Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет информатики и радиоэлектроники»

Инженерно-экономический факультет

Кафедра экономической информатики

Дисциплина: Средства и технологии анализа и разработки информационных систем

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту

на тему:

WEB-ПРИЛОЖЕНИЕ ДЛЯ ТУРИСТИЧЕСКОЙ СФЕРЫ

Выполнил: гр.773601

Манкевич Данила Андреевич

Проверил: Сторожев Дмитрий Алексеевич

Минск 2020

# **ВВЕДЕНИЕ**

Современный мир трудно представить себе без использования сайтов. Практически все системы в той или иной степени связаны с функциями долговременного хранения и обработки информации. Фактически информация становится фактором, определяющим эффективность любой сферы деятельности. Современные технологии программирования постоянно развиваются именно в направлении web, давая программистам новые инструменты по разработке именно web-ориентированных приложений.

Благодаря развитию этих технологий появилась возможность автоматизировать многие процессы. Внедрение этих технологий позволит увеличить качество предоставляемых услуг, в которых нуждается заказчик, а по итогу и конечный пользователь.

Туризм представляет собой одну из самых крупных и динамичных отраслей экономики. На различные сектора экономики активно влияют следующие факторы: высокие темпы развития туризма и большие объемы валютных поступлений, которые в свою очередь способствуют формированию собственной туристической индустрии. На зону туризма приходится около 6% мирового валового национального продукта, 7% мировых инвестиций и 11% мировых потребительских расходов. Исходя из этого, можно сделать вывод, что в наши дни индустрия туризма оказывает огромное влияние на мировую экономику.

В данной курсовом проекте объектом исследования рассматривается разработка веб-приложения для туристической сферы, которое позволит эффективно и удобно подобрать тур, также представляет подсистему материальных элементов экономической деятельности и хозяйственных процессов.

Предметом исследования в данном курсовом проекте будет являться построение всех бизнес-процессов, начиная с пользовательских настроек, заканчивая конечными и мелкими настройками для запуска web-приложения.

Ожидается, что автоматизация данной функции приведет к значительному сокращению времени и трудоемкости данного учета, позволит создать единую базу клиентов, а также позволит получать и анализировать отчетность в маркетинговых целях.

При создании приложения для управления всеми процессами, нужно учесть множество факторов, для этого в данном приложении должно быть предусмотрено решение следующих задач:

* хранение информацию о различных видах туров;
* реализация функций добавления, удаления тура;
* предоставление пользователю тура в соответствии с его потребностями;
* предоставление пользователю возможности выбора тура, с последующим заказом;
* увеличение клиентской базы.

Целью курсового проекта является улучшение качества предоставления туристических услуг туроператорами и турагентствами путем создания единой рабочей среды для работы с разветвленной сетью партнеров и быстрого доступа к информации о доступности туристических услуг.

Для достижения данной цели необходимо решить следующие задачи:

* описание бизнес-процесса работы компании, предоставляющая тур-поездки (IDEF0);
* функциональное моделирование процессов предметной области;
* разработка алгоритмов работы приложения;
* создание абстрактной системы работы приложения при помощи UML-диаграмм;
* разработка клиент-серверного взаимодействия;
* разработка интерфейса программы для удобной и эффективной работы онлайн-сервиса помощи;
* тестирование программного продукта.
* создание надежной базы данных.

Информационная система позволяет печатать данные о путешествии на типографском бланке - нет необходимости заполнять бланк от руки, намного облегчается ведение и анализ клиентской базы, упрощается работа и расчеты с клиентами. Данные клиента и информация о путешествии вносятся в базу и используются для оформления всех необходимых документов: путевок, заявок, договоров, памяток и т.д. При этом значительно уменьшается вероятность ошибок при заполнении документов (например, ошибок в ФИО туристов или их паспортных данных). Информационная система позволяет легко собирать и анализировать статистические данные о количестве и характере путешествий туристов, качестве работы менеджеров агентства, популярности туристических маршрутов, отелей и т.д.

Что касается ближайшего будущего, то согласно прогнозам этой же Всемирной туристской организации, сфера туризма будет активно развиваться.

Таким образом данный сервис поможет пользователям быстро и удобно выбирать и оформлять покупку тура в зависимости от его потребности и желания.

# **1 ОПИСАНИЕ ТУРИЗМА**

Туризм — это временные выезды (путешествия) людей в другую страну или местность, отличную от места постоянного жительства на срок от 24 часов до 6 месяцев в течение одного календарного года или с совершением не менее одной ночевки в развлекательных, оздоровительных, спортивных, гостевых, познавательных, религиозных и иных целях без занятия деятельностью.

Туризм как массовое социальное явление стал складываться лишь после Второй мировой войны, хотя корни туризма уходят в глубокое прошлое. В истории развития туризма различают четыре этапа:

Первый этап – от древности до начала XIX века.  
Второй этап – от начала XIX века до начала XX века.  
Третий этап – от начала XX века до Второй мировой войны.  
Четвертый этап – после Второй мировой войны до наших дней.

Четвертый этап называется этапом массового туризма. Именно в этот период туризм приобретает массовый характер. Из предмета роскоши он становится потребностью для большинства населения индустриально развитых стран. Формируется индустрия отдыха и развлечений со своими институтами, продуктом, производственным циклом, методами организации и управления производством.

Классификация туризма — выделение внутренне однородных таксонов туристской деятельности по принятым основаниям. Наиболее распространенной классификацией туризма является его деление на типы, категории, виды и формы.

Тип туризма определяется национальной принадлежностью туристов. Всемирная туристская организация (ВТО) предложила классификацию туризма по следующим типам: внутренний, въездной (активный) и выездной (пассивный). Основным критерием классификации является пересечение государственных границ. Внутренний туризм — путешествия по стране лиц, постоянно проживающих в ней; выездной туризм — путешествия лиц, постоянно проживающих в государстве, в другую страну. Въездной туризм — путешествия по своей стране лиц, проживающих постоянно в других странах. Указанные типы туризма по-разному сочетаются между собой, образуя категории туризма.

Категории туризма включают международный и национальный туризм, а также туризм в пределах страны. С точки зрения приведенной классификации туризма по типам к национальному туризму относится внутренний и выездной туризм, к международному туризму — въездной и выездной. Туризмом в пределах страны занимаются как граждане данной страны, так и приезжающие из других стран и регионов. При этом ночующие посетители считаются туристами, а однодневные посетители — экскурсантами. В качестве экскурсантов выступают и туристы, прибывшие в страну на круизном судне, если они не используют для ночевки местные средства размещения.

Различают следующие основные сферы турбизнеса:

- производство;

- коммерция;

- финансовое предпринимательство;

- консалтинг.

Такое деление является условным, поскольку часто очень сложно разграничить производственный и непроизводственный бизнес. Если кратко определить важность каждой сферы в туризме и национальной экономике, то можно сказать следующее: в разное время каждая из этих сфер может давать различный доход, однако главной среди них остается производство. Именно в сфере производства создаются материальные блага (туристические продукты и услуги), необходимые для жизнедеятельности человека. Да, коммерция в турбизнесе существует только потому, что туроператор создал продукт, который турагент перепродает. Состояние туризма как особого межотраслевого комплекса экономики, определяется количеством высококачественных туристических продуктов и услуг, которые приходятся на одного человека. Эти же продукты и услуги в составе ВВП страны влияют на уровень развития национальной экономики. Весомой по количеству зарегистрированных туристических организаций, сейчас есть доля коммерции. Это объясняется прежде всего относительно низким стартовым капиталом и конкурентоспособной рентабельностью бизнеса, однако по уровню доходов доминирует производство.

В 2010 году Республику Беларусь с деловыми, туристическими и служебным целями посетили 4 млн. иностранных граждан, экспорт туристических услуг составил 146,7 млн. долларов США, что в 6,8 раза больше, чем за аналогичный период 2005 года. Объем платных туристических и экскурсионных услуг, оказанных населению, составил 468,7 млрд. рублей и увеличился в фактических ценах в 13 раз по сравнению с 2005 годом. В настоящее время эксперты не могут точно сказать, как будет развиваться турбизнес в Республике Беларусь в связи с короновирусной пандемией. На основании изложенного целью развития туризма в Беларуси является формирование современного, эффективного и конкурентоспособного туристского рынка, обеспечивающего широкие возможности для удовлетворения потребностей как белорусских, так и иностранных граждан в туристских услугах, повышение занятости и уровня доходов населения.

Сравнив наш сервис с booking.com и с airbnb.ru, можно заметить, что необходимый функционал нашего приложения будет меньше, однако основная функция и цель его будет такой же.

# **2 ОПИСАНИЕ ОСНОВНЫХ ПРОЦЕССОВ РАБОТЫ СЕРВИСА ПО УПРАВЛЕНИЮ БРОНИРОВАНИЕМ ТУРОВ**

Моделирование бизнес-процессов нужно проводить для выявления слабых и уязвимых мест компании, так же для повышения эффективности её работы.

Анализ начинается с построения модели как есть (AS-IS), которая отражает положение дел организации на текущий момент, позволяет систематизировать протекающие в ней процессы и информационные потоки в рамках этих процессов.

Данный метод применяется для определения процессов, которые подлежат автоматизации, хотя может применяться и только для организационных изменений.

Моделирование процессов AS-IS позволяет: определить набор действий, составляющих процесс и определить порядок выполнения действий в рамках бизнес-процесса; произвести разделение зон ответственности; определить ресурсы, потребляемые в ходе выполнения бизнес-процесса; определить степень взаимодействия между участвующими в процессе сотрудниками и подразделениями; определить информационные потоки в рамках исследуемого процесса, включая движение документов в бумажной или электронной форме; определить потенциально узкие места и возможности для улучшения процесса, которые будут использованы для его оптимизации в дальнейшем.

После исправления недостатков бизнес-модели AS-IS происходит переход к модели TO-BE (как должно быть). На основе данной модели начинается процесс проектирования модели данных, а также информационной системы.

Построим модель AS-IS на основе «очного» туристического агентства.

## **2.1 Описание модели понятным языком для заказчика**

В данной работе была разработана диаграмма IDEF0. Стандарт IDEF0 предназначен для функционального моделирования.

На рисунке 2.1 представлена Контекстная диаграмма оптимизации процесса работы оформления покупки в тур-салоне, а также определены потоки входных и выходных данных, механизмы ограничения и управления данными. Входными данными для основного процесса являются запросы данные об актуальном ассортименте и подготовленные клиентом документы. Функционирование магазина без клиентов и менеджера невозможно. Механизмом управления выбраны клиент и менеджер. Чтобы организовать все на законном основании были выбраны следующие правила: Законодательство РБ и защита прав потребителя. Результатом деятельности являются: Заключенный договор и отчет о продаже.

