

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий

(наименование института полностью)

Кафедра «Прикладная математика и информатика»

(наименование кафедры полностью)

09.03.03 Прикладная информатика

(код и наименование направления подготовки, специальности)

Корпоративные информационные системы

(направленность (профиль) / специализация)

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА  
(БАКАЛАВРСКАЯ РАБОТА)**

на тему «Разработка АИС букинга мобильных рабочих мест в офисе (на примере ООО "Холмонт БР")»

Обучающийся

Д.А. Миронов

(Инициалы Фамилия)

(личная подпись)

Руководитель

кандидат технических наук, доцент, О.В. Аникина

(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Консультант

к.п.н., доцент Т.С. Якушева

(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Тольятти 2022

## **Аннотация**

Бакалаврская работа на тему: «Разработка АИС букинга мобильных рабочих мест в офисе (на примере ООО "Холмонт БР")».

Объектом исследования выпускной квалификационной работы является бизнес-процесс букинга мобильных рабочих мест.

Предметом исследования выпускной квалификационной работы является автоматизация букинга мобильных рабочих мест.

Во введении обосновывается актуальность выбранной темы, ставится цель, определяются объект и предмет исследования, а также ставятся задачи.

В первой главе произведен анализ деятельности, дана характеристика деятельности предприятия, проведено концептуальное моделирование предметной области.

Во второй главе описано логическое проектирование информационной системы букинга мобильных рабочих мест.

В третьей главе описано физическое проектирование АИС букинга мобильных рабочих мест в офисе и проведено тестирование разработанной информационной системы.

В заключении представлены результаты и выводы о проведенной работе.

Объем выполненной бакалаврской работы – 42 страницы, рисунков – 13, таблиц – 7, литературных источников – 34.

## **Annotation**

Bachelor's work on the topic: "Development of AIS booking of mobile workplaces in the office (using the example of Haulmont BR LLC)".

The object of the study of the final qualifying work is the business process of booking mobile jobs.

The subject of the study of the final qualifying work is the automation of booking mobile jobs.

During the conduct, the relevance of the chosen topic is justified, a goal is set, the object and subject of the study are determined, and tasks are set.

In the first chapter, the analysis of the activity is carried out, the characteristics of the activity of the enterprise are given, conceptual modeling of the subject area is carried out.

The second chapter describes the logical design of the mobile workplace booking information system.

The third chapter describes the physical design of AIS booking of mobile workplaces in the office and conducted testing of the developed information system.

In conclusion, the results and conclusions of the work carried out are presented.

The volume of bachelor's work completed is 42 pages, figures – 13, tables – 7, literary sources – 34.

## Оглавление

|  |    |
|--|----|
| Введение.....  | 6  |
| Глава 1 Функциональное моделирование предметной области.....   | 8  |
| 1.1 Техничко-экономическая характеристика ООО «Haulmont» .....   | 8  |
| 1.2 Анализ бизнес-процессов ООО «Haulmont».....  | 10 |
| 1.2.1 Выбор технологии моделирования бизнес-процессов .....  | 10 |
| 1.2.2 Разработка и анализ модели бизнес-процесса «КАК ЕСТЬ».....   | 11 |
| 1.3 Анализ существующих технологий букинга мобильных рабочих мест  |    |
| 13   |    |
| 1.4 Разработка требований к проектируемой информационной системе.  | 14 |
| 1.5 Разработка модели бизнес-процесса «КАК ДОЛЖНО БЫТЬ» .....  | 15 |
| Глава 2 Логическое проектирование автоматизированной информационной системы букинга мобильных рабочих мест .....   | 18 |
| 2.1 Выбор технологии логического моделирования информационной системы.....   | 18 |
| 2.2 Логическая модель информационной системы и ее описание.....  | 20 |
| 2.2.1 Диаграмма вариантов использования информационной системы букинга .....   | 20 |
| 2.2.2 Разработка диаграммы классов информационной системы .....  | 22 |
| 2.2.3 Разработка диаграммы последовательности информационной системы .....   | 23 |
| 2.3 Разработка логической модели базы данных информационной системы и составление требований к аппаратно-программному обеспечению информационной системы ..... | 25 |
| Глава 3 Физическое проектирование информационной системы компании..  | 27 |

|  |    |
|--|----|
| 3.1 Выбор архитектуры информационной системы .....                         | 27 |
| 3.2 Выбор технологии веб-программирования информационной системы           | 28 |
| 3.3 Выбор системы управления базами данных информационной системы<br>..... | 30 |
| 3.4 Разработка физической модели данных информационной системы.....        | 32 |
| 3.5 Описание функциональности информационной системы.....                  | 33 |
| 3.6 Тестирование информационной системы.....                               | 35 |
| Заключение .....   | 38 |
| Список используемой литературы .....                                       | 39 |

## Введение

В настоящее время информационные системы занимают значимое место в жизни людей. Самые первые из них были созданы еще в 50-х годах прошлого столетия и осуществляли в основном арифметические расчеты, незначительно сокращая издержки производства и затраты времени.

Развитие информационных систем не стояло на месте, продвигаясь в ногу со временем и деловыми потребностями человека. К банальным возможностям расчета зарплат добавились возможности анализировать информацию, упрощая процесс принятия решений для управленческого персонала. Также, с каждым годом степень автоматизации систем увеличивалась, позволяя все сильнее наращивать производственные показатели предприятий их использующих [1], [2].

В современных условиях, человек вынужден работать с гигантскими объемами информации. В связи с этим разработка программных продуктов, служащих для автоматизированного учета, весьма актуальна. Системы обязаны представлять собой мощные средства, способные обрабатывать гигантские потоки данных высокой структурной сложности за минимум затраченного времени, обеспечивая дружественный диалог с пользователем.

Объектом исследования ВКР является бизнес-процесс букинга мобильных рабочих мест.

Предметом исследования ВКР является автоматизация букинга мобильных рабочих мест.

Целью выпускной квалификационной работы является разработка информационной системы букинга мобильных рабочих мест.

Для достижения поставленной цели необходимо решить следующие задачи:

- проанализировать выбранную предметную область;
- описать существующий бизнес-процесс;

- рассмотреть предоставляемые аналоги;
- разработать логическую модель;
- разработать физическую модель;
- выбрать технологии разработки;
- выбрать СУБД;
- разработать информационную систему;
- провести тестирование.

Практическая значимость работы заключается в разработке информационной системы букинга мобильных рабочих мест.

## Глава 1 Функциональное моделирование предметной области

### 1.1 Технико-экономическая характеристика ООО «Haulmont»

Объектом для автоматизации является компания «Haulmont». Haulmont российская компания по разработке корпоративного программного обеспечения. Основана в 2008 году.

Деятельность данной компании направлена на разработку отраслевых продуктов. Компания предоставляет следующие виды услуг.

Рассмотрим основные виды деятельности компании:

- разработка уникальных продуктов для отдельных заказчиков;
- разработка популярных отраслевых продуктов;
- развитие open-source фреймворка Jmix.

Рассмотрим организационную структуру предприятия. Помимо высшей административной должности, в компании работает пять отделов: отдел разработки платформы, отдел заказной разработки, бухгалтерия, отдел кадров, отдел маркетинга.

Схема организационной структуры изображена на рисунке 1.



Рисунок 1 – Схема организационной структуры ООО «Haulmont»



Для большего понимания структуры знакомимся с данной схемой более подробно.

Генеральный директор - управляет компанией, анализирует и выбирает определяет направления развития и технологий поддержания бизнеса. Помимо этого, занимает пост лидера для своих подчиненных, принимает окончательное решение требуемых задач и достижений поставленных перед подразделением целей.

Отдел разработки платформы – разработка и поддержание разрабатываемого продукта для дальнейшей продажи и использования в проектах.

Отдел заказной разработки – создание уникального решения, согласно потребностям клиента.

Бухгалтерия - контролирует движение денежных средств на банковских счетах предприятия; отражает в документах бухгалтерского учета движение денежных средств; обеспечивает сохранность документов бухгалтерского учета, контроль проведения хозяйственных операций, с соблюдением технологии обработки бухгалтерской информации.

Отдел кадров – отдел, занимающийся отбором и наймом персонала требуемой квалификации. Главная задача данного отдела – привлечь новых людей в компанию. Для этого отдел составляет и выставляет вакансии. После отклика человека на вакансию, отдел проводит собеседование и тестирования и исходя из результатов оглашают решение о найме. Помимо этого, отдел выполняет учет работы сотрудников, формирует расписания, составляет график отпусков, создает планы для повышения квалификации работников и ведет трудовой учет.

Отдел маркетинга - взаимодействует с партнерами по поводу рекламных акций; разрабатывают и продвигают группы в социальных сетях; сотрудничают с веб-разработчиком для создания сайта; разрабатывают рекламные кампании контекстной рекламы; обратная связь с клиентами.

Организация имеет огромный штат персонала, некоторая часть которого работает на удаленном режиме работы. За такими сотрудниками нет закреплённого рабочего места, однако, иногда сотрудникам необходимо появляться в офисе для решения рабочих вопросов. Для этого компания создала мобильные рабочие места, которое зарезервировать может любой сотрудник в электронном файле.

## **1.2 Анализ бизнес-процессов ООО «Haulmont»**

### **1.2.1 Выбор технологии моделирования бизнес-процессов**

Чтобы определились существующие процессы, протекаемые в компании необходимо выбрать технологии для моделирования бизнес-процессов.

Согласно источнику Elma «BPMN (The Business Process Modeling Notation) — это новый стандарт для моделирования бизнес процессов и сетевых услуг, который впервые был выпущен BPMI Notation Working Group в мае 2004 года [15].

Нотация BPMN описывает условные обозначения для отображения бизнес-процессов в виде диаграмм бизнес-процессов. BPMN ориентирована как на технических специалистов (разработчиков, ответственных за реализацию процессов), так и на бизнес-пользователей (бизнес-аналитиков, создающих и улучшающих процессы) и менеджеров, следящих за процессами и управляющих ими. Следовательно, BPMN призвана служить связующим звеном между фазой дизайна бизнес-процесса и фазой его реализации. Для этого язык использует базовый набор интуитивно понятных элементов, которые позволяют определять сложные семантические конструкции» [8].

Рассмотрев и выбрав требуемую технологию моделирования, приступим к разработке и анализу функциональной модели бизнес-процесса «КАК ЕСТЬ».

### 1.2.2 Разработка и анализ модели бизнес-процесса «КАК ЕСТЬ»

Функциональная модель букинга мобильных рабочих мест представлена в виде диаграммы в нотации BPMN. Для того, чтобы построить модели воспользуемся бесплатным сервисом от компании «Elma» BPMN.Studio [25].

По утверждению электронного ресурса BPMN.Studio «проект задуман как бесплатный облачный инструмент для моделирования бизнес-процессов и быстрого формирования пакета документов. Часто до автоматизации процессов необходимо их быстро описать без лишних деталей и обсудить с участниками проекта» [24].

Однако необходимо помнить о том, что анализ «КАК ЕСТЬ» изображает то, что нужно улучшить [26].

Ниже, на рисунке 2 представлена диаграмма бизнес-процесса букинга мобильных рабочих мест «КАК ЕСТЬ» [3], [4].

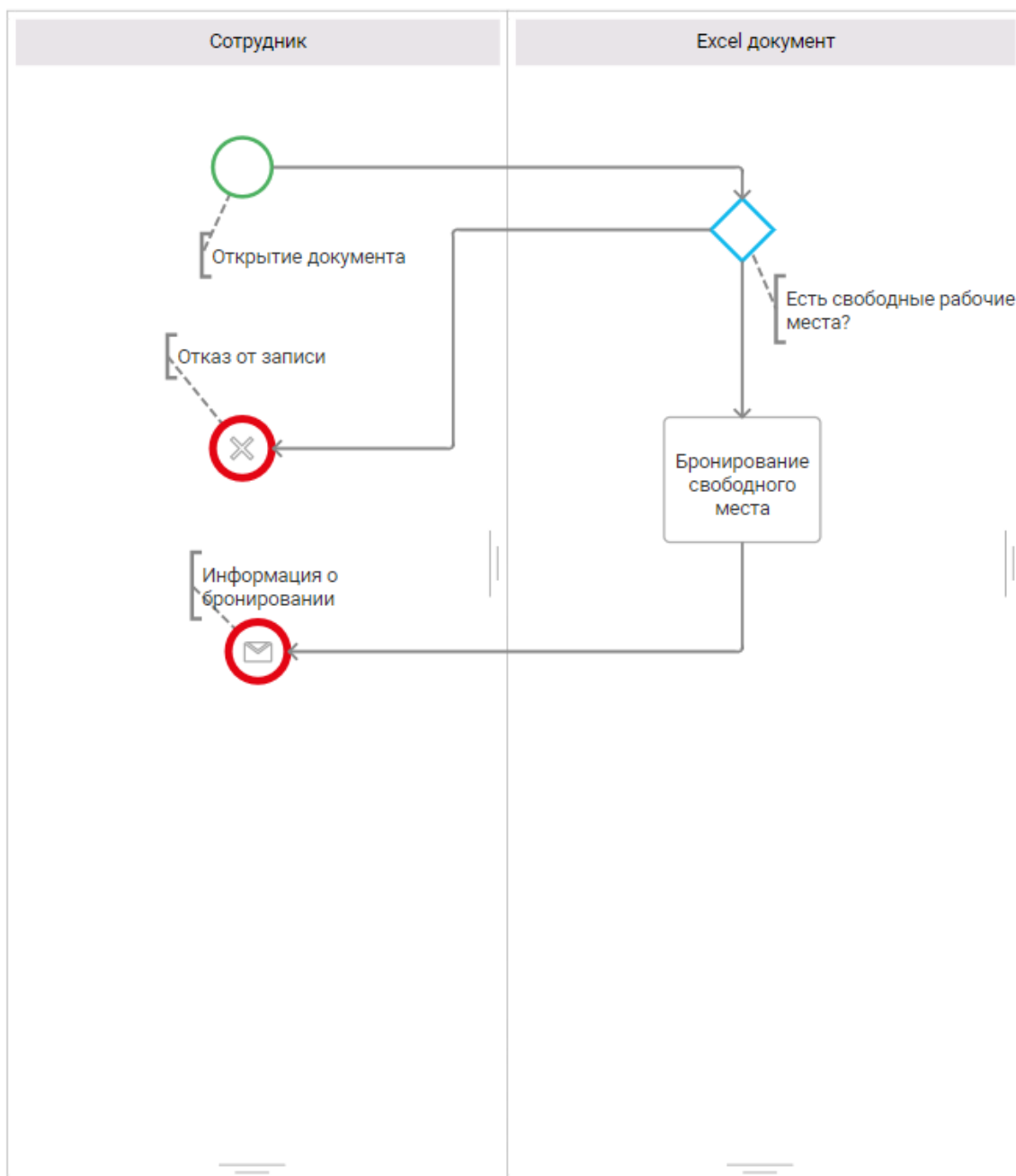


Рисунок 2 – Диаграмма бизнес-процесса букинга мобильных рабочих мест «КАК ЕСТЬ»

Представленная выше диаграмма изображена от лица сотрудника предприятия. Исходя из анализа, были выявлено то, что отсутствует отображение даты и времени бронирования, а также возможности забронировать рабочее место за определенную дату [5].

Улучшить существующий бизнес-процесс возможно за счет внедрения информационной системы, за счет которой будет производиться автоматизация [30].

### 1.3 Анализ существующих технологий букинга мобильных рабочих мест

Сегодня множество компаний предлагают свои решения по услугам букинга мобильных рабочих мест в офисе. Их продукты могут быть самостоятельными приложениями или же онлайн-сервисом [31], [32].

Рассмотрим самые популярные решения, существующие на данный момент. Результаты представлены в таблице 1.

Таблица 1 – Сравнительная оценка аналогов программного обеспечения

| Требования                                   | Office Flexispace | Merusoft White label | ROOMz |
|--|-------------------|----------------------|-------|
| 1  | 2                 | 3                    | 4     |
| Интуитивно-понятный интерфейс                | +                 | +                    | +     |
| Распределение ролей                          | -                 | +                    | +     |
| Создание пользователей                       | +                 | +                    | +     |
| Возможность развернуть систему на сервере    | -                 | -                    | +     |
| Отображение результатов                      | +                 | +                    | +     |
| Низкая цена                                  | +                 | -                    | -     |
| Возможность изменения программистом компании | -                 | -                    | -     |

Представленные информационные системы, предназначенные для автоматизации процесса бронирования, имеют схожий функционал. Каждая система имеет как плюсы, так и минусы.

Проведя анализ систем, было принято решение разработать информационную систему, которая будет соответствовать существующим требованиям.

#### **1.4 Разработка требований к проектируемой информационной системе**

Перед началом разработки информационной системы необходимо установить чёткие требования. Для этих целей будет использована методология FURPS+ [16], [17].

Согласно источнику Studme.org «FURPS-I— это одна из распространенных классификаций требований к информационным системам. Название произошло от первых букв слов Functionality, Usability, Reliability, Performance, Supportability (Функциональность, Удобство использования, Надежность, Производительность, Поддерживаемость). Впервые классификация уровней требований была предложена в 1990-е гг. Робертом Грейди, сотрудником компании Hewlett Packard» [12].

Приступим к составлению требований:

- Функциональность (Functionality). Данный фактор оценивается представляемыми функциями программы, её возможностями и безопасностью;
- Удобство использования (Usability). Фактор, который оценивается простотой и комфортом взаимодействием с информационной системой;
- Надежность (Reliability). Фактор, оценка которого осуществляется в устойчивости информационной системы, а также ее быстрой способности самовосстановления в результате выхода из строя из-за непредвиденных ситуаций;

- Производительность (Performance). Данный фактор оценивается скоростью работы программы, потреблением используемых ресурсов и временем отклика;
- Поддерживаемость (Supportability), которая сочетает в себе возможность расширения программы, адаптивность, удобство обслуживания.

Составив необходимые требования, приступим к разработке и визуализации модели бизнес-процесса «КАК ДОЛЖНО БЫТЬ» [28].

### **1.5 Разработка модели бизнес-процесса «КАК ДОЛЖНО БЫТЬ»**

При реинжиниринге исследуемого процесса был произведен анализ, результатом которого является модель «Как должно быть». Модель «Как должно быть» представляет собой модернизированный и усовершенствованный бизнес-процесс.

На основе данной модели производится дальнейшее проектирование информационной системы. Теперь пользователь будет взаимодействовать с информационной системой, а не документом. Теперь для доступа к информационной системе, администратор, выбранный начальством, будет следить за работоспособностью системы, а также создавать логины и пароль для новых сотрудников компании. Сам же пользователь, после процесса авторизации, получит полный список всех забронированных мест в офисе и исходя из этих данных выберет нужное.

На рисунке 3 представлена диаграмма бизнес-процесса.

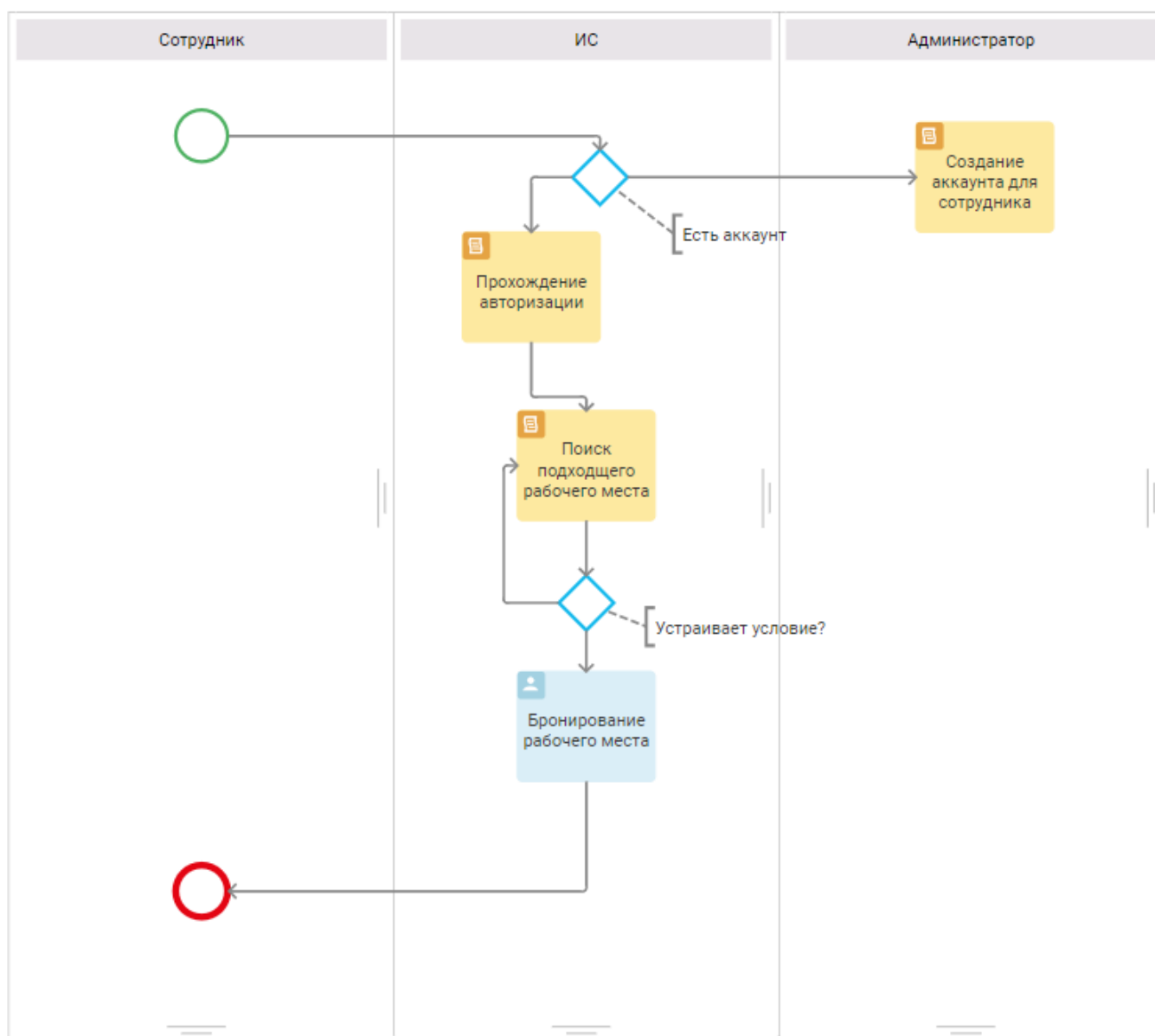


Рисунок 3 – Диаграмма бизнес-процесса бронирования мобильных рабочих мест «КАК ДОЛЖНО БЫТЬ»

Полученная в результате реинжиниринга диаграмма демонстрирует улучшение бизнес-процесса за счет разработки и внедрения информационной системы бронирования рабочих мест в офисе, в соответствии с составленными требованиями.



Завершив построение модели бизнес-процесса «КАК ДОЛЖНО БЫТЬ», перейдем к логическому проектированию автоматизированной информационной системы бронирования мобильных рабочих мест.

#### Выводы по главе 1

В результатах выполнения первой главы был произведен анализ и представлена характеристика компании «Haulmont». Было выполнено построение модели «КАК ЕСТЬ» с описанием существующего бизнес-процесса и модель «КАК ДОЛЖНО БЫТЬ» с представленным решением устранения существующих проблем.

Помимо этого, выполнен анализ существующих решений по бронированию мест в офисе. Далее были сформированы требования к проектируемой информационной системе.

## **Глава 2 Логическое проектирование автоматизированной информационной системы букинга мобильных рабочих мест**

### **2.1 Выбор технологии логического моделирования информационной системы**

Перед выбором технологии необходимо понять, с какой целью применяется логическое моделирование. Согласно источнику Ngpedia.ru «Логическое моделирование применяется для качественного описания процесса и его развития. В этом методе создают сложные сценарии, направленные на увязку логической последовательности и значимости событий, факторов, воздействующих на рыночные процессы» [13]. Процесс моделирования будет осуществлен на основании модели бизнес-процесса «КАК ДОЛЖНО БЫТЬ», что была описана ранее.

Логическое моделирование информационной системы будет осуществлено при помощи с использованием языка моделирования UML. В учебном пособии «Проектирование ИС» заявлено, что «UML обеспечивает поддержку всех этапов жизненного цикла ИС и предоставляет для этих целей ряд графических средств – диаграмм.

На этапе создания логической модели ИС описание требований к системе задается в виде модели и описания системных прецедентов, а предварительное проектирование осуществляется с использованием диаграмм классов, диаграмм последовательностей и диаграмм состояний».

Дальнейшим действием следует выбрать CASE-средство. Произведем выбор, путем сравнения, из существующих популярный решений: StarUML, MS Visio, IBM Rational Rose.

Согласно источнику Citforum Rational Rose «CASE-средство фирмы Rational Software Corporation, предназначенное для автоматизации этапов анализа и проектирования ПО, а также для генерации кодов на различных языках и выпуска проектной документации» [14].

По утверждению источника Tadviser.ru «MS Visio — это векторный графический редактор, позволяющий создавать и редактировать блок-схемы и диаграммы, т.е. решать задачи, с которыми регулярно сталкиваются сотрудники самых разных подразделений любой компании» [27].

Источник Russianblogs утверждает, что «StarUML (сокращенно SU) - это инструмент для создания диаграмм классов UML и автоматической генерации «кода-заглушки» Java. SU также может выполнять обратное проектирование JAVA для создания соответствующих диаграмм UML» [11].

Для дальнейшего сравнения существующих CASE-средств воспользуемся таблицей 2.

Таблица 2 - Анализ CASE-средств визуального моделирования

| Требования                         | StarUML | MS Visio | IBM Rational Rose |
|------------------------------------|---------|----------|-------------------|
| Функциональная полнота             | +       | +        | +                 |
| Простота использования             | +       | +        | +                 |
| Поддержка обратного инжиниринга    | +       | +        | +                 |
| Поддержка генерации исходного кода | +       | -        | +                 |
| Бесплатное использование           | +       | -        | -                 |
| Итого                              | 5       | 3        | 4                 |

В результате сравнения CASE-средств была выбрано средство StarUML.

Дальнейшим действием приступим к разработке логической модели бронирования.

## 2.2 Логическая модель информационной системы и ее описание

Перед непосредственной разработкой логической модели для более точного проектирования необходимо выбрать и создать основные диаграммы. В качестве диаграмм были следующие: диаграмма вариантов использования, классов и последовательностей. Данные диаграммы отобразят процессы, протекающие в системе.

### 2.2.1 Диаграмма вариантов использования информационной системы букинга

Согласно электронному ресурсу Wikipedia «Диаграмма вариантов использования в UML — диаграмма, отражающая отношения между акторами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

Основное назначение диаграммы — описание функциональности и поведения, позволяющее заказчику, конечному пользователю и разработчику совместно обсуждать проектируемую или существующую систему» [10].

Выявим действующих акторов, проанализировав модель бизнес-процесса «КАК ДОЛЖНО БЫТЬ».

В результате были выявлены акторы Сотрудник и Администратор. Рассмотрим их более подробно:

- Сотрудник – работник компании, желающий забронировать свободное рабочее место;
- Администратор – работник компании, следящий за работоспособностью системы и созданием аккаунтов для новых сотрудников.

Созданная диаграмма вариантов использования изображена на рисунке 4.

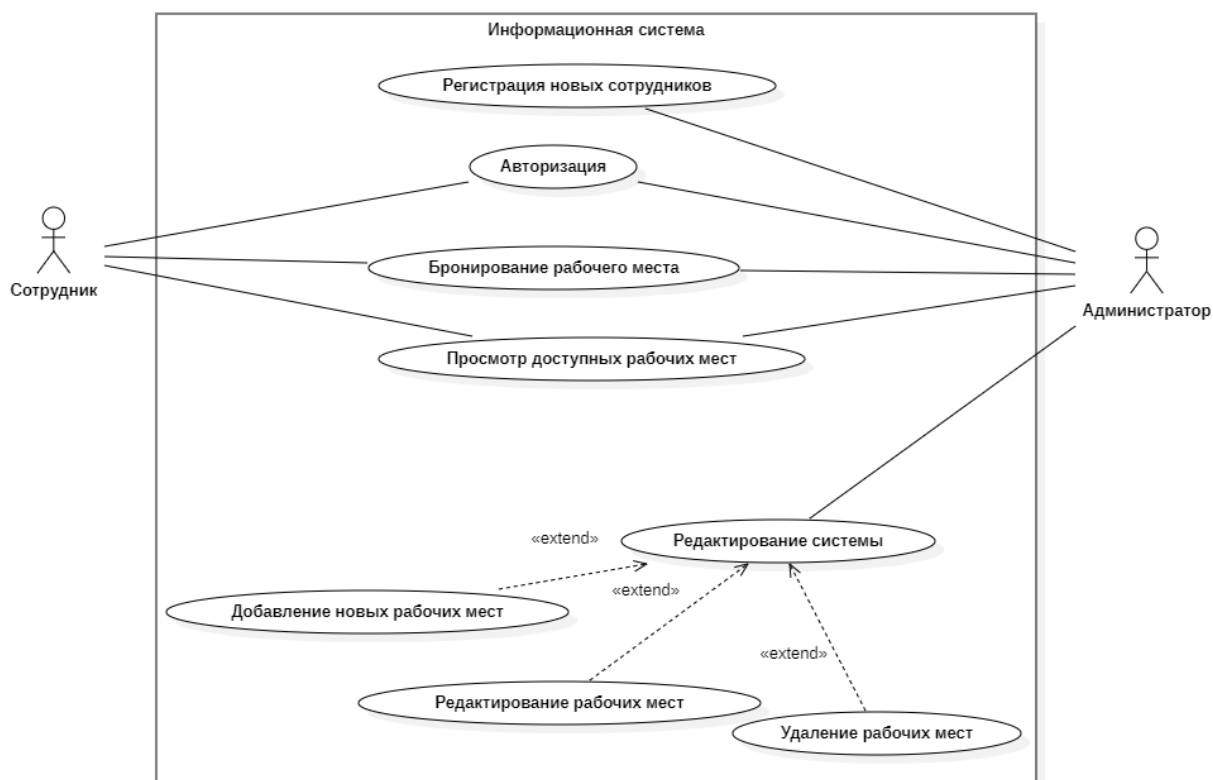


Рисунок 4 – Диаграмма вариантов использования

Рассмотрим более подробно существующие прецеденты. Они представлены в таблице 4.

Таблица 4 – Описание существующих прецедентов

| Прецедент                       | Описание   |
|---------------------------------|--|
| Регистрация новых сотрудников   | Администратор создает логин и пароль для новых сотрудников                                       |
| Авторизация                     | Авторизация пользователя информационной системы  |
| Бронирование рабочего места     | Пользователь имеет возможность забронировать рабочее место на нужную ему дату                    |
| Просмотр доступных рабочих мест | Пользователь может посмотреть список свободных и занятых рабочих мест                            |
| Редактирование системы          | Администратор имеет возможность редактирования данных системы в соответствии с требуемыми целями |
| Добавление новых рабочих мест   | Администратор добавляет новое рабочее место при создании новых рабочих мест                      |

| Прецедент                   | Описание   |
|-----------------------------|--|
| Редактирование рабочих мест | Администратор изменяет данные существующего рабочего места |
| Удаление рабочих мест       | Администратор удаляет рабочее место                        |

Далее приступим к разработке диаграммы классов.

### 2.2.2 Разработка диаграммы классов информационной системы

Для представления внутренней структуры информационной системы создадим диаграмму классов. Диаграмма классов UML отображает структуру разрабатываемой системы. Помимо этого, она изображает взаимодействие и отношения объектов, описывает классы и их атрибуты. Разработанная диаграмма изображена на рисунке 5.



Рисунок 5 – Диаграмма классов

Однако одной диаграммы недостаточно для понимания существующих классов, поэтому при разработке создается спецификация классов. Спецификация классов — это подробное описание используемых классов ИС, представленные в виде таблицы. Данная спецификация представлена в таблице 5.

Таблица 5 – Спецификация классов

| Класс         | Описание                                     |
|---------------|--|
| Пользователь  | Пользователи информационной системы          |
| Администратор | Администрирует информационную систему        |
| Сотрудник     | Бронирует необходимое рабочее место          |
| Бронирование  | Объект «Бронирование» информационной системы |
| Рабочие места | Содержит информацию о рабочих местах         |

Завершив разработки, диаграмма в дальнейшем позволит, просто взглянув на нее, просмотреть и понять внутреннюю структуру информационной системы. Далее приступим к разработке диаграммы последовательности.

### 2.2.3 Разработка диаграммы последовательности информационной системы

Согласно источнику Creately.com «Диаграммы последовательностей, обычно используемые разработчиками, моделируют взаимодействия между объектами в едином сценарии использования. Они иллюстрируют, как различные части системы взаимодействуют друг с другом для выполнения функции, а также порядок, в котором происходит взаимодействие при выполнении конкретного случая использования. Проще говоря, диаграмма последовательности показывает различные части работы системы в “последовательности”, чтобы что-то сделать» [18].

Разработанные диаграммы представлены на рисунках 6 и 7.

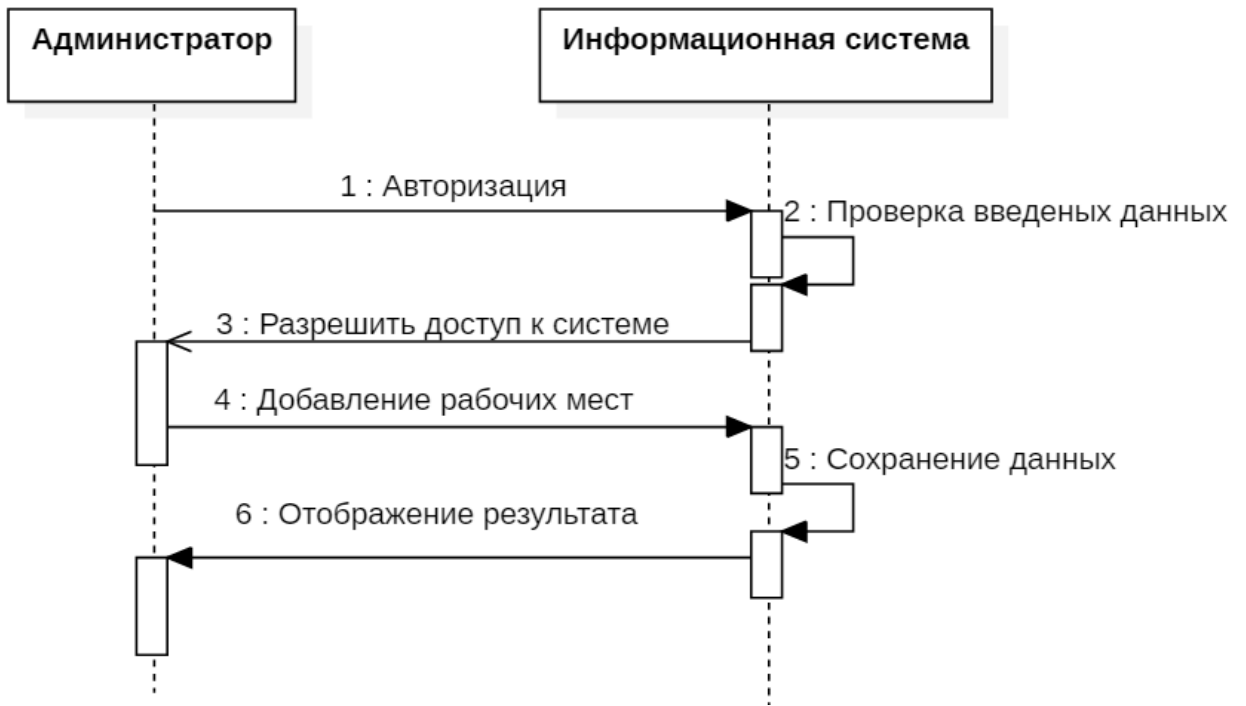


Рисунок 6 – Диаграмма последовательности процесса добавления рабочих мест

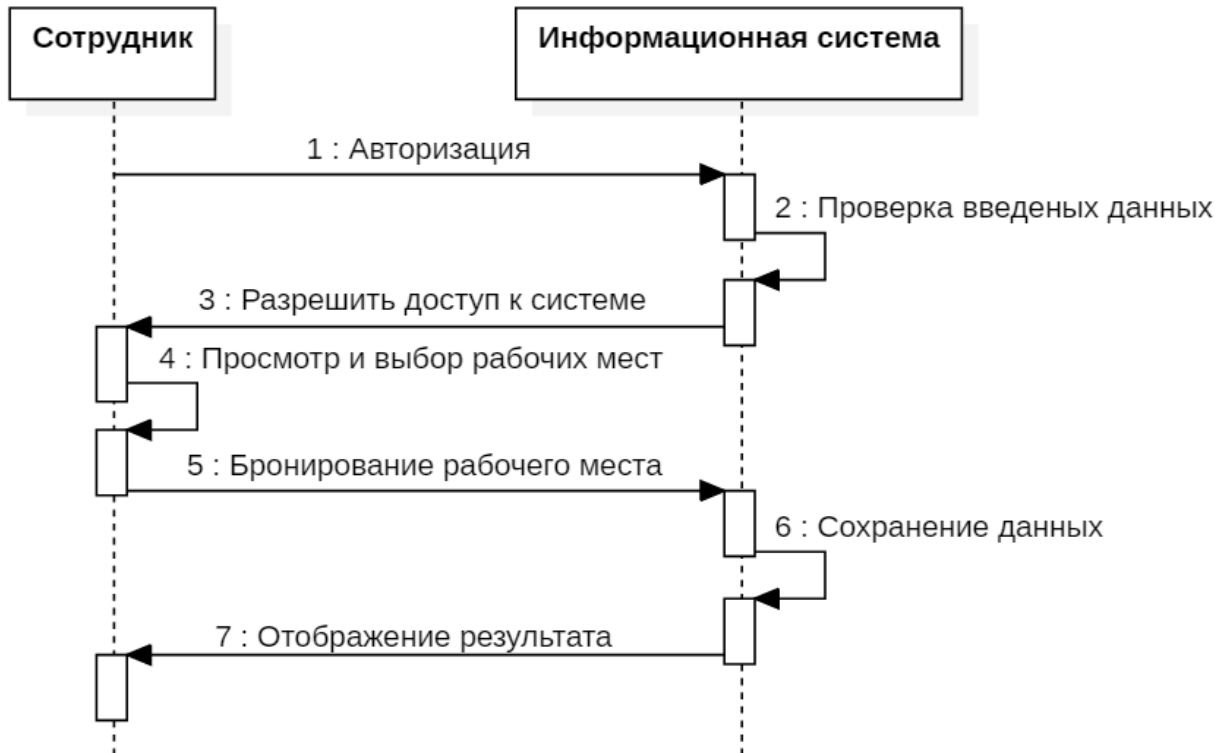


Рисунок 7 – Диаграмма последовательности процесса бронирования



Диаграммы последовательности помогают понять, почему существуют отношения на диаграмме классов. Далее приступим к разработке логической модели базы данных.

### 2.3 Разработка логической модели базы данных информационной системы и составление требований к аппаратно-программному обеспечению информационной системы

Согласно электронному ресурсу Inftis.ru «Логическая модель данных является начальным прототипом будущей базы данных. Она строится в терминах информационных единиц, но без привязки к конкретной СУБД. Более того, логическая модель данных необязательно должна быть выражена средствами именно реляционной модели данных. Основным средством разработки логической модели данных в настоящий момент являются различные варианты ER-диаграмм» [20].

Данная модель данных информационной системы представлена на рисунке 8.

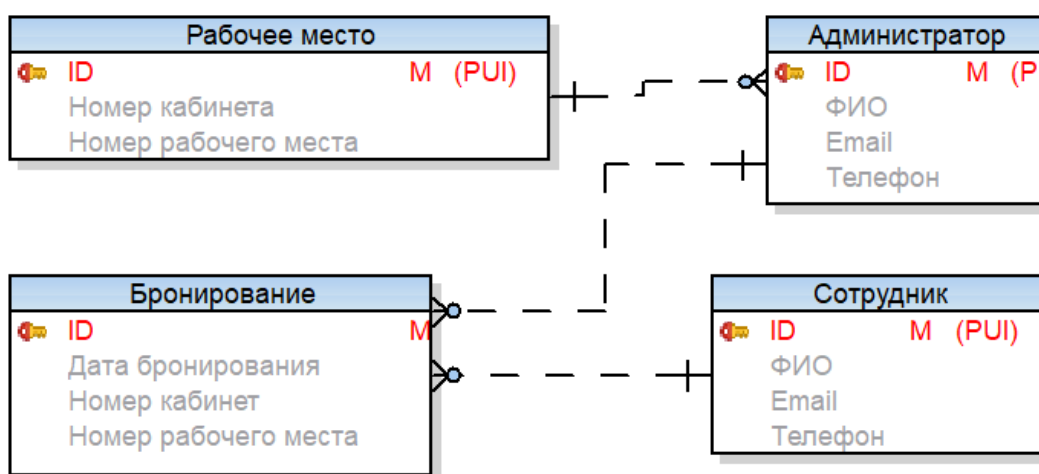


Рисунок 8 – Логическая модель данных информационной системы

Представленная выше логическая модель будет использоваться в качестве основы при дальнейшем проектировании информационной системы.

Далее перейдем к составлению требований к аппаратно-программному обеспечению информационной системы. Проанализировав схожие по функционалу информационные системы были составлены требования представленные ниже.

Требования к аппаратно-программному обеспечению, следующие:

- операционная система Windows или Linux.
- круглосуточный режим работы;
- двухзвездная клиент-серверная архитектура;
- поддержка СУБД;

Минимальные требования к персональным компьютерам пользователей:

- процессор: Intel Pentium III 600 МГц и выше;
- оперативная память 2 Gb RAM;
- объем дискового пространства не меньше 40 Gb;
- внешнее устройство хранения данных для осуществления регулярного резервного копирования

Минимальные требования к серверу:

- процессор Intel Pentium Celeron 2700 МГц и выше;
- оперативная память 1024 Мб RAM;
- объем дискового пространства не меньше 15 Гб.

Составив требования аппаратно-программному обеспечению, подведём итоги второй главы.

Выводы по главе 2

В ходе выполнения второй главы были выбраны технологии логического моделирования, составлена логическая модель с ее описанием, составлены диаграммы вариантов использования, классов и последовательностей информационной системы. Разработана логическая модель и составлены требования к аппаратно-программному обеспечению.

## **Глава 3 Физическое проектирование информационной системы компании**

### **3.1 Выбор архитектуры информационной системы**

Согласно автору Сергею Гаваге «Веб-приложение – это клиент-серверное приложение, в котором клиентом выступает браузер, а сервером – веб-сервер (в широком смысле).

Основная часть приложения, как правило, находится на стороне веб-сервера, который обрабатывает полученные запросы в соответствии с бизнес-логикой продукта и формирует ответ, отправляемый пользователю. На этом этапе в работу включается браузер, именно он преобразовывает полученный ответ от сервера в графический интерфейс, понятный пользователю.

Архитектура «клиент-сервер» определяет общие принципы организации взаимодействия в сети, где имеются серверы, узлы-поставщики некоторых специфичных функций (сервисов) и клиенты (потребители этих функций). Практические реализации такой архитектуры называются клиент-серверными технологиями» [33].

На электронном ресурсе [Sergeygavaga.gitbooks.io](https://Sergeygavaga.gitbooks.io) дано следующее понятие архитектуры: «Двухзвенная архитектура - распределение трех базовых компонентов между двумя узлами (клиентом и сервером). Двухзвенная архитектура используется в клиент-серверных системах, где сервер отвечает на клиентские запросы напрямую и в полном объеме» [6].

Двухзвенная архитектура представлена на рисунке 9.

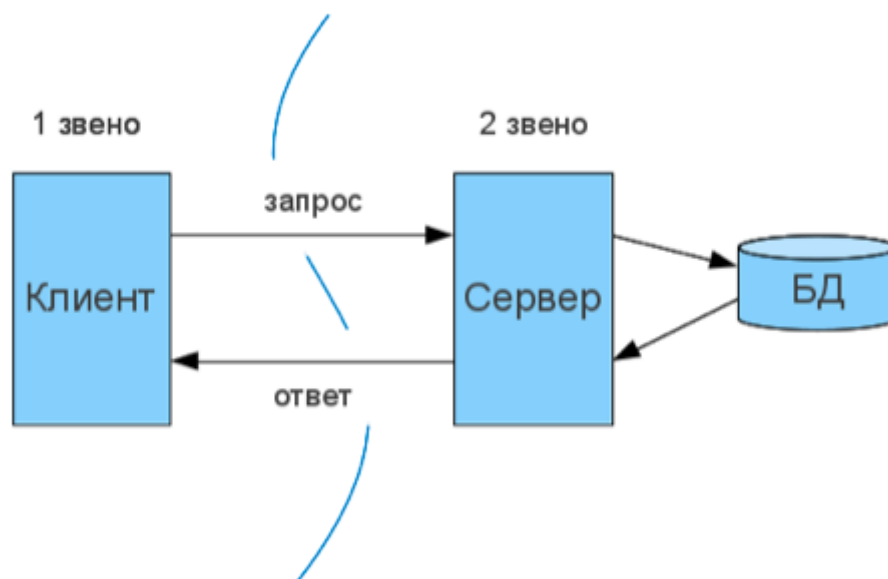


Рисунок 9 – Логическая модель данных информационной системы

Первое звено состоит из клиента, использующего браузер в качестве графического интерфейса и обрабатывающего ответ веб-сервера.

Второе звено состоит из сервера, взаимодействующего с базой данных и выполняющего необходимые функции.

После выбора архитектуры информационной системы будет сделан выбор технологий разработки информационной системы [34].

### **3.2 Выбор технологии веб-программирования информационной системы**

Проектируемая система предназначена для автоматизации процесса бронирования. Лучшим вариантом для презентации системы является веб-сайт. Существует несколько языков веб-программирования, каждый из которых имеет свои плюсы и минусы. Давайте рассмотрим некоторые из них.

Согласно источнику Prog.bobrodobro.ru «Java — объектно-ориентированный язык программирования. Язык Java потребовался для

создания интерактивных продуктов для сети Internet. Фактически, большинство архитектурных решений, принятых при создании Java, было продиктовано желанием предоставить синтаксис, сходный с С и С++. В Java используются практически идентичные соглашения для объявления переменных, передачи параметров, операторов и для управления потоком выполнением кода. В Java добавлены все хорошие черты С++» [27].

По утверждению интернет-источника Web-creator.ru «PHP — язык программирования, который наиболее распространён в сфере веб-разработки. В настоящее время PHP является одним из лидеров среди серверных языков программирования, применяющихся для создания динамических веб-сайтов и веб-приложений. Большая часть коробочных систем управления сайтами написана именно на PHP, язык поддерживается подавляющим большинством хостинг-провайдеров, Язык получил широкое распространение благодаря своей простоте, скорости, мультипарадигмальности, богатой функциональности и кроссплатформенности» [21].

По заверению источника Web-creator «Ruby (ruby — рубин, руби) — интерпретируемый мультипарадигмальный язык программирования: динамический, объектно-ориентированный, рефлексивный, императивный, функциональный. Активно используется в веб-разработке, в системном администрировании и в работе операционных систем (Mac OS X, Linux, BSD)» [22].

Результат анализа рассматриваемых технологий представлен в таблице 6.

Таблица 6 – Сравнительный анализ технологий веб-программирования

| Критерий                                | Java | PHP | Ruby |
|---|------|-----|------|
| Низкая стоимость разработки             | +    | +   | +    |
| Взаимодействие с СУБД                   | +    | +   | -    |
| Актуальность                            | +    | +   | -    |
| Большое количество функций безопасности | +    | -   | -    |
| Итого:                                  | 4    | 3   | 1    |

Проведенный анализ показал, что лучший язык для разрабатываемой системы является Java. Далее перейдем к выбору фреймворка разработки.

По заверению электронного ресурса Web-creator «Фреймворки — это программные продукты, которые упрощают создание и поддержку технически сложных или нагруженных проектов. Фреймворк, как правило, содержит только базовые программные модули, а все специфичные для проекта компоненты реализуются разработчиком на их основе. Тем самым достигается не только высокая скорость разработки, но и большая производительность и надёжность решений» [19]. Рассмотрим существующие фреймворки для языка Java.

Vaadin — «свободно распространяемый фреймворк для создания RIA-веб-приложений, разрабатываемый одноимённой финской компанией. В отличие от библиотек на Javascript и специфических плагинов для браузеров, Vaadin предлагает сервер-ориентированную архитектуру, базирующуюся на Java Enterprise Edition» [29].

Завершив выбор требуемых технологий, перейдем к выбору систем управления базами данных.

### **3.3 Выбор системы управления базами данных информационной системы**

Система управления базой данных – это совокупность программных и языковых средств управления данными. В качестве СУБД для разрабатываемого приложения будет использована MariaDB.

MariaDB — ответвление реляционной СУБД MySQL, разрабатываемое сообществом под лицензией GPL. MariaDB полностью совместима с приложениями, использующими MySQL, а переход на эту СУБД оправдан тем, что MySQL уже не так активно развивается.

Однако следует учитывать то, что установка системы управления базой данных невозможна без веб-сервера. Поэтому далее необходимо провести анализ существующих веб-серверов и выбрать наиболее подходящий.

Согласно официальному сайту Nginx «это HTTP-сервер и обратный прокси-сервер, почтовый прокси-сервер, а также TCP/UDP прокси-сервер общего назначения, изначально написанный Игорем Сысоевым. Уже длительное время он обслуживает серверы многих высоконагруженных российских сайтов, таких как Яндекс, Mail.Ru, ВКонтакте и Рамблер. Согласно статистике Netcraft nginx обслуживал или проксировал 21.79% самых нагруженных сайтов в апреле 2022 года» [9].

Далее рассмотрим веб-сервер AOLServer. AOLServer «это веб-сервер разработанный в NaviSoft как «NaviServer», но выкупленный в 1995 году AOL, которая начала распространять его как freeware. В 1999 году версия 3.0 AOLserver'a выпущена как открытое программное обеспечение. Это многопоточный веб-сервер со встроенным скриптовым языком Tcl. Часто используется совместно с CMS/CMF OpenACS написанной на Tcl/XOTcl. Ключевые особенности этого веб-сервера: многопоточность (которая появилась в Apache только начиная с версии 2.0), встроенный скриптовый язык (в роли которого выступает Tcl) и поддержка постоянного соединения с SQL-сервером» [23].

По утверждению источника «Eternalhost.net» Apache HTTP-сервер это «свободное программное обеспечение для размещения веб-сервера. Он хорошо показывает себя в работе с масштабными проектами, поэтому заслуженно считается одним из самых популярных веб-серверов. Кроме того, Apache очень гибок в плане настройки, что даёт возможность реализовать все особенности размещаемого веб-ресурса. Ядро Apache разработано Apache Software Foundation на языке C. Основные функции — обработка конфигурационных файлов, протокол HTTP/HTTPS и загрузка модулей. Ядро может работать без модулей, но будет иметь ограниченный функционал» [24].

Проанализировав и выбрав как СУБД, так и веб сервер перейдем к разработке физической модели информационной системы.

### 3.4 Разработка физической модели данных информационной системы

Для грамотного составления данных, необходимые СУБД разработаем физическую модель данных. Данная модель изображает все связи, свойства и столбцы базы данных.

Разработанная физическая модель данных информационной системы представлена на рисунке 10.

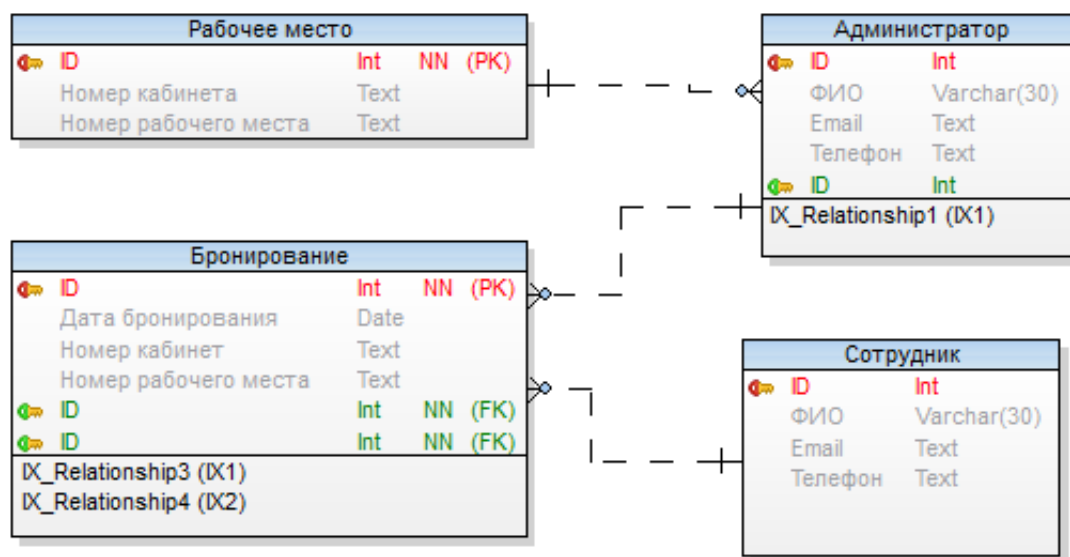


Рисунок 10 – Физическая модель данных информационной системы

Благодаря спроектированной ранее логической модели, была разработана физическая модель данных. Далее перейдем к описанию функциональности информационной системы.



### 3.5 Описание функциональности информационной системы

Установив информационную систему на веб-сервер и перейдя по заданному на сервере адресу неавторизованному пользователю, предоставляется пройти авторизацию. Для получения логина и пароля необходимо обратиться к администратору. За визуализацию пользовательского интерфейса отвечает файл Login.java (рисунок 11).

```
@Route("login")
public class LoginView extends Composite<LoginOverlay> {

    public LoginView() {
        getContent().setOpened(true);
        getContent().setAction("login");
    }
}
```

Рисунок 11 – Работа с файлом Login.java

Завершив процедуру авторизации, пользователю предоставляется возможность просмотреть свободные и занятые рабочие места. Выполнением данных функций отвечает файл Booking.java (рисунок 16), который формирует форму ввода данных.

```
@Route("user")
@RolesAllowed("USER")
public class Booking extends VerticalLayout {

    public Booking(BookingService service) {
        var crud = new GridCrud<>(Book.class, service);
        crud.getGrid().setColumns("fullName", "bookedDate", "workplace");
        crud.getCrudFormFactory().setVisibleProperties("fullName", "bookedDate", "workplace");
        crud.setAddOperationVisible(false);
        crud.getCrudLayout().addToolbarComponent(new RouterLink("Booking", NewView.class));

        add(
            new H1("Booking mobile jobs"),
            new H2("Choose a free workplace"),
            crud
        );
    }
}
```

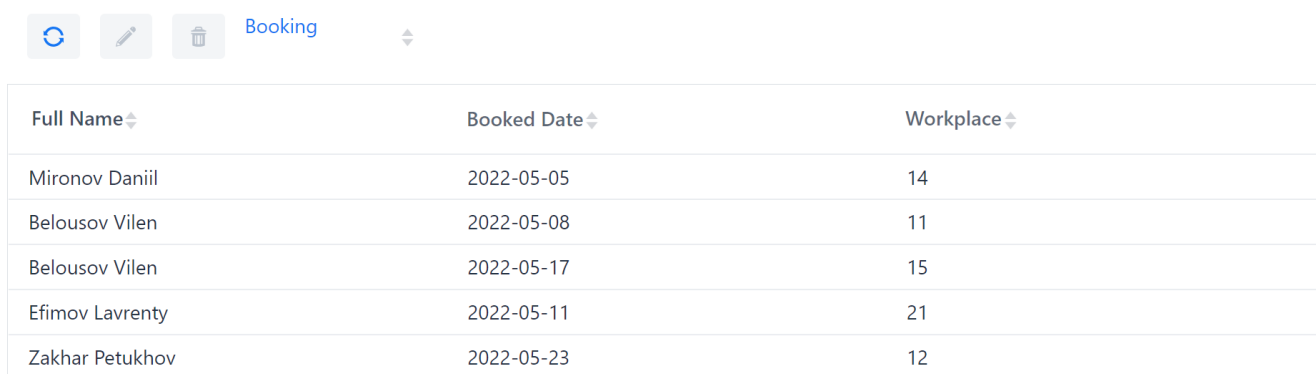
Рисунок 11 – Работа с файлом Booking.java

Данные приложения выполняют требуемый функционал для приложения, создают защиту от несанкционированного доступа путем добавления системы авторизации, предоставляют возможность просмотреть свободные и занятые рабочие места, забронировать необходимое рабочее место в удобную дату.

Внешний вид страниц приложения изображен на рисунках 12-13. Страница бронирования рабочих мест изображена на рисунке 12. Форма включает поля «Full Name», «Booked Date», и «Workspace». Выше списка бронированных рабочих мест располагаются кнопки «обновления», «редактирование», «удаление» и «Booking».

## Booking mobile jobs

### Choose a free workplace



| Full Name       | Booked Date | Workplace |
|-----------------|-------------|-----------|
| Mironov Daniil  | 2022-05-05  | 14        |
| Belousov Vilen  | 2022-05-08  | 11        |
| Belousov Vilen  | 2022-05-17  | 15        |
| Efimov Lavrenty | 2022-05-11  | 21        |
| Zakhar Petukhov | 2022-05-23  | 12        |

Рисунок 12 – Внешний вид формы бронирования рабочих мест


При нажатии на кнопку «Booking» перед пользователем появится окно бронирования рабочего места. Окно включает в себя поля «Full Name», «Booked Date», и «Workspace», а также кнопку «Save». Данное окно изображено на рисунке 14.

# Fill out the form

Full Name

Mironov Daniil

Booked Date

31.05.2022 

Workplace

19

Save

Рисунок 13 – Окно бронирования рабочих мест

В данном окне бронирования сотрудник заполняет указывает свое имя, требуемую ему дату, а также нужное рабочее место. После заполнения всех полей пользователю и нажатия кнопки «Save», введенные ранее данные, информационная система сохранит и представит их в таблице забронированных рабочих мест.

## 3.6 Тестирование информационной системы

Тестирование систем – важный этап производства ПО, направленный на детальное исследование программного кода и выявление ошибок в работе систем.

Чтобы правильно протестировать приложение сперва необходимо составить требования, по которым будет оцениваться функционал

разработанного приложения. Рассмотрим каждый из составленных требований, представленных ниже:

а) объект тестирования:

1) в качестве объекта для тестирования будет использоваться разработанная информационная система бронирования;

б) цель тестирования:

2) целями тестирования являются следующие:

- проверка работоспособности программы,
- проверка отказоустойчивости системы,
- проверка адаптивности приложения,
- проверка установки приложения на веб-сервер,
- выявление ошибок работоспособности приложения;

в) методика тестирования:

3) тестирование ПО ИСУ проводится по методу бета-тестирования (beta testing).

