безопасность***

Для обеспечения безопасности был использован Spring Security. Spring Security - это фреймворк, который сфокусирован на обеспечение как аутентификации, так и авторизации в Java-приложениях. Как и все Spring проекты, настоящая сила Spring Security в том, что он может быть легко дополнен нужным функционалом. На рисунке 6 представлена архитектура безопасности библиотеки Security.



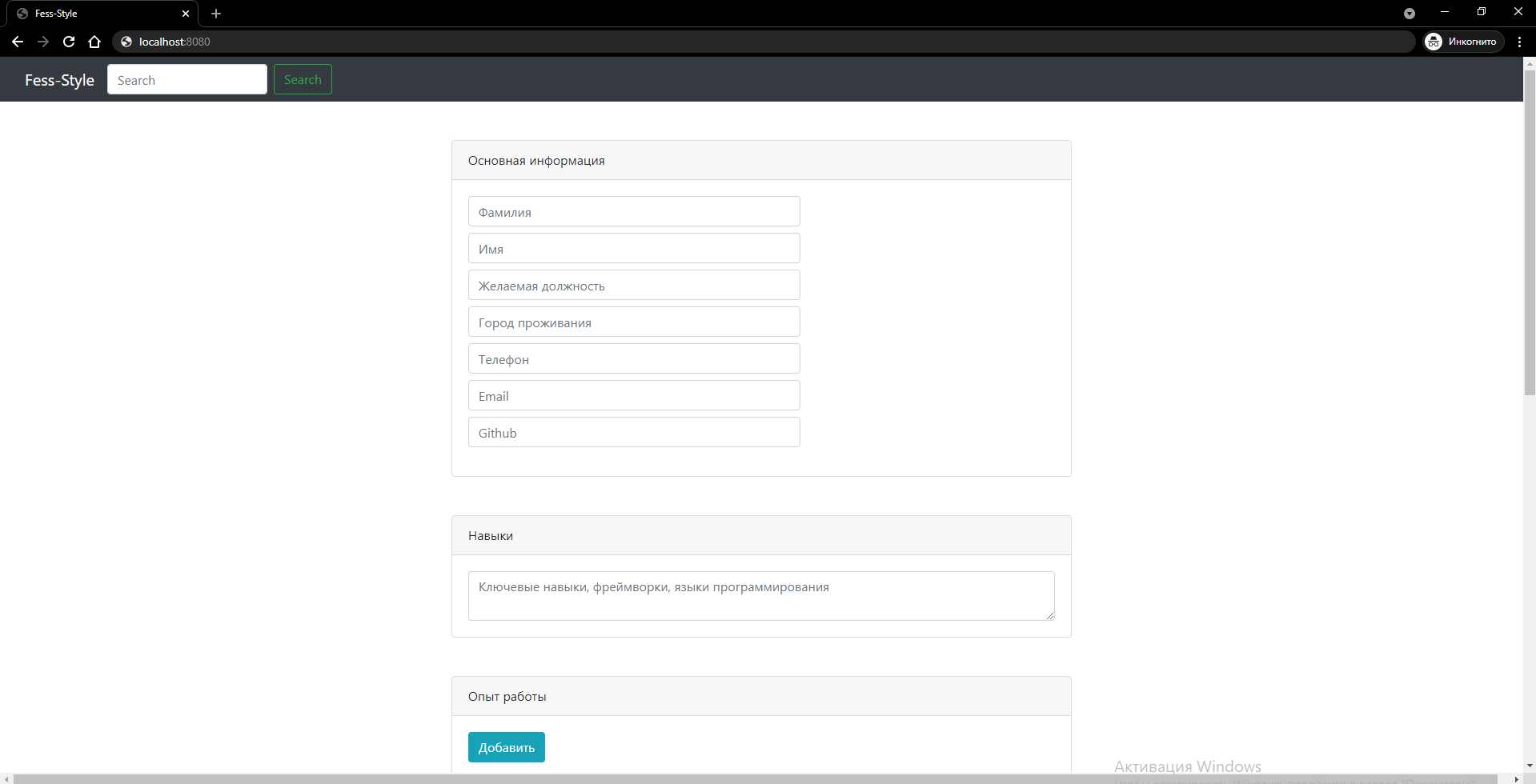
*Рисунок 6. Архитектура Spring Security.*

Самым фундаментальным объектом является SecurityContextHolder. В нем хранится информация о текущем контексте безопасности приложения, который включает в себя подробную информацию о пользователе (принципале), работающим с приложением. Spring Security использует объект Authentication, пользователя авторизованной сессии.

«Пользователь» – это просто Object. В большинстве случаев он может быть приведен к классу UserDetails. UserDetails можно представить, как адаптер между БД пользователей и тем что требуется Spring Security внутри SecurityContextHolder.

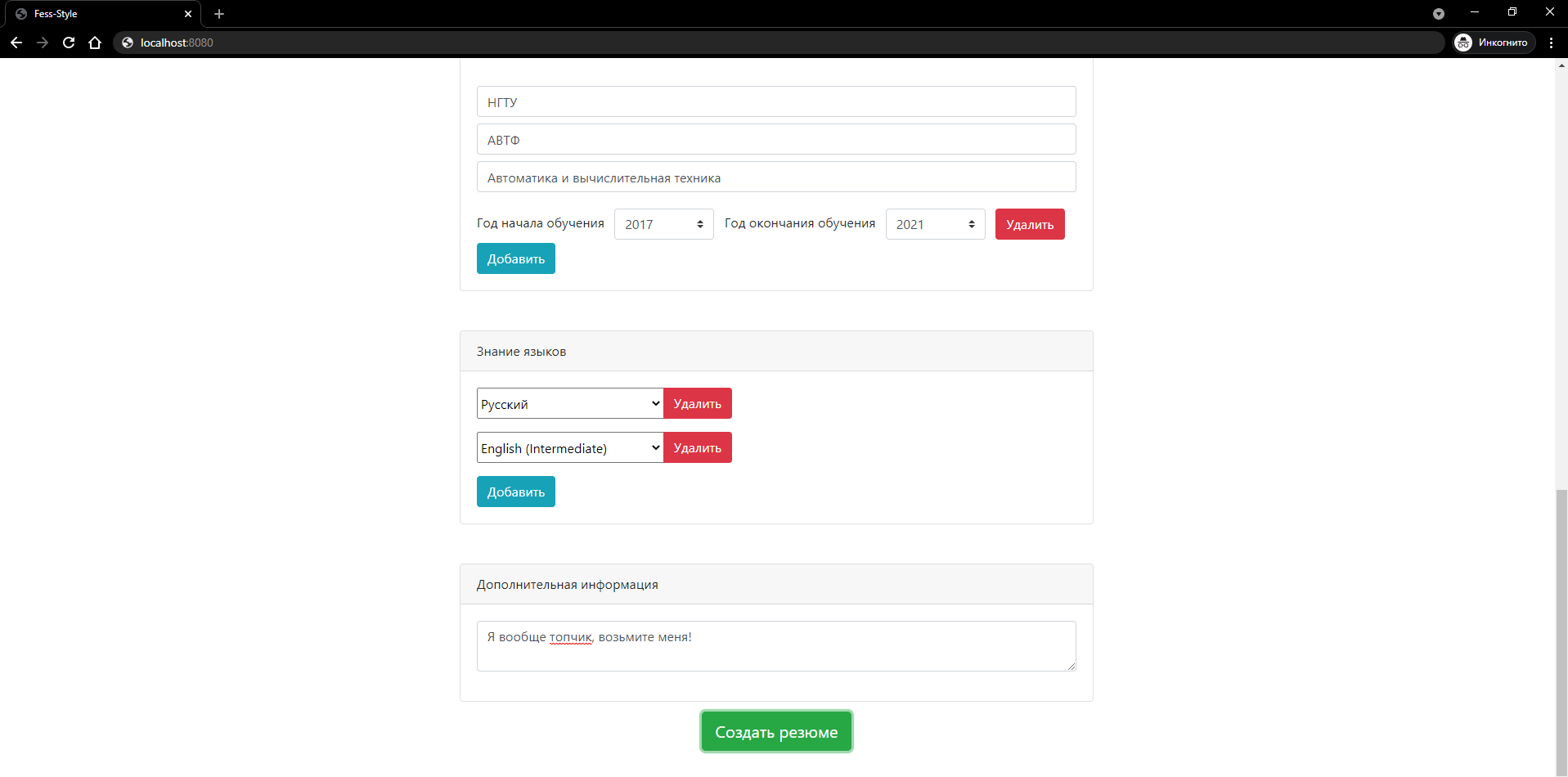
## ***Результат***

При переходе на наш Web-сервер, пользователь встречается с формой по созданию резюме.



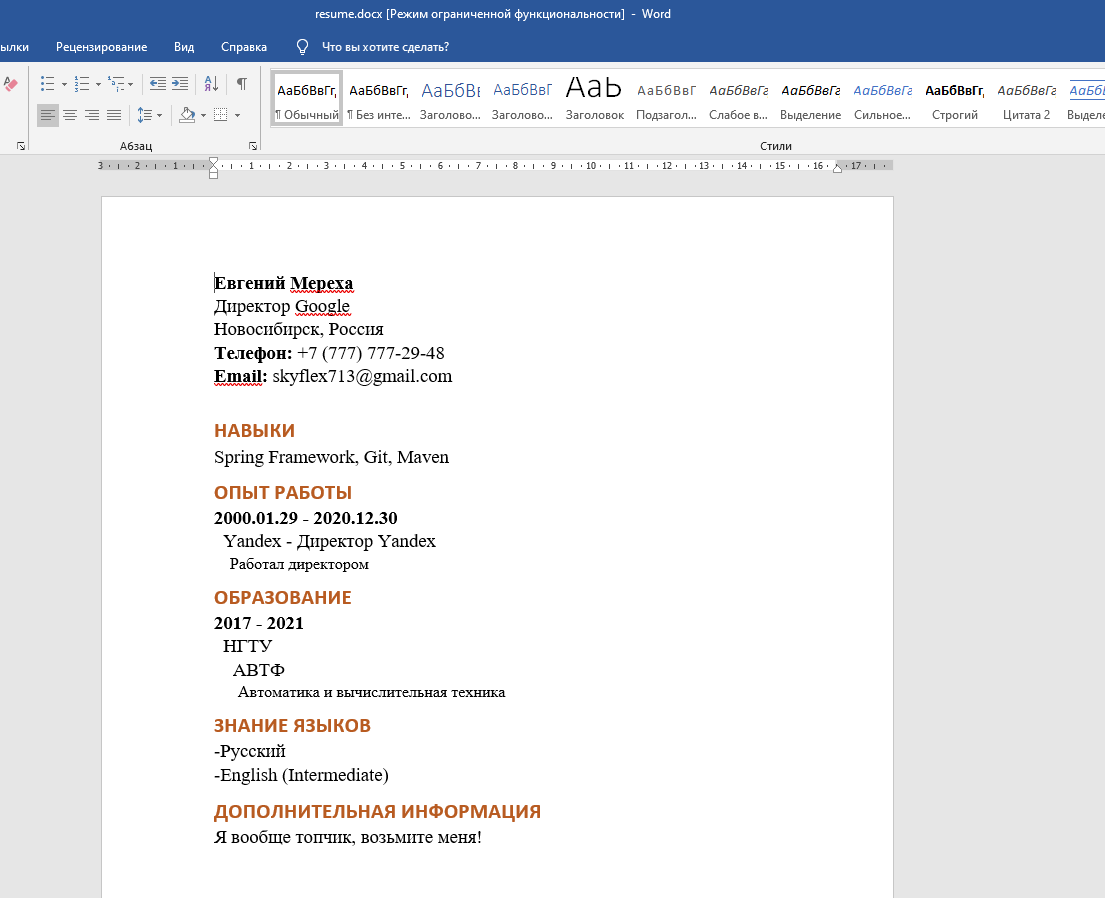
*Рисунок 7. Стартовая страница конструктора.*

После заполнения данных, пользователю предложат создать резюме.



*Рисунок 8. Завершение создания резюме.*

По нажатию кнопки “Создать резюме” веб приложение сформирует резюме в формате Word, при условии, что все данные введены правильно.



*Рисунок 9. Пример готового резюме.*

Как видно из рисунка 9, приложение сформировало правильно-структурированное IT резюме в формате docx и, что самое главное, уместил все на одной странице.

## ***Заключение***

В основе данной работы лежит проблема формирования грамотного IT-резюме, в ходе обсуждения которой был проанализирован рынок, рассмотрена актуальность проекта, сформирована команда, определен конечный потребитель, а также определены предварительные риски.

Проведя декомпозицию проекта, была создана структура реализации проекта, распределены работы по участникам команды и размечены этапы проектирования, отображенные в календарном планировании, диаграмме Ганта и диаграмме предшествования.

Исходя из средних почасовых заработных плат членов команды, был сформирован бюджет проекта, совместно с которым обозначены ресурсы проекта и определены риски проекта, а также их оценка.

В рамках данной работы была спроектирована и реализована система по созданию IT-Резюме, обладающая всем требуемым функционалом. Перед проектированием были рассмотрены возможные решения и выделены их плюсы и минусы.

В ходе выполнения работы были выбраны инструменты и технологии, выделены требования, была спроектирована база данных и бизнес-логика.

Для решения поставленной задачи использовались технологии как: Java 8, Mysql, Apache Tomcat, Maven, Spring Framework, Thymeleaf, Git, Hibernate, HTML, CSS, JavaScript.

Была разработана система, которая позволяет автоматизировать работу Web-приложения по хранению информации такой как: данные пользователя, готовые резюме.

Для системы разработан веб-интерфейс с возможностью поиска, просмотра, редактирования данных для создания IT-резюме.

## ***Список литературы***

1. Официальная документация Spring. URL: https: // spring. io/docs (Дата обращения: 28.02.2021)

2. Spring IO Platform Reference Guide. URL: http://docs. spring. io/spring/docs/4.3.0. RC2/spring-framework-reference/htmlsingle/ (Дата обращения 29.02.2021)

3. Accessing Relational Data using JDBC with Spring. URL: https: // spring. io/guides/gs/relational-data-access/ (Дата обращения: 29.02.2021)

4. Seth Ladd, Darren Davison. Expert Spring MVC and Web Flows [2006]

5. Кларенс Хо, Роб Харроп. Spring 5 для профессионалов [2019]

6. Быстрый старт spring. URL: http://spring-projects.ru/projects/spring-framework/ (Дата обращения: 29.02.2021)