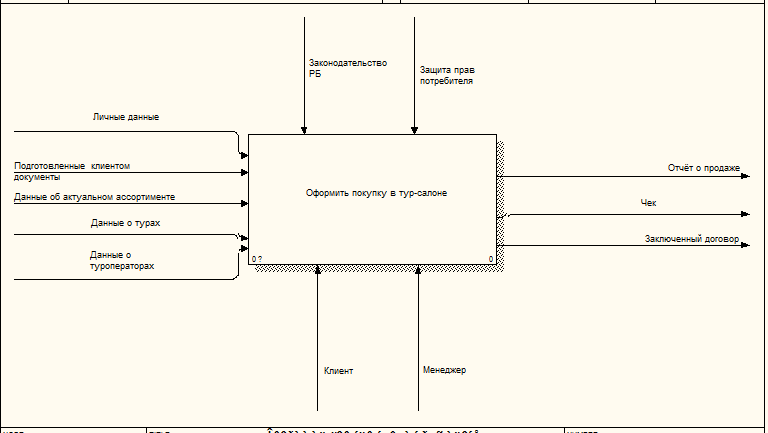


Рисунок 2.1 – Контекстная диаграмма процесса работы туристического агентства

Декомпозиция верхнего уровня контекстной диаграммы представлена на рисунке 2.2. Она разбивается на три блока: «Выбрать тур», «Оформить заказ», «Заключить договор купли-продажи».

Проанализируем каждый уровень.

Результатом декомпозиции блока «Выбрать тур» (рисунок 2.3) являются следующие уровни: «Выбрать категорию тура», «Выбрать место поездки», «Выбрать переезд», «Выбрать время поездки».

Результатом декомпозиции блока «Оформить заказ» (рисунок 2.4) являются следующие уровни: «Заказать отель и переезд», «Оформить дополнительные услуги», «Оплатить тур».

Здесь можно предоставить как будет происходить оплата тура и результатом декомпозиции блока «Оплатить тур» (рисунке 2.5), где можно увидеть следующие уровни: «Оформить рассрочку», «Оформить кредит», «Оплатить одним платежом».

Результатом декомпозиции блока «Заключить договор купли-продажи» (Рисунок 2.6) являются следующие уровни: «Заполнить личные данные», «Заполнить паспортные данные», «Проверить оплату», «Оформить гарантию», «Заключить договор».

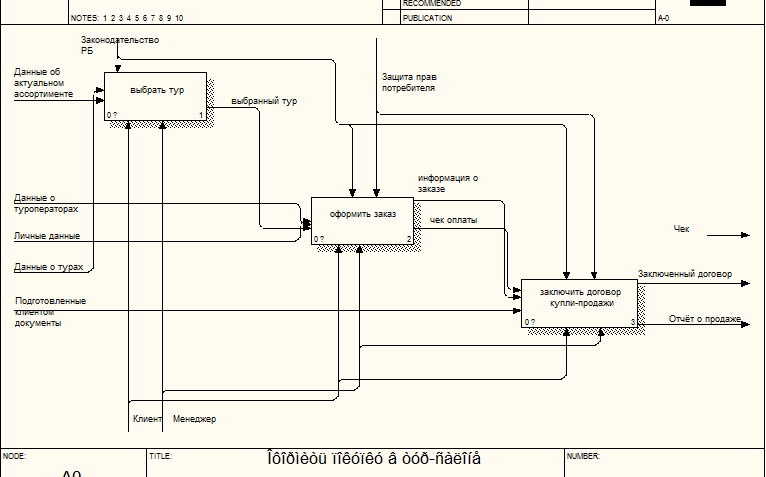


Рисунок 2.2 – Декомпозиция верхнего уровня контекстной диаграммы

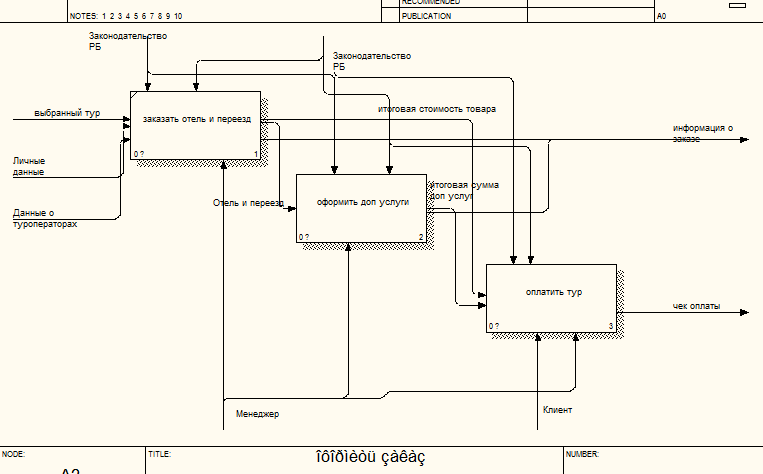


Рисунок 2.3 – Декомпозиция компонента «Выбрать тур»

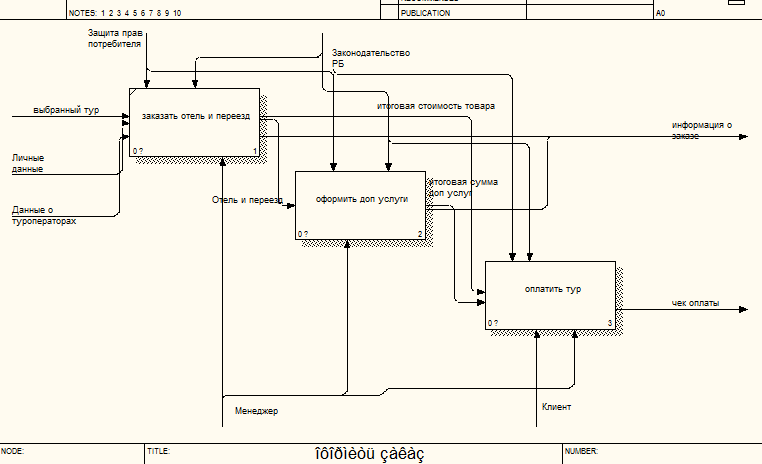


Рисунок 2.4 – Декомпозиция компонента «Оформить заказ»

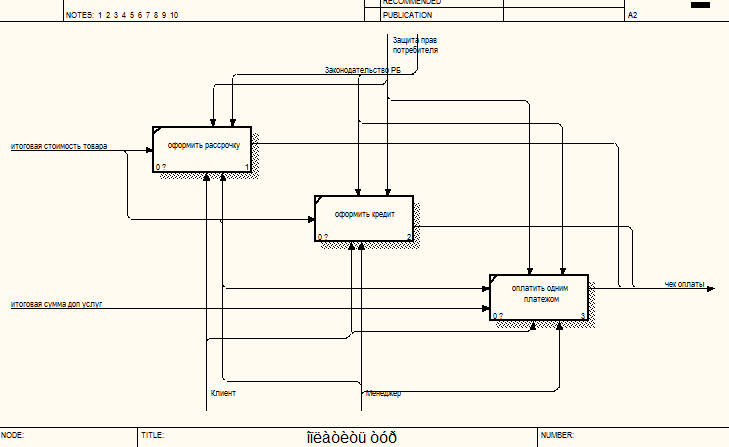


Рисунок 2.5 – Декомпозиция компонента «Оплатить тур»

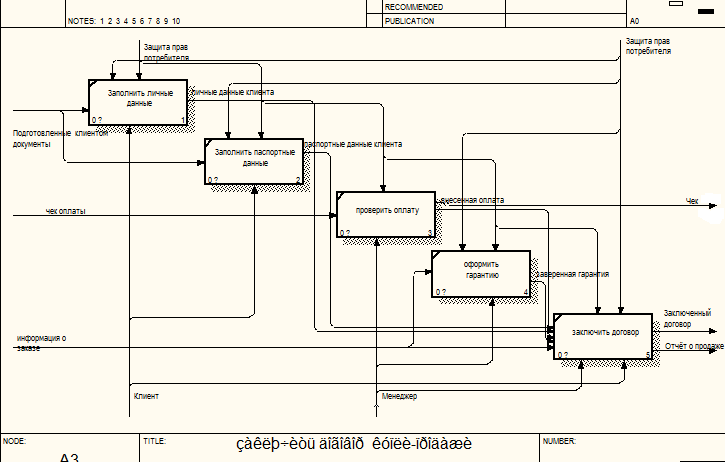


Рисунок 2.6 – Декомпозиция компонента «Заключить договор купли-продажи»

# **3 СПЕЦИФИКАЦИЯ ВАРИАНТОВ ИСПОЛЬЗОВАНИЯ СИСТЕМЫ**

Суть диаграммы вариантов использования состоит в следующем: проектируемая система представляется в виде множества сущностей или актеров, взаимодействующих с системой с помощью так называемых вариантов использования. При этом актером (actor) или действующим лицом называется любая сущность, взаимодействующая с системой извне. Это может быть человек, техническое устройство, программа или любая другая система, которая может служить источником воздействия на моделируемую систему так, как определит сам разработчик. В свою очередь, вариант использования (use case) служит для описания сервисов, которые система предоставляет актеру. Другими словами, каждый вариант использования определяет некоторый набор действий, совершаемый системой при диалоге с актером. При этом ничего не говорится о том, каким образом будет реализовано взаимодействие актеров с системой.

Для разрабатываемой системы была спроектирована диаграмма вариантов использования (рисунок 3.1). На данной диаграмме представлены действия администратора. Его основной задачей является управление пользователями. Он дает специальные права компаниям, для того, чтобы они могли размещать свои туры на портале, а так же следит за правилами поведения на сайте, и в случае их нарушения может удалить аккаунт пользователя.

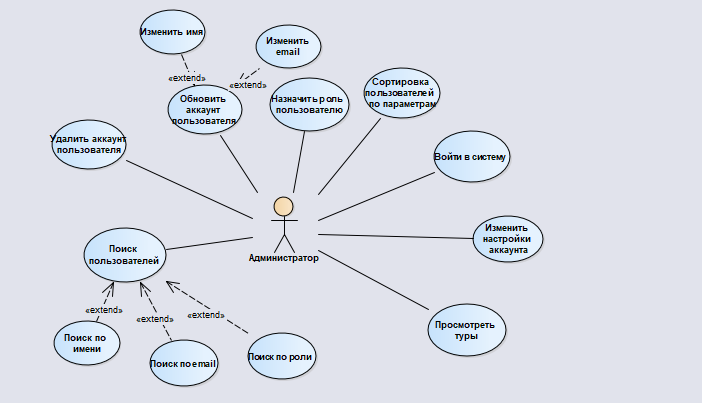


Рисунок 3.1 – Диаграмма вариантов использования

**4 ИНФОРМАЦИОННАЯ МОДЕЛЬ СИСТЕМЫ И ЕЁ ОПИСАНИЕ**

Для реализации web-приложения для туристической схемы была разработана информационная модель, на основании которой строилась база данных (Рисунок 4.1)

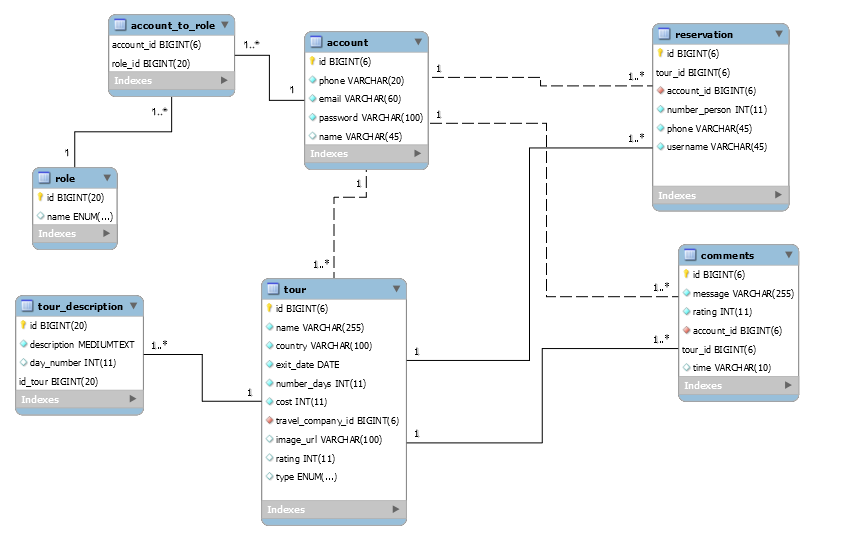


Рисунок 4.1 – Информационная модель web-приложения для туристической сферы

Для хранения данных в БД было создано 7 связанных сущностей, которые будут описаны в таблице 4.1.

Таблица 4.1 – Описание таблиц базы данных

|  |  |  |  |
| --- | --- | --- | --- |
| Название таблицы | Предназначение таблицы | Поля таблицы | Предназначение полей |
| account | Хранение данных о пользователях | id BIGINT(6),  phone VARCHAR(20),  email VARCHAR(60),  password VARCHAR(100),  name VARCHAR(45). | id – первичный ключ,  phone – телефон,  email – эл.почта,  password – пароль,  name - имя |

Продолжение таблицы 4.1 – Описание таблиц базы данных

|  |  |  |  |
| --- | --- | --- | --- |
| role | Хранение возможных ролей пользователей | id BIGINT(20),  name ENUM('ROLE\_ADMIN', 'ROLE\_USER', 'ROLE\_COMPANY') | id – первичный ключ,  name – название ролей |
| account\_to\_role | Используется для связи многие-ко-многим между таблицами account и role | account\_id BIGINT(6),  role\_id BIGINT(6) | account\_id – внешний ключ с таблицей account,  role\_id – внешний ключ с таблицей role |
| tour | Хранение данных о турах | id BIGINT(6),  name VARCHAR(225), country VARCHAR(100), exit\_date DATE, number\_days INT(11),  cost INT(11),  compony\_id BIGINT(6), image\_url VARCHAR(100), rating INT(11),  type ENUM('BUS\_TOUR', 'AIR\_TOUR', 'SEA\_TOUR') | id – первичный ключ,  name – название тура,  country – страна отправления,  exit\_date – дата отправления,  number\_days – продолжительность тура,  company\_id – внешний ключ с таблицей account,  image\_url – путь хранение изображений для тура,  type – тип тура |
| tour\_description | Используется для хранения описаний программы тура | id BIGINT(20),  description MEDIUMTEXT, day\_number(11),  id\_tour BIGINT(20) | id – первичный ключ,  description – описание тура,  day\_number – день тура,  id\_tour – внешний ключ с таблицей tour |

Продолжение таблицы 4.1 – Описание таблиц базы данных

|  |  |  |  |
| --- | --- | --- | --- |
| reservation | Хранит данные о бронях пользователей | id BIGINT(6),  tour\_id BIGINT(6),  number\_person INT,  phone VARCHAR(45),  username VARCHAR(45), status ENUM('IN\_PROCESS', 'DENIED', 'APPROVED') | id – первичный ключ,  tour\_id – внешний ключ с таблицей tour,  account\_id – внешний ключ с таблицей account,  username – имя,  phone – телефон,  status – статус брони |
| comments | Хранит комментарии пользователей | id BIGINT(11),  message VARCHR(255), rating INT,  account\_id BIGINT,  tour\_id BIGINT,  time VARCHAR(10) | id – первичный ключ,  message – текст комментария,  rating – рейтинг,  account\_id – внешний ключ с таблицей account,  tour\_id – внешний ключ с таблицей tour,  time – дата комментария |

# **5 ОБОСНОВНИЕ ВЫБОРА КОМПОНЕНТ И ТЕХНОЛОГИЙ ДЛЯ РЕАЛИЗАЦИИ КУРСОВОГО ПРОЕКТА**

## **5.1 Описание выбранных компонентов технологий и обоснование их выбора**

При написании web-приложения использовались следующие компоненты и технологии: язык Java, Apache Maven, Tomcat, Spring Core, Spring MVC, Spring Security, Hibernate, HTML, CSS, Bootstrap, jQuery, ajax.

