

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

Институт компьютерных и инженерных наук
Кафедра информационных и управляющих систем
Направление подготовки 09.04.04 – Программная инженерия
Направленность (профиль) образовательной программы Управление разработкой программного обеспечения

ДОПУСТИТЬ К ЗАЩИТЕ
Зав. кафедрой
_____ А.В. Бушманов
«___» _____ 2025 г.

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

на тему: Проектирование и реализация информационной системы онлайн-бронирования отелей

Исполнитель
студент группы 3105-ом2

Яо Жуй

(подпись, дата)

Руководитель
доцент, канд. техн. наук

О.В. Жилиндина

(подпись, дата)

Руководитель научного
содержания программы
магистратуры
профессор, доктор техн. наук

И.Е. Ерёмин

(подпись, дата)

Нормоконтроль
инженер кафедры

В.Н. Адаменко

(подпись, дата)

Рецензент
доцент, канд. техн. наук

Л.В. Никифорова

(подпись, дата)

Благовещенск, 2025

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

Институт компьютерных и инженерных наук
Кафедра информационных и управляющих систем

УТВЕРЖДАЮ

Зав. кафедрой

_____ А.В. Бушманов

« ____ » _____ 2025 г.

З А Д А Н И Е

К магистерской диссертации студента группы 3105-ом2 Яо Жуй

1. Тема магистерской диссертации: Проектирование и реализация информаци-
онной системы онлайн-бронирования отелей

(Утверждено приказом от 06.03.2025 № 609-уч)

2. Срок сдачи студентом законченной работы (проекта): 10.06.2025

3. Исходные данные к магистерской диссертации: научная литература, доку-
ментация разработчиков, интернет ресурсы

4. Содержание магистерской диссертации (перечень подлежащих разработке
вопросов): описание текущего состояния системы управления онлайн-брони-
рованием отелей, описание алгоритма работы веб-сайта системы и каждого
модуля системы, программная реализация системы и тестовые случаи

5. Перечень материалов приложения (наличие чертежей, таблиц, графиков,
схем, программных продуктов, иллюстративного материала и т.п.): _____

6. Рецензент магистерской диссертации: Л.В. Никифорова, доцент, канд. техн.
наук

7. Дата выдачи задания 14.01.2025

Руководитель выпускной квалификационной работы: _____

О.В. Жилиндина, доцент, канд. техн. наук

(фамилия, имя, отчество, должность, уч.степень, уч.звание)

Заявление принял к исполнению _____

РЕФЕРАТ

Магистерская диссертация содержит 70 страницы, 29 рисунков, 28 источников, таблиц 11

УПРАВЛЕНИЕ ИНФОРМАЦИЕЙ ОБ ОНЛАЙН-БРОНИРОВАНИИ ОТЕЛЕЙ, БРОНИРОВАНИЕ НОМЕРОВ, БРОНИРОВАНИЕ ПИТАНИЯ, ASP.NET, MYSQL

Архитектура B/S объединяет ASP.NET и MySQL для обеспечения возможностей кроссплатформенного доступа; многомодульное управление разрешениями различает администраторов, сотрудников и пользователей для обеспечения безопасности данных; интегрированная система, объединяющая онлайн-бронирование, сообщения, отзывы, питание и управление номерами; использует технологию шифрования MD5 для обеспечения безопасности пароля пользователя; поддерживает много-терминальный доступ для улучшения пользовательского опыта; при проектировании системы основное внимание уделяется простоте использования и обслуживания для упрощения рабочих процедур.

СОДЕРЖАНИЕ

Введение	5
1 Общие характеристики системы управления информацией онлайн-бронирования отелей	7
1.1 Анализ процессов по бронированию отелей	7
1.2 Обзор существующих методов решения аналогичных задач	17
2 Алгоритмическое и программное обеспечение решения задачи	22
2.1 Общий алгоритм управления онлайн-бронирования	22
2.2 Функциональные модули и методы их управления	24
2.3 Обзор возможностей профильного программного обеспечения	34
3 Программная реализация предлагаемого алгоритма решения задачи	43
3.1 Основные этапы практической разработки программного продукта	43
3.2 Примеры фактического тестирования программного продукта	61
Заключение	68
Библиографические ссылки	70

ВВЕДЕНИЕ

Систему управления информацией онлайн-бронирования отеля можно разделить на три основных модуля: модуль администратора, модуль сотрудника и модуль пользователя. Функции модуля администратора включают управление пользователями, управление сообщениями, управление новостными данными, управление регистрациями и выездами, управление товарами, управление размещением, управление гостевыми комнатами, а функции модуля управления рецептами включают регистрацию; управление входом и выездом, управление номерами для гостей, управление размещением, управление рецептами, управление клиентами.

Функции пользовательского модуля включают бронирование номеров, изменение данных, онлайн-сообщения, онлайн-обзоры, бронирование столