|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ | | | | |
| |  | | --- | | НАБЕРЕЖНОЧЕЛНИНСКИЙ ИНСТИТУТ  (ФИЛИАЛ) ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО АВТОНОМНОГО  ОБРАЗОВАТЕЛЬНОГО УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ  «КАЗАНСКИЙ (ПРИВОЛЖСКИЙ) ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ» | | Кафедра «Информационные системы» | | | | | |
| Утверждаю | | | | |
| Заведующий кафедрой ИС | | | | |
|  | | | | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Р.А.Валиев | | | | |
|  | | | | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_г. | | | | |
|  | | | | |
|  | | | | |
| **КУРСОВАЯ РАБОТА** | | | | |
| по дисциплине: | | | | |
| **«Проектирование и архитектура программных систем»** | | | | |
| на тему: | | | | |
| **«Проектирование требований на разработку архитектуры программных систем»** | | | | |
|  | | | | |
|  | | | | |
|  | | | | |
| Автор: | | Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | |
| студент группы 2181121 | |  | | |
|  | | Руководитель: | | |
| \_\_\_\_\_\_\_\_\_\_\_\_ | Е.С.Гаврилов | Доцент кафедры ИС | | |
|  |  |  |  | |
|  |  | \_\_\_\_\_\_\_\_\_\_\_\_ | Ш.А.Хамадеев | |
|  |  |  | | |
|  |  | Дата защиты:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_г. | | |
|  |  |  |  | |
|  | | | | |
|  | | | | |
|  | | | | |
| Набережные Челны | | | | |
| 2022 | | | | |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ | | | | |
| НАБЕРЕЖНОЧЕЛНИНСКИЙ ИНСТИТУТ (ФИЛИАЛ) ФЕДЕРАЛЬНОГО государственноГО АВТОНОМНОГО образовательноГО учреждениЯ высшего образования  «КАЗАНСКИЙ (ПРИВОЛЖСКИЙ) ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ» | | | | |
| КАФЕДРА ИНФОРМАЦИОННЫХ СИСТЕМ (ИС) | | | | |
| Направление подготовки 09.03.04  «Программная инженерия» | | | | |
|  | | | | |
| Утверждаю | | | | |
| Заведующий кафедрой ИС | | | | |
|  | | | | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Р.А.Валиев | | | | |
|  | | | | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_г. | | | | |
|  | | | | |
| **ЗАДАНИЕ НА КУРСОВУЮ РАБОТУ** | | | | |
|  | | | | |
| Студент: | | | | |
| Гаврилов Евгений Святославович | | | | |
| 1 Тема: | | | | |
| **«Проектирование требований на разработку архитектуры программных систем»** | | | | |
| 2 Срок представления к защите | | | | |
| \_\_\_\_\_\_\_\_\_\_\_\_\_ | | | | |
| 3 Исходные данные | | | | |
| - Информационная система для контроля знаний сотрудников на предприятии | | | | |
| 4 Перечень подлежащих разработке вопросов  - Описание бизнес-требований; | | | | |
| - Разработка модели данных;  - Моделирование бизнес-процессов;  - Разработка пользовательских требований;  - Разработка функциональных требований;  - Разработка нефункциональных требований. | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Задание выдано | \_\_\_\_\_\_\_\_\_\_\_\_г. | \_\_\_\_\_\_\_\_\_\_ | Ш.А.Хамадеев |
| Задание принято | \_\_\_\_\_\_\_\_\_\_\_\_г. | \_\_\_\_\_\_\_\_\_\_ | Е.С.Гаврилов |

**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ** 5](#_Toc103647890)

[**1.1 Описание бизнес-требований** 7](#_Toc103647891)

[1.1.1 Исходные данные 7](#_Toc103647892)

[1.1.2 Возможности бизнеса 7](#_Toc103647893)

[1.1.3 Бизнес цели 7](#_Toc103647894)

[1.1.4 Критерии успехов 7](#_Toc103647895)

[1.1.5 Видение решения 7](#_Toc103647896)

[1.1.6 Бизнес-риски 8](#_Toc103647897)

[1.1.7 Предположения и зависимости 8](#_Toc103647898)

[1.1.8 Основные функции решения 8](#_Toc103647899)

[1.1.9 MVP 9](#_Toc103647900)

[1.1.10 Прочие версии продукта 9](#_Toc103647901)

[1.1.11 Стейкхолдеры 9](#_Toc103647902)

[1.1.12 Приоритеты проекта 9](#_Toc103647903)

[1.1.13 Варианты пользования 10](#_Toc103647904)

[**1.2 Разработка модели данных** 10](#_Toc103647905)

[1.2.1 Выделение существительных и глаголов 10](#_Toc103647906)

[1.2.2 Разработка концептуальной карты 10](#_Toc103647907)

[1.2.3 Разработка ER-модели 11](#_Toc103647908)

[**1.3 Моделирование бизнес-процессов** 12](#_Toc103647909)

[1.3.1 Описание предметной области. 12](#_Toc103647910)

[1.3.2 Табличное представление результатов анализа процесса 13](#_Toc103647911)

[1.3.3 BPMN2 модель процессов 14](#_Toc103647912)

[**1.4 Разработка пользовательских требований** 15](#_Toc103647913)

[1.4.1 Диаграмма вариантов использования 15](#_Toc103647914)

[1.4.2 Спецификации для каждого варианта использования. 15](#_Toc103647915)

[**1.6 Разработка нефункциональных требований** 17](#_Toc103647916)

[1.6.1 Пользовательские интерфейсы 17](#_Toc103647917)

[1.6.2 Коммуникационные интерфейсы 17](#_Toc103647918)

[1.6.3 Требования к удобству использования 18](#_Toc103647919)

[1.6.4 Требования к производительности 18](#_Toc103647920)

[1.6.5 Требования к безопасности 18](#_Toc103647921)

[1.6.6 Требования к надежности 18](#_Toc103647922)

[**2 Проектирование Web-приложения для контроля знаний сотрудников на предприятии** 19](#_Toc103647923)

[2.1 Архитектура веб-приложения 19](#_Toc103647924)

[2.2 Функциональная модель системы 20](#_Toc103647925)

[2.3 Разработка алгоритмов системы 25](#_Toc103647926)

[2.3.1 Алгоритм функции создания теста 25](#_Toc103647927)

[2.3.2Алгоритм функции прохождения теста 26](#_Toc103647928)

[2.3.3 Структура базы данных 27](#_Toc103647929)

[2.4 Вывод по главе 29](#_Toc103647930)

[**ЗАКЛЮЧЕНИЕ** 30](#_Toc103647931)

[**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ** 31](#_Toc103647932)

# **ВВЕДЕНИЕ**

Компания «ПерТехПроф» - платформа, специализирующаяся на проверке квалификаций сотрудников по разным техническим специальностям.

Проблема состоит в том, что в технологической сфере сотрудникам постоянно требуется проходить переаттестации для проверки актуальности знаний, повышения квалификации и трудоустройства, поэтому компаниям требуются средства для упрощения проведения этих мероприятий и минимизации уровня человеческого фактора в процессе.

Целью данной работы является автоматизация работы отдела кадров, сертификации квалификаций сотрудников с помощью автоматизированного тестирования

Для достижения поставленной цели требуется решить следующие задачи:

* проанализировать рассматриваемый процесс;
* разработать модели рассматриваемого процесса;
* разработать пользовательские требования процесса;
* разработать функциональные требования системы;
* разработать нефункциональные требования системы.

Структура работы обусловлена предметом, целью и задачами исследования. Работа состоит из введения, шести глав и заключения. Введение раскрывает актуальность, определяет степень научной разработки темы, объект, предмет, цель, задачи и методы исследования, раскрывает теоретическую и практическую значимость работы «Информационная система для учёта кадров предприятия». В первой главе рассматриваются бизнес-требования. Во второй главе описываются модели данных. Третья глава посвящена моделированию бизнес-процессов. Четвёртая, пятая и шестая главы раскрывают все необходимые требования к информационной системе. В заключении подводятся итоги исследования, формируются окончательные выводы по рассматриваемой теме.

# **1.1 Описание бизнес-требований**

## 1.1.1 Исходные данные

Компаниям промышленной и IT сфер требуется постоянно проверять квалификацию сотрудников, проводить переаттестации и опросы. Очевидна неэффективность использования при этом проверяющих сотрудников и бумажных средств аттестации, так как можно минимизировать затраты на данных сотрудников, подготовку средств переаттестации, бумагу и места для хранения и учета результатов. В связи с этим было решено создать автоматизированную систему аттестации для сотрудников различных специализаций и квалификаций этих компаний.

## 1.1.2 Возможности бизнеса

Сервис с возможность создания, прохождения и учета автоматизированных тестов позволит эффективно использовать время сотрудников и затраченные ресурсы компании.

## 1.1.3 Бизнес цели

1. Сократить время на проведение аттестаций на 75% ко IV кварталу 2022 года.
2. Сократить время на проведение аттестаций на 75% ко IV кварталу 2022 года.
3. Провести тестовую эксплуатацию в III квартале 2022 года.

## 1.1.4 Критерии успехов

1. 100% реализация базовой функциональности ко IV кварталу 2022 года.
2. Время формирования плана при заданных значениях составит не более 1,5 часов.

## 1.1.5 Видение решения

Решение представляет собой одностраничное веб-приложение с API. Данная платформа даст техническим учреждениям сократить расходы, увеличить скорость, точность и независимость процесса проверки знаний сотрудников, избавиться от лишних издержек, связанных с организацией и распечаткой, это необходимо предприятиям с высокой кадровой нагрузкой, и становится выгодным для команд, содержащих более 10 сотрудников. Веб-приложение будет иметь возможности создания удобных тестов работниками отдела кадров, прохождения тестов в любом месте и в любое время, безопасного сохранения статистики без возможности ошибки из-за человеческого фактора, удобного просмотра статистики по прохождению тестов.

Также приложение даст возможность интеграции внутренних систем с результатами тестов, с помощью чего можно временно аннулировать пропуска рабочих и не допускать их к работе до прохождения теста.

## 1.1.6 Бизнес-риски

1. Вероятность того, что некоторые предприятия уже используют подобные внутренние системы.
2. Периодические требования к интеграциям с внутренними системами предприятий, вынуждающие либо потратить дополнительные ресурсы на создание интеграции, либо отказаться от сотрудничества.
3. Возможность утечки персональных данных. Недостаточный уровень качества планирования.

## 1.1.7 Предположения и зависимости

1. Наличие интернет соединения для доступа к прохождению тестов

## 1.1.8 Основные функции решения

1. Создание аккаунта компании для сотрудников отдела кадров
2. Добавление аккаунтов сотрудников компании
3. Создание тестов сотрудниками отдела кадров
4. Прохождение тестов сотрудниками компании
5. Просмотр решений тестов
6. Просмотр сертификатов по решениям сотрудниками компании

## 1.1.9 MVP

1. Создание учетных записей компании
2. Добавление тестов с вопросами и ответом
3. Просмотр прохождения теста сотрудниками

## 1.1.10 Прочие версии продукта

1. Создание удобной мобильной версии сайта и мобильного приложения.

## 1.1.11 Стейкхолдеры

* Руководство:
  1. Ценность: снижение издержек, повышение эффективности
  2. Отношение: высокая заинтересованность
  3. Интересы: минимизация затрат, уменьшение рабочей нагрузки
  4. Ограничения: стоимость обслуживания
* Сотрудники:
  1. Ценность: удобство и ускорение проверки знаний
  2. Отношение: высокая заинтересованность
  3. Интересы: простота функциональности данной системы
  4. Ограничения: время разработки данной системы

## 1.1.12 Приоритеты проекта

* Функции:
  + Все функции MVP должны быть выполнены
* Качество:
  + Критическая функциональность должна работать без ошибок.
* Срок
  + Версия MVP должны быть запущена в тестовую эксплуатацию до конца IV квартала 2022 года.
* Расходы:
  + 500 000 рублей на разработку.
* Персонал
  + Предполагаемый персонал: руководитель проекта, 1 бэк-энд разработчик С#, 1 фронт-энд разработчик React, 1 тестировщик, 1 специалист по интеграции.

## 1.1.13 Варианты пользования

* Руководство:
  1. Регистрация новых сотрудников
  2. Создание тестов
  3. Проверка результатов тестов
* Сотрудники предприятия:
  1. Прохождение тестов

# **1.2 Разработка модели данных**

## 1.2.1 Выделение существительных и глаголов

В ходе анализа бизнес-требований был сформирован список из существительных-объектов, глаголов и существительных-ролей.

Таблица 1 – Список из существительных объектов, глаголов и ролей

|  |  |  |
| --- | --- | --- |
| Существительные | Глаголы | Роли |
| Тест | Создаёт | Руководство |
| Решение | Получает | Сотрудник предприятия |
| Сертификат | Просматривает |  |
| Результат | Выдает |  |
|  | Формирует |  |

## 1.2.2 Разработка концептуальной карты

В результате анализа были определены основные сущности Тест, Решение, Сертификат и Результат. Эти сущности еще называются ресурсами. Ресурсы могут иметь разные атрибуты, которые хранятся в Справочнике атрибутов.

Руководство создаёт тесты и, после их прохождения сотрудниками, просматривает результаты.

Сотрудник предприятия получает доступ к тесту, проходит его и выдает решение, на основе которого формируется результат, и в случае успешного прохождения теста, создается сертификат, подтверждающий успешное прохождение тестирования.

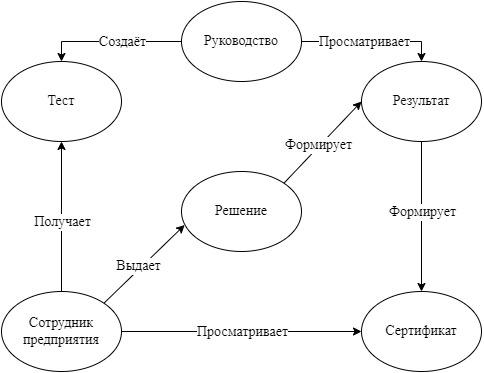


Рисунок 1 - Концептуальная карта предметной области

## 1.2.3 Разработка ER-модели

Исходными данными является концептуальная карта, в которой уже выявлены основные сущности и связи между ними. Чтобы получить ER-модель, требуется сформировать их связи и определить основные атрибуты сущностей.

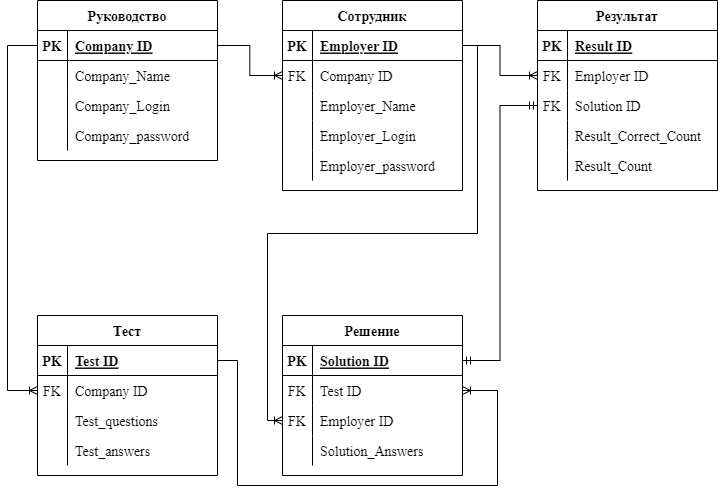


Рисунок 1.1 – Пример ER-модели с атрибутами

# **1.3 Моделирование бизнес-процессов**

## 1.3.1 Описание предметной области.

ООО «ХимТех» является крупным предприятием в области химической промышленности. В данный момент в этой организации для перепроверки квалификации сотрудников и при приеме новых требуется проводить крупное мероприятие по созданию, прохождению и проверке квалификационных тестов. Для минимизации издержек по проведению данных мероприятий компания решила ввести систему автоматизированного тестирования сотрудников.

## 1.3.2 Табличное представление результатов анализа процесса

Результаты анализа процесса создания и редактирования теста:

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название | Тип | Роль |
| 1 | Необходимость создания или редактирования теста | Событие | Руководство |
| 2 | Необходимость добавления вопроса | Событие | Руководство |
| 3 | Отсутствие необходимости добавления вопроса | Событие | Руководство |
| 4 | Добавление вопроса и ответа | Действие | Руководство |
| 5 | Необходимость удаления вопроса | Событие | Руководство |
| 6 | Отсутствие необходимости удаления вопроса | Событие | Руководство |
| 7 | Удаление вопроса | Действие | Руководство |
| 8 | Требуется оформить тест | Событие | Руководство |
| 9 | Не требуется оформить тест | Событие | Руководство |
| 10 | Оформление теста | Результат | Руководство |

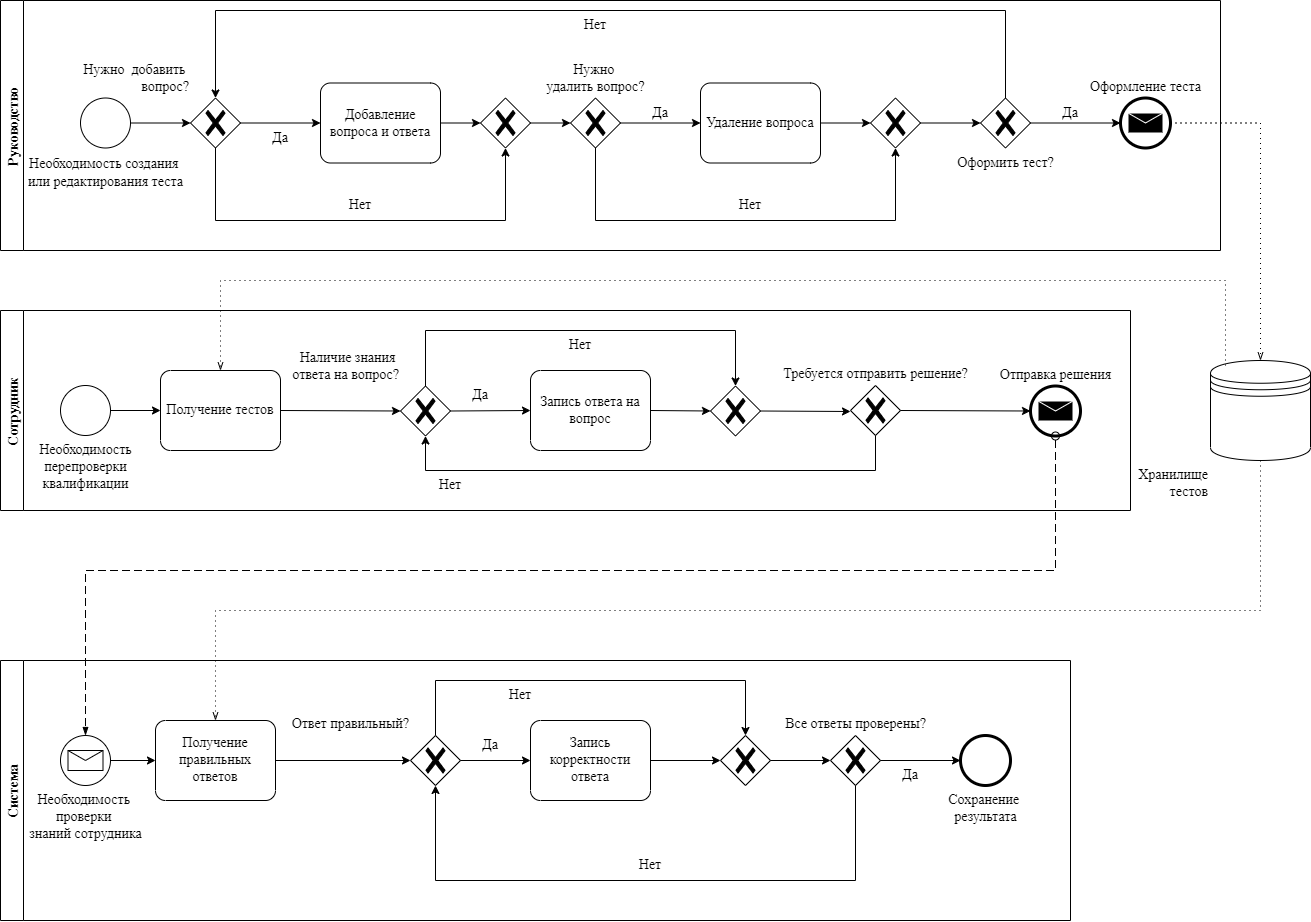
Результаты анализа процесса прохождения теста:

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название | Тип | Роль |
| 1 | Необходимость перепроверки квалификации | Событие | Сотрудник |
| 2 | Наличие знания ответа на вопрос | Событие | Сотрудник |
| 3 | Отсутствие знания ответа на вопрос | Событие | Сотрудник |
| 4 | Запись ответа на вопрос | Действие | Сотрудник |
| 5 | Требуется отправить решение | Событие | Сотрудник |
| 6 | Не требуется отправить решение | Событие | Сотрудник |
| 7 | Отправка решения | Результат | Сотрудник |

Результат анализа процесса проверки теста:

|  |  |  |  |
| --- | --- | --- | --- |
| № | Название | Тип | Роль |
| 1 | Необходимость проверки знаний сотрудника | Событие | Система |
| 2 | Получение правильных ответов | Действие | Система |
| 3 | Ответ совпадает с правильным | Событие | Система |
| 4 | Ответ не совпадает с правильным | Событие | Система |
| 5 | Запись корректности ответа | Действие | Система |
| 6 | Все ответы проверены | Событие | Система |
| 7 | Не все ответы проверены | Событие | Система |
| 8 | Сохранение результата | Результат | Система |

## 1.3.3 BPMN2 модель процессов



# **1.4 Разработка пользовательских требований**

## 1.4.1 Диаграмма вариантов использования

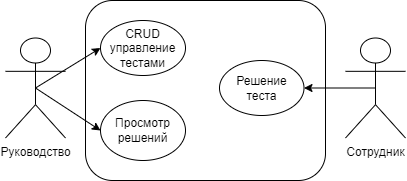


Рисунок 8 – Диаграмма вариантов использования

## 1.4.2 Спецификации для каждого варианта использования.

|  |  |
| --- | --- |
| **UC-1. CRUD управление тестами** | |
| **Определение/**  **Ценность** | Операция предназначена для создания, изменения и удаления тестов |
| **Пользователь** | Руководство |
| **Предусловие** | Пользователь авторизован и инициировал выполнение операции |
| **Основной сценарий** | 1. Пользователь создает новый тест.  2. Пользователь добавляет вопросы.  3. Система сохраняет тест в базе данных. |
| **Альтернативные пути** | 1a. Пользователь начал редактировать тест |
| **Результат** | Тест сохранён |

|  |  |
| --- | --- |
| **UC-2. Просмотр решений** | |
| **Определение/**  **Ценность** | Операция предназначена для контроля сотрудников |
| **Пользователь** | Руководство |
| **Предусловие** | Пользователь авторизован и инициировал выполнение операции |
| **Основной сценарий** | 1. Система предлагает пользователю просмотреть решения теста  2.Пользователь открывает решения.  3. Пользователь просматривает результаты сотрудников. |
| **Альтернативные пути** | - |
| **Результат** | Решение просмотренно |

|  |  |
| --- | --- |
| **UC-3. Решение теста** | |
| **Определение/**  **Ценность** | Операция предназначена для фиксации решения теста сотрудником |
| **Пользователь** | Сотрудник |
| **Предусловие** | Пользователь инициировал выполнение операции |
| **Основной сценарий** | 1. Пользователь выбирает тест.  2. Пользователь отвечает на вопросы теста.  3. Пользователь отправляет решение.  4. Система проверяет решение пользователя и сохраняет результат в базу данных. |
| **Альтернативные пути** | 2а.Пользователь не отвечает на вопросы |
| **Результат** | Решение проверено и получен результат |

**1.5 Разработка функциональных требований**

|  |  |
| --- | --- |
| **Идентификатор** | **Функции/требования** |
| **F1** | CRUD управление тестами |
| **FR-1.1** | Система должна предоставить пользователю форму для ввода вопроса и ответа. |
| **FR-1.2** | Система должна предоставить пользователю кнопку для добавления вопроса. |
| **FR-1.3** | Система должна предоставить пользователю кнопки для удаления вопросов. |
| **FR-1.4** | Система должна предоставить форму для ввода количества правильных ответов, необходимого для прохождения теста. |
| **FR-1.5** | Система должна предоставить кнопку для сохранения теста. |
| **FR-1.6** | Система должна отображать информацию о всех тестах принадлежащих компании. |
| **FR-1.7** | Система должна предоставить кнопку для добавления теста. |
| **FR-1.8** | Система должна предоставить кнопки для удаления тестов. |

|  |  |
| --- | --- |
| **Идентификатор** | **Функции/требования** |
| **F2** | Просмотр решений |
| **FR-2.1** | Система должна предоставить кнопку для перехода к просмотру решений теста. |
| **FR-2.2** | Система должна отображать список решений теста с данными о сотруднике и успехе его решения. |

|  |  |
| --- | --- |
| **Идентификатор** | **Функции/требования** |
| **F3** | Решение теста |
| **FR-3.1** | Система должна отображать список всех вопросов с формой ответа. |
| **FR-3.2** | Система должна предоставить кнопку для отправки решения. |

# **1.6 Разработка нефункциональных требований**

## 1.6.1 Пользовательские интерфейсы

|  |  |
| --- | --- |
| **UI-1** | Интерфейс создания тестов должен быть интуитивно понятен. |
| **UI-2** | Все списки должны иметь фильтрацию по имени. |
| **UI-3** | Отображения результатов должны сопровождаться зеленым цветом в случае успешного решения и красным в ином случае. |

## 1.6.2 Коммуникационные интерфейсы

|  |  |
| --- | --- |
| **CI-1** | Система на вызов метода API может дать два варианта ответов:  1. Успешный: код “200 ОК” и, в некоторых вызовах, данные (например, идентификатор объекта).   1. Ошибки:   1. код ошибки “400” - ошибка валидации и причина;  2. код ошибки “401” - ошибка авторизации;  3. код ошибки “500” - ошибка на сервере. |
| **CI-2** | API системы должно быть реализовано с помощью архитектуры REST |

## Требования к удобству использования

|  |  |
| --- | --- |
| **USE-1** | Система должна позволить пользователю создавать вопросы минимальным количеством действий. |
| **USE-2** | Система должна иметь высокий уровень доступности сайта. |

## 1.6.4 Требования к производительности

|  |  |
| --- | --- |
| **PER-1** | Система должна проводить операции с тестом не более 30 секунд. |

## 1.6.5 Требования к безопасности

|  |  |
| --- | --- |
| **SEC-1** | Пользователи для выполнения всех операций системы обязательно должен быть авторизованным в системе. |
| **SEC-2** | Должен использоваться протокол HTTPs |
| **SEC-3** | Токен для авторизации методов API должен быть действителен 30 минут и не должен требовать повторного запроса при каждой операции. |

## 1.6.6 Требования к надежности

|  |  |
| --- | --- |
| **ER-1** | При возникновении ошибки обработки данных должно быть сформировано уведомление для пользователя. |

# **2 Проектирование Web-приложения для контроля знаний сотрудников на предприятии**

Главной целью разработки информационной системы является решение проблемы путём оптимизации или усовершенствования текущего бизнес процесса.

Для разработки данного проекта требуется определить:

* архитектуру веб-приложения;
* функциональную модель системы;
* алгоритмы основных функций системы;
* структуру базы данных.

### 2.1 Архитектура веб-приложения

Разработка программного обеспечения представляет собой достижение нескольких прикладных целей, среди которых:

* возможность увеличения и масштабирования приложения;
* абстрактность архитектуры;
* модульность;
* скорость разработки.

Данные прикладные цели определяют качество программы с технической стороны.

Для достижения всех этих целей будет применён паттерн проектирования архитектуры MVC (Model-View-Controller).

Преимущества использования MVC заключается в:

* чётком разделении логики;
* поддержке различных типов пользователей, использующих различные устройства;
* упрощение больших приложений;
* функциональная структуризация кода, что облегчает поддержку приложения.

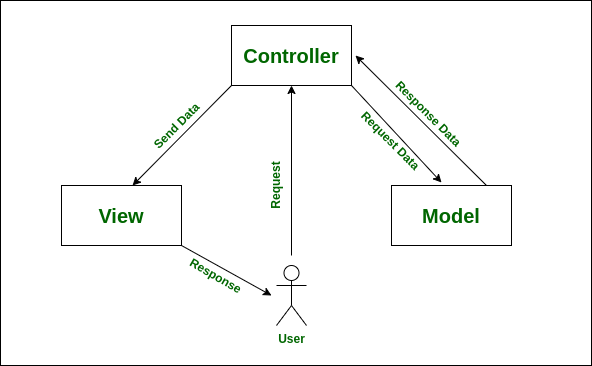


Рисунок 2.1 – Модель MVC

Данный паттерн состоит из 3 блоков:

* Контроллер (Controller), необходимый для управления запросами пользователей, получаемых по протоколу HTTP. Его основная функция - обработка требующихся для выполнения действий, задаваемых пользователем;
* Модель (Model) включает в себя данные и правила, которые используются для работы с данными;
* Вид (View) обеспечивает представление данных, получаемых контроллером из модели, различными способами.

### 2.2 Функциональная модель системы

На основе требований, предъявляемых к системе, были выявлены основные функции информационной системы. Опираясь на них, были определены исходные данные и получаемые результаты. Ниже будут представлены диаграмма функциональной модели системы, а также функциональная модель IDEF0.

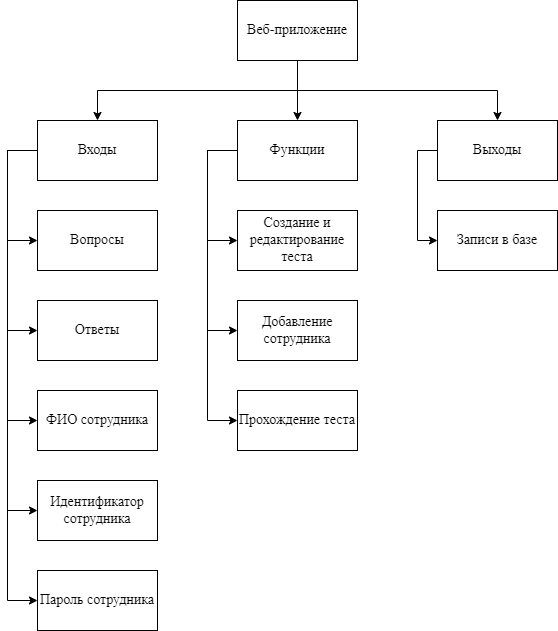


Рисунок 2.2 – Функциональная модель системы

На этапе анализа было выявлено три основные функций:

* Создание и редактирование содержания теста;
* Добавление сотрудника;
* Прохождение теста сотрудником;

Для работы данных функций будут необходимы следующие входные данные:

* Вопросы;
* Ответы;
* ФИО сотрудника;
* Идентификатор сотрудника;
* Пароль сотрудника;

Результатом выполнения всех вышеописанных функций будут:

* Соответствующие записи в базе данных;

Далее представлена более подробная функциональная модель IDEF0.

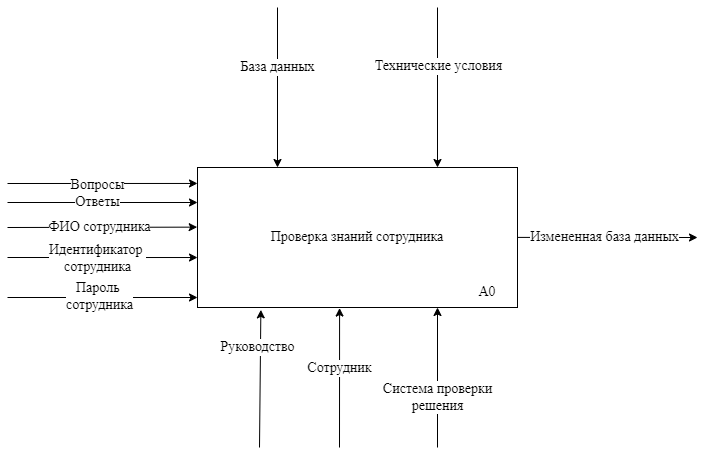


Рисунок 2.3 – Контекстная диаграмма «Проверка знаний сотрудника»

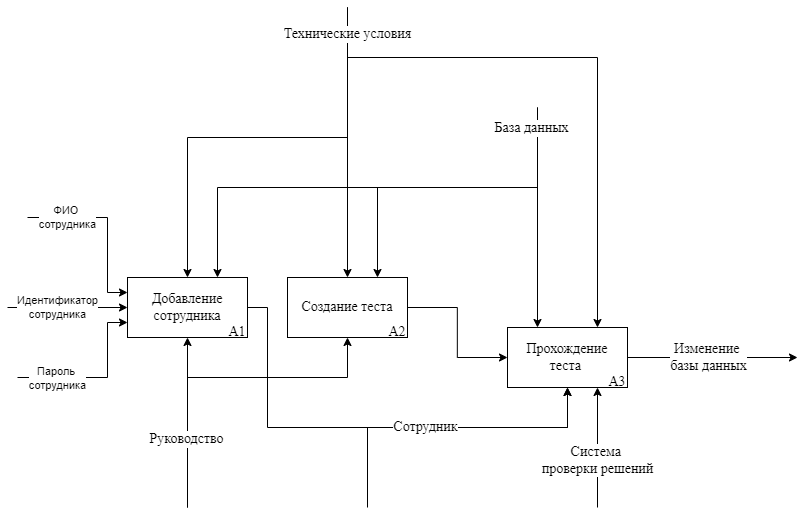
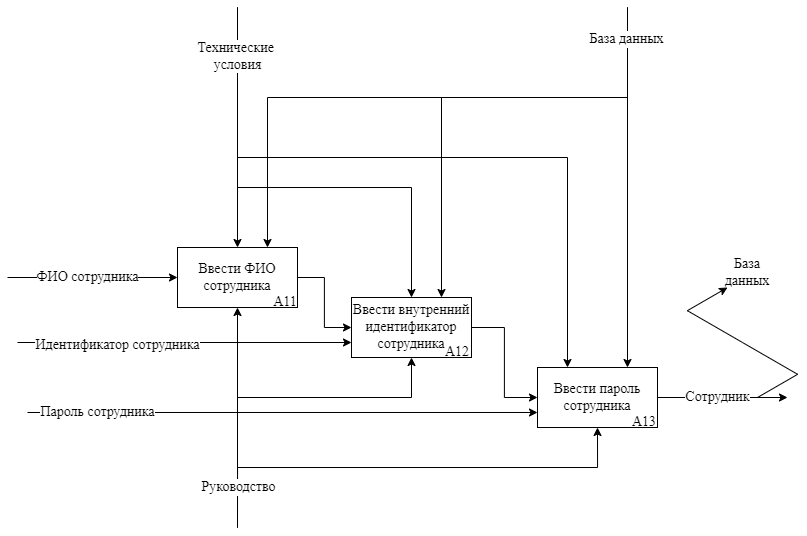


Рисунок 2.4 – Диаграмма декомпозиции первого уровня IDEF0

Рисунок 2.5 – Диаграмма декомпозиции процесса добавление сотрудника

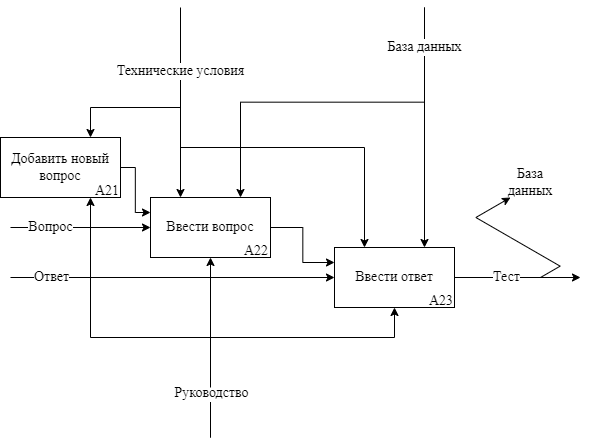


Рисунок 2.6 – Диаграмма декомпозиции процесса создания теста

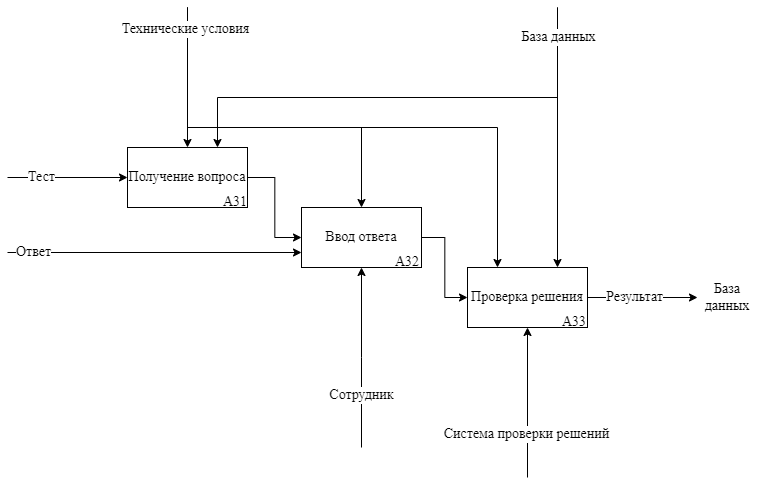


Рисунок 2.7 – Диаграмма декомпозиции процесса прохождения теста

### 2.3 Разработка алгоритмов системы

Для доступа в систему у каждого сотрудника компании есть свой персональный аккаунт. Каждому аккаунту присвоена определенная роль, это сделано в целях разграничения прав доступа.

### 2.3.1 Алгоритм функции создания теста

Руководство предприятия, при необходимости проведения тестирования, инициирует процесс создания теста, заполняя форму с полями вопроса и ответа, и при необходимости добавляет новые, после чего завершает процесс создания теста и позволяет системе сформировать тестирование для прохождения его сотрудником.

Ниже на рисунке 2.8 представлена UML диаграмма деятельности «Создание теста».



Рисунок 2.8 – UML диаграмма деятельности «Создание теста»

### 2.3.2Алгоритм функции прохождения теста

Сотрудник запускает тест, в результате чего получает вопрос и заполняет поле ответа, затем приступает к следующему вопросу. После завершения теста, система сверяет результат с порогом прохождения и в случае успешного его преодоления сохраняет результат в базе данных.

Ниже на рисунке 2.9 представлена UML диаграмма деятельности «Прохождение теста».

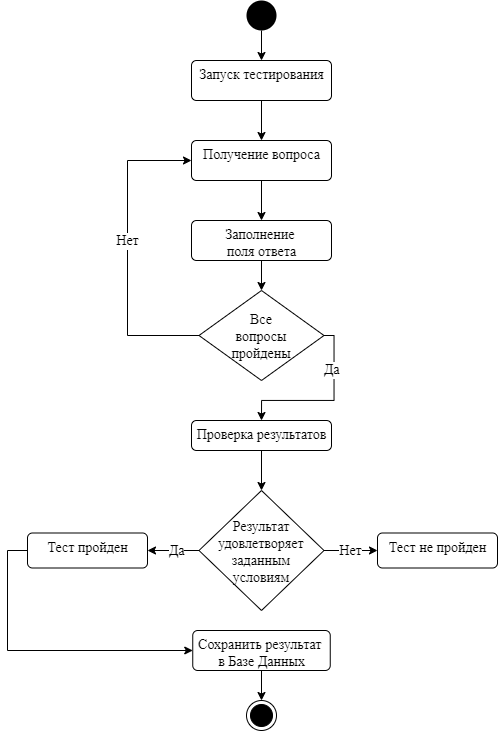
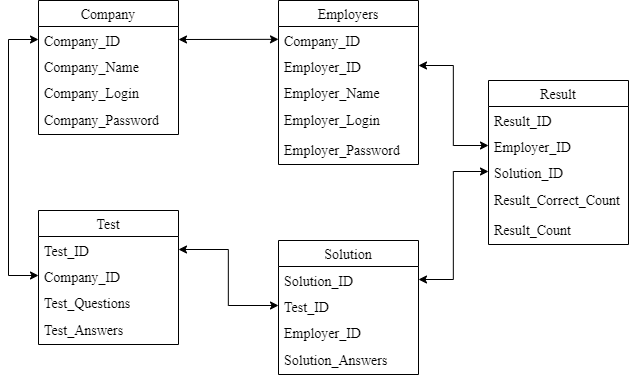


Рисунок 2.8 – UML диаграмма деятельности «Прохождение теста»

### 2.3.3 Структура базы данных

База данных для данной системы имеет пять таблиц: Company, Employers, Test, Solution, Result.

 Рисунок 2.9 – Диаграмма базы данных

Каждая из таблиц базы данных содержит определённые типы объектов:

* Company – здесь хранятся данные руководства предприятия;
* Employers – здесь хранятся данные о сотрудниках;
* Test – здесь хранятся тесты;
* Solution – здесь хранятся решения сотрудников;
* Result – здесь хранятся результаты тестирований;

Далее представлены поля таблиц баз данных и их типы данных.

Таблица 2.1 – таблица Company базы данных.

|  |  |  |
| --- | --- | --- |
| **№** | **Имя поля** | **Тип поля** |
| 1 | Company\_ID | Int |
| 2 | Company\_Login | String |
| 3 | Company\_Password | String |

Таблица 2.2 – таблица Employers базы данных.

|  |  |  |
| --- | --- | --- |
| **№** | **Имя поля** | **Тип поля** |
| 1 | Company\_ID | Int |
| 2 | Employer\_ID | Int |
| 3 | Employer\_Name | String |
| 4 | Employer\_Login | String |
| 5 | Employer\_Password | String |

Таблица 2.3 – таблица Test базы данных.

|  |  |  |
| --- | --- | --- |
| **№** | **Имя поля** | **Тип поля** |
| 1 | Test\_ID | Int |
| 2 | Company\_ID | Int |
| 3 | Test\_Questions | String |
| 4 | Test\_Answers | String |

Таблица 2.4 – таблица Solution базы данных.

|  |  |  |
| --- | --- | --- |
| **№** | **Имя поля** | **Тип поля** |
| 1 | Solution\_ID | Int |
| 2 | Test\_ID | Int |
| 3 | Employer\_ID | Int |
| 4 | Solution\_Answers | String |

Таблица 2.5 – таблица Result базы данных.

|  |  |  |
| --- | --- | --- |
| **№** | **Имя поля** | **Тип поля** |
| 1 | Result\_ID | Int |
| 2 | Employer\_ID | Int |
| 3 | Solution\_ID | Int |
| 4 | Result\_Correct\_Count | Int |
| 5 | Result\_Count | Int |

### 2.4 Вывод по главе

В данной главе была рассмотрена архитектура веб-приложения. Также были разработаны функциональная модель системы, и функциональная модель IDEF0. В ходе анализа были выявлены функции данной информационной системы, необходимые входные и получаемые выходные данные. Для наглядности было создано три диаграммы декомпозиции второго уровня для функциональной модели IDEF0.

# **ЗАКЛЮЧЕНИЕ**

В результате выполнения данной курсовой работы была разработана информационная система для работы с отделом кадров.

В первой главе были рассмотрены бизнес-требования, такие как: исходные данные, возможности, риски, критерии успеха. Так же было составлено видение решения, выделены предположения и зависимости, а так же основные функции решения. В ходе анализа бизнес-требований был сформирован список из существительных-объектов, глаголов и существительных-ролей, была разработана концептуальная карта, а так же создана ER-модель. Описав предметную область и проанализировав процессы, разработали BPMN-модель. Были рассмотрены варианты использования и их спецификации, разработаны функциональные требования системы, с помощью которых были выявлены все функции, которые должны будут выполняться системой.

Были описаны нефункциональные требования, такие как пользовательские и коммуникационные интерфейсы, требования к производительности, безопасности.

Во второй главе была рассмотрена архитектура веб-приложения. Также были разработаны функциональная модель системы, и функциональная модель IDEF0. В ходе анализа были выявлены функции данной информационной системы, необходимые входные и получаемые выходные данные. Для наглядности было создано три диаграммы декомпозиции второго уровня для функциональной модели IDEF0. Были разработаны алгоритмы системы: алгоритм создания теста, алгоритм прохождения теста. Была рассмотрена структура базы данных разрабатываемой информационной системы.

Все поставленные в данной курсовой работе задачи выполнены, цель работы достигнута.

# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Вигерс К., Битти Д. Разработка требований к программному обеспечению. 3-е изд., дополненное / Пер. с англ. — М.: Издательство «Русская редакция»; СПб. : БХВ-Петербург, 2014. — 736 стр.: ил.
2. Арлоу Д., Нейштадт И. UML 2 и Унифицированный процесс. Практический объектно-ориентированный анализ и проектирование, 2е издание. – Пер. с англ. – СПб: СимволПлюс, 2007. – 624 с., ил.
3. ГОСТ 7.32-2001. Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления. – Введ. 2002–07–01. – М.: Стандартинформ, 2006.–22 с.
4. Хамадеев Ш.А. Методология моделирования бизнес-процессов BPMN2. Учебно-методическое пособие по дисциплине «Проектирование АСОИУ». – Набережные Челны: ИПЦ НЧИ К(П)ФУ, 2017. – 36 с.
5. Хамадеев Ш.А. Методология описания пользовательских требований. Учебно-методическое пособие по дисциплине «Проектирование АСОИУ». – Набережные Челны: ИПЦ НЧИ К(П)ФУ, 2017. – 28 с
6. Функциональное моделирование на базе стандарта IDEF0. Учебный курс – Минск: 2002 – 35 с.
7. Фримен, A. ASP.NET МVС 3 для профессионалов /А. Фримен, С. Сандерсон. - Москва: Вильямс, 2012. - 680 с.: ил.
8. 13. Грейди, Б. Язык UML. Руководство пользователя / Б. Грейди - 2-е изд. - М., СПб.: ДМК Пресс, 2004. - 432 с.