Согласно источнику Education-wiki «Бета-тестирование — это тип тестирования программного обеспечения, которое выполняется конечными пользователями в реальной среде перед официальным выпуском любого программного продукта для коммерческих пользователей с целью обеспечения качества, достоверности и надежности программного обеспечения. Бета-тестирование часто называют приемочным тестированием и считается последним этапом тестирования перед выпуском программного обеспечения. Для бета-тестирования не существует специальной тестовой / лабораторной среды, вместо этого программное обеспечение предоставляется ограниченному числу пользователей (конечных пользователей) для проведения тестирования в реальной среде» [7].

Результаты тестирования представлены ниже в таблице 7.

Таблица 7 – Результаты тестирования разработанного приложения

| № этапа | Описание   | Результат тестирования  | Рекомендации   |
|---------|--|---|--|
| 1       | Установка программного продукта                  | Установка прошла успешно  |  |
| 2       | Визуализация системы                             | Интерфейс полностью исправен                                    |  |
| 3       | Проверка адаптивности системы                    | Приложение полностью подстраивается под устройство пользователя |  |
| 4       | Проверка отказоустойчивости приложения           | Нагруженное приложение полностью выполняет свои функции.        |  |
| 5       | проверка работоспособности программного продукта | Все функции системы прошли проверку                             | Добавить счетчик количества свободных и занятых рабочих мест |

В результате тестирования были получены ожидаемые результаты, поэтому с уверенностью можно утверждать об успешном прохождении тестирования разработанного программного продукта.

### Выводы по главе 3

В ходе работы по третьей главе, были выбрана архитектура информационной системы. Был произведен выбор технологий веб-программирования, выбрана систему управления базами данных информационной системы. Разработана физическая модель данных, было составлено описание функциональности информационной системы, а также произведено её тестирование.

## Заключение

В ходе выполнения выпускной квалификационной работы для компании «Холмонт БР» был произведен анализ предметной области, который показал, что существующий процесс букинга мобильных рабочих мест имеет некоторые недостатки. Поэтому было принято решение разработать автоматизированную информационную систему бронирования рабочих мест.

В результате выполнения поставленной задачи было произведено проектирование автоматизированной информационной системы. Также были проанализированы требования к разрабатываемой системе букинга мобильных рабочих мест и выявлены функции, которые она должна выполнять. Были построены логическая и физическая модели данных, что позволило проверить логику информационной системы на наличие логических ошибок.

Итогом выпускной квалификационной работы является автоматизированная информационная система букинга мобильных рабочих мест. Информационная система выполняет следующие функции:

- авторизация пользователей;
- просмотр занятых и доступных рабочих мест;
- бронирование мобильного рабочего места;
- добавление новых рабочих мест администратором системы.

В результате тестирования разработанной информационной системы было продемонстрировано, что система выполняет все поставленные задачи в соответствии со сформированными требованиями, демонстрируя корректную работу.

При необходимости данную информационную систему можно усовершенствовать, благодаря фреймворку, имеющему множество возможностей.

## Список используемой литературы

1. Бодров О.А. Предметно-ориентированные экономические информационные системы: учеб. пособие / О.А. Бодров. – М.: Гор. линияТелеком, 2013. – 244 с.
2. Заботина Н.Н. Проектирование информационных систем – М.: ДРОФА, 2013. – 336 с.
3. Николаев В.Н. Метод и устройство управления информационно-вычислительными ресурсами вида "клиент-сервер". 2013. №1. С. 208-211.а
4. Олейник П.П. Корпоративные информационные системы: Учебник для вузов. Стандарт третьего поколения / П.П. Олейник. – СПб.: Питер, 2012. – 176 с.
5. Чистов Д.В. Проектирование информационных систем. Учебник и практикум / Д.В. Чистов, П.П. Мельников, А.В. Золотарюк, Н.Б. Ничепорук. – М.: Юрайт, 2016. – 260 с.
6. Архитектура клиент-сервер [Электронный ресурс]. URL: <https://sergeygavaga.gitbooks.io/kurs-lektsii-testirovanie-programmnogo-obespecheni/content/lektsiya-6-ch1-arhitektura-klient-server.html> (дата обращения: 20.02.22).
7. Бета-тестирование [Электронный ресурс]. URL: <https://ru.education-wiki.com/1061547-beta-testing> (дата обращения: 30.01.22).
8. Ввод в нотацию BPMN [Электронный ресурс]. URL: <https://www.elma-bpm.ru/journal/urok-1-vvod-v-notaciyu-bpmn/> (дата обращения: 25.03.22).
9. Веб-сервер nginx [Электронный ресурс]. URL: <https://itglobal.com/ru-ru/company/glossary/nginx/> (дата обращения: 17.02.22).
10. Диаграмма прецедентов [Электронный ресурс]. URL: [https://ru.wikipedia.org/wiki/Диаграмма\\_прецедентов](https://ru.wikipedia.org/wiki/Диаграмма_прецедентов) (дата обращения: 11.12.21).

11. Инструкции StarUML [Электронный ресурс]. URL: <https://russianblogs.com/article/76221098526/> (дата обращения: 11.12.21).
12. Классификация требований по FURPS+ [Электронный ресурс]. URL: [https://studme.org/226097/informatika/klassifikatsiya\\_trebovaniy\\_furps](https://studme.org/226097/informatika/klassifikatsiya_trebovaniy_furps) (дата обращения: 11.12.21).
13. Логическое моделирование [Электронный ресурс]. URL: <https://www.ngpedia.ru/id159031p1.html> (дата обращения: 25.03.22).
14. Объектно-ориентированные CASE-средства (Rational Rose) [Электронный ресурс]. URL: [http://citforum.ru/database/case/glava5\\_5.shtml](http://citforum.ru/database/case/glava5_5.shtml) (дата обращения: 08.03.22).
15. Онлайн-ресурс BPMN.Studio [Электронный ресурс]. URL: <https://bpmn.studio/ru> (дата обращения: 19.02.2022).
16. Подходы к управлению требованиями в IBM OpenUP и FURPS+
17. Подходы к управлению требованиями в IBM OpenUP и FURPS+ [Электронный ресурс]. URL: <https://analytics.infozone.pro/requirements-in-ibm-openup-furps/> (дата обращения: 14.12.21).
18. Учебное пособие по диаграммам последовательностей [Электронный ресурс]. URL: <https://creately.com/blog/ru/диаграмма/учебное-пособие-по-последовательной/> (дата обращения: 11.12.21).
19. Фреймворки в веб-разработке [Электронный ресурс]. URL: [https://web-creator.ru/articles/about\\_frameworks](https://web-creator.ru/articles/about_frameworks) (дата обращения: 01.02.22).
20. Этапы разработки баз данных [Электронный ресурс]. URL: <http://inftis.narod.ru/db/db-10.htm> (дата обращения: 12.02.22).
21. Язык программирования PHP [Электронный ресурс]. URL: [https://web-creator.ru/articles/subjects/programming\\_languages](https://web-creator.ru/articles/subjects/programming_languages) (дата обращения: 22.03.22).
22. Язык программирования Ruby [Электронный ресурс]. URL: <https://web-creator.ru/articles/ruby> (дата обращения: 22.03.22).
23. AOLServer [Электронный ресурс]. URL: <https://ru.wikipedia.org/wiki/AOLServer> (дата обращения: 17.02.22).



24. Apache HTTP Server [Электронный ресурс]. URL: <https://eternalhost.net/blog/hosting/web-server-apache> (дата обращения: 17.02.22).

25. Bpmn.studio [Электронный ресурс]. URL: <https://bpmn.studio/ru/about> (дата обращения: 01.01.22).

26. Cope, G. The Future of Java Applets, 2017 // *Algosome Software Design* [Электронный ресурс]: сайт о разработке программного обеспечения: <http://www.algosome.com/articles/future-of-java-applets.html>

27. Java [Электронный ресурс]. URL: <https://prog.bobrodobro.ru/107465> (дата обращения: 22.03.22).

28. Microsoft Visio [Электронный ресурс]. URL: [https://www.tadviser.ru/index.php/Продукт:Microsoft\\_Visio#](https://www.tadviser.ru/index.php/Продукт:Microsoft_Visio#) (дата обращения: 22.03.22).

29. Vaadin [Электронный ресурс]. URL: <https://ru.wikipedia.org/wiki/Vaadin> (дата обращения: 22.03.22).

30. As Is -To Be: The Essential Business Model for Process Improvement

31. Bailis P. Readings in Database Systems – 5th edition, MIT Press, 2015.

32. Fusco M. Java 8 in action – 2st edition, Pearson PLC, 2014.

33. Haridi S. Concepts, Techniques, and Models of Computer Programming – 3rd edition, MIT Press, 2014.

34. Kalloniatis C. Innovative Information Systems Modelling Techniques – InTech Press, 2016.