Для написания веб-приложения был выбран язык Java. Причиной этому являются некоторое особенности языка. Во-первых, в отличие от многих других языков, включая C и C++, Java, когда был создан, он не компилировался в платформе конкретной машины, а в независимом от платформы байт-коде. Этот байт код распространяется через интернет и интерпретируется в Java Virtual Machine (JVM), на которой он в настоящее время работает. Во-вторых, Java прилагает усилия, чтобы устранить ошибки в различных ситуациях, делая упор в основном на время компиляции, проверку ошибок и проверку во время выполнения. В-третьих, в Java встроено огромное количество мощных стандартных библиотек. В-четвертых, у Java есть встроенная поддержка работы в сетях (как локальных, так и Internet). В-пятых, у Java есть функции многопоточности, можно писать программы, которые могут выполнять множество задач одновременно. Введение в язык Java этой конструктивной особенности позволяет разработчикам создавать отлаженные интерактивные приложения. В-шестых, введение Just-In-Time компилятора позволило получить высокую производительность.

Для автоматизации сборки проектов использовался фреймворк Apache Maven. При этом сборка проектов осуществляется на основе описания структуры и зависимостей в файле pom.xml.

Tomcat позволяет запускать [веб-приложения](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5) и содержит ряд программ для самоконфигурирования, используется в качестве самостоятельного [веб-сервера](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D1%80), в качестве сервера контента.

Для разработки web-приложения использовался фреймворк Spring MVC. Фреймворк Spring MVC обеспечивает архитектуру паттерна Model-View-Controller 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. В приложении это реализовано тем, что его работа разделена на три составляющие: пакет controller, пакет service и пакет views.

Http-запрос, который приходит со страницы, обрабатывается классами-контроллерами, которые в свою очередь передают управление дальше сервисами, который будут обращаться через репозитории к базе данных. Ответом от контроллера является либо jsp-страница, либо ResponseEntity, которая содержит какой-либо объект, а также http-статус, который является ответом контроллера и результатом обработки запроса.

Spring Security это Java/JavaEE framework, предоставляющий механизмы построения систем аутентификации и авторизации, а также другие возможности обеспечения безопасности для корпоративных приложений, созданных с помощью Spring Framework.

В разрабатываемом приложении с помощью Spring Security реализована авторизация пользователей, а также контроль доступа к ресурсам, с помощью класса конфигураций.

Hibernate – самая популярная реализация спецификации JPA, предназначенная для решения задач объектно-реляционного отображения (ORM). Целью Hibernate является освобождение разработчика от значительного объёма сравнительно низкоуровневого программирования при работе в объектно-ориентированных средствах в реляционной базе данных. Исходя из этого, способом работы с базой данных был выбран Hibernate.

Для создания веб-страниц использовались такие языки, как HTML, CSS, Bootstrap, JS и его фреймворки.

HTML – это гипертекстовый язык разметки, который используется для создания документов в Интернет(веб-страниц). При помощи HTML создавалась веб-страница, содержащая все нужные элементы.

CSS (каскадные таблицы стилей) — это язык, который отвечает за визуальное представление документов пользователю. Стили для элементов веб-страницы можно прописывать как в самом html-документе, так и в отдельном файле. В проекте используется внешний файл, который при помощи специального тега подключается к html-документу. Такие файлы имеют расширение \*.css.

Для упрощения создания разметки страницы использовались готовые шаблоны Bootstrap.

Bootstrap — свободный набор инструментов для создания [сайтов](https://ru.wikipedia.org/wiki/%D0%A1%D0%B0%D0%B9%D1%82) и [веб-приложений](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B5). Включает в себя [HTML](https://ru.wikipedia.org/wiki/HTML)- и [CSS](https://ru.wikipedia.org/wiki/CSS)-шаблоны оформления для [типографики](https://ru.wikipedia.org/wiki/%D0%A2%D0%B8%D0%BF%D0%BE%D0%B3%D1%80%D0%B0%D1%84%D0%B8%D0%BA%D0%B0), веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая [JavaScript](https://ru.wikipedia.org/wiki/JavaScript" \o "JavaScript)-расширения.

jQuery — [набор функций JavaScript](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D0%B1%D0%BB%D0%B8%D0%BE%D1%82%D0%B5%D0%BA%D0%B0_JavaScript), фокусирующийся на взаимодействии [JavaScript](https://ru.wikipedia.org/wiki/JavaScript" \o "JavaScript) и [HTML](https://ru.wikipedia.org/wiki/HTML). Библиотека jQuery помогает легко получать доступ к любому элементу [DOM](https://ru.wikipedia.org/wiki/Document_Object_Model), обращаться к атрибутам и содержимому элементов [DOM](https://ru.wikipedia.org/wiki/Document_Object_Model), манипулировать ими. Также библиотека jQuery предоставляет удобный [API](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F_%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D0%B9) для работы с [AJAX](https://ru.wikipedia.org/wiki/AJAX).

AJAX (аббревиатура от «Asynchronous Javascript And Xml») – технология обращения к серверу без перезагрузки страницы. За счёт этого уменьшается время отклика и веб-приложение по интерактивности больше напоминает десктоп.

JQuery и AJAX в проекте используются для взаимодействия веб-страниц с контроллерами. С их помощью формируются запросы на страницах.

## **5.2 Диаграмма последовательности для функции «Бронирование»**

Самой богатой и гибкой формой диаграмм взаимодействий является диаграмма последовательностей. Она представляет взаимодействие между линиями жизни как упорядоченную последовательность событий.

Рассмотрим диаграмму последовательностей, представленную на рисунке 5.1.

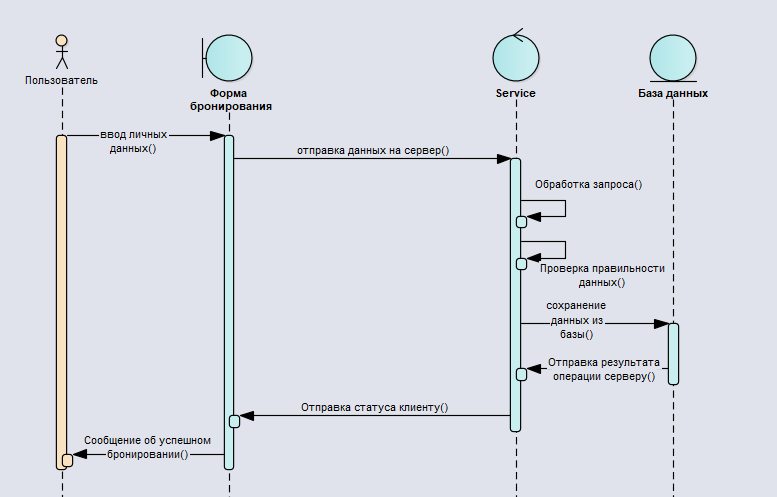


Рисунок 5.1 – Диаграмма последовательности для процесса бронирования тура

Диаграмма последовательностей отображает временные особенности передачи и приема сообщений между объектами. Время на диаграмме развивается сверху вниз, а жизненный цикл горизонтально. Фокус управления первоначально находится на актере «Пользователь». На странице выбранного туры он заполняет специальную форму, где вводит свое имя, номер телефона для связи, а также количество интересующих его мест в туре. После этого данные отправляются на сервер и, если все поля заполнены верно, то запись о брони сохраняется в базу, после чего ее должен будет рассмотреть туроператор. Если сохранение в базу произошло успешно, то клиенту отправляется сообщение об успешном бронировании.

## **5.3 Диаграмма компонентов**

Диаграмма компонентов (component diagram) ‒ показывает взаимосвязи между модулями (логическими, физическими), из которых состоит моделируемая система. Основной тип сущностей на диаграмме компонентов – компоненты и интерфейсы, посредством которых указывается взаимосвязь между компонентами. На диаграмме компонентов применяются следующие отношения:

1. реализации между компонентами и интерфейсами (компонент реализует интерфейс);
2. зависимости между компонентами и интерфейсами (компонент использует интерфейс).На рисунке А.1 представлена диаграмма компонентов, которая обуславливает архитектуру разрабатываемой системы.

## **5.4 Диаграмма развертывания**

Диаграмма размещения (deployment diagram) наряду с отображением состава и связей элементов системы показывает, как они физически размещены на вычислительных ресурсах во время выполнения. Таким образом, на диаграмме размещения, по сравнению с диаграммой компонентов, добавляется два типа сущностей: артефакт, который является реализацией компонента и узел (может быть как классификатор описывающий тип узла, так и конкретный экземпляр), а также отношение ассоциации между узлами, показывающее, что узлы физически связаны во время выполнения.

Процесс создания диаграммы состоит из двух этапов. Первый, основное внимание уделяется на узлы и соединения. Второй, распределение экземпляров артефактов по экземплярам узлов или же артефактов по узлам. [4].

Рассмотрим созданную диаграмму развертывания системы (Рисунок А.2).

**6 МОДЕЛИ ПРЕДСТАВЛЕНИЯ СИСТЕМЫ И ИХ ОПИСАНИЕ**

Для визуального моделирования систем используется универсальный язык моделирования UML. Основной идеей данного языка является возможность моделирования ПО и других систем как наборы взаимодействующих объектов. В UML существует три строительных блока, такие как сущности, отношения, диаграммы. Сущности представляют элементы модели. Отношения определяют связь между сущностями. Диаграммы показывают наборы сущностей, которые являются способом представления того, что будет делать система или как. [4].

**6.1 Диаграмма состояния для варианта использования «Бронирование тура»**

Состояние определяется как ситуация в жизни объекта, при которой он удовлетворяет некоторому условию. Состояние объекта со временем меняется. События, которые могут привести к изменению состояния получили название сообщения. Чтобы успешно смоделировать состояния нужно выяснить наиболее значимые из них.

На рисунке А.3 представлен конечный автомат моделирования процесса бронирования тура. Процесс начинается с желания пользователя поехать в поездку и выбрать тур. После этого он начинает ознакомление с предлагаемыми турами и выбирает подходящий для него. Если тур выбран, то клиент может уточнить данные у туроператора. После этого, для бронирования тура, пользователь должен авторизоваться на сайте, после чего он заполняет специальную форму и бронирование проходит успешно.

**6.2 Алгоритм регистрации пользователя**

# **8 РУКОВОДСТВО ПО РАЗВЕРТЫВНИЮ СИСТЕМЫ**

Для запуска приложения необходимо установить MySQL, JDK версии 1.8 и выше, а также JBoss EAP.

**ПРИЛОЖЕНИЕ А**

**(обязательное)**

**Модели представления системы**

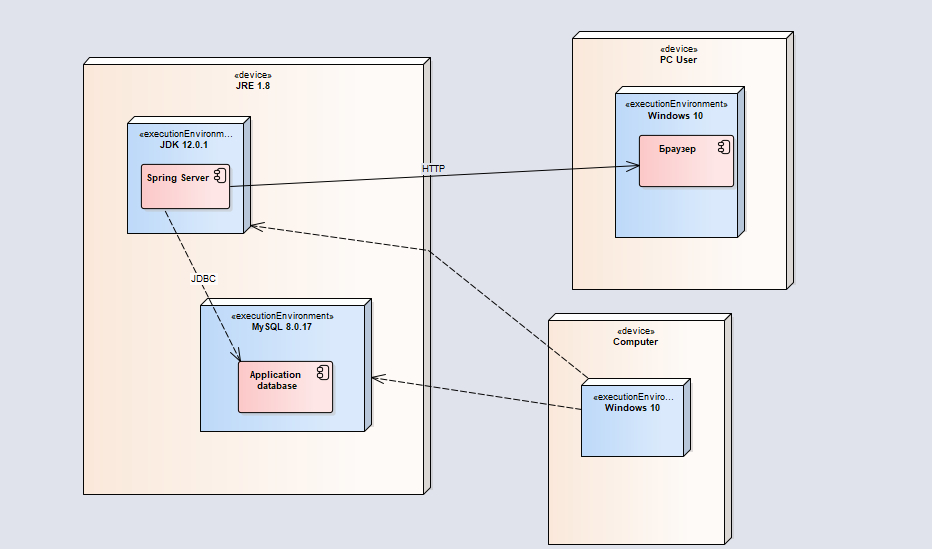


Рисунок А.1 – Диаграмма компонентов

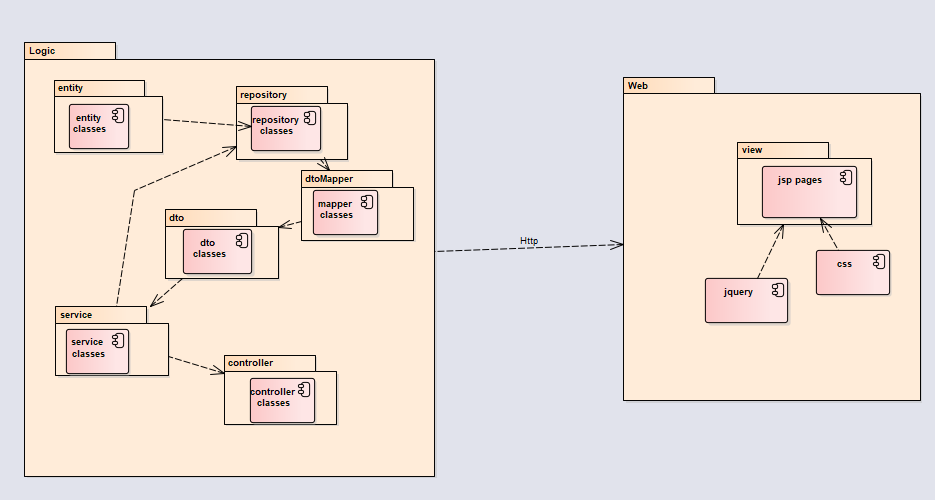


Рисунок А.2 – Диаграмма развертывания

Продолжение приложения А

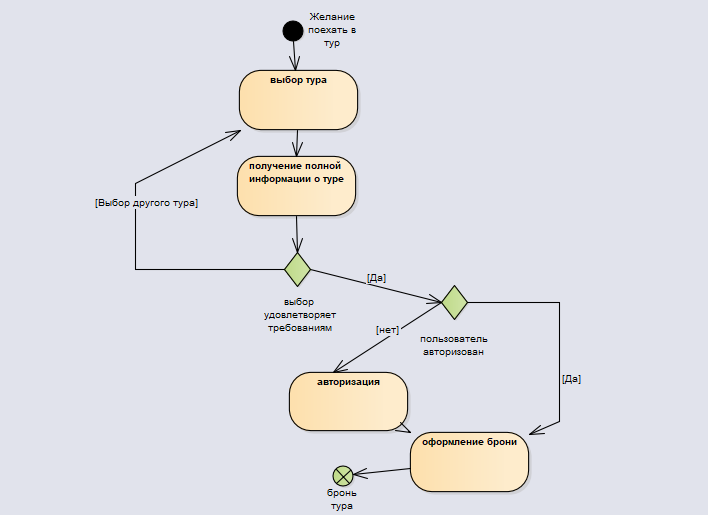


Рисунок А.3 – Диаграмма состояний для процесса оформления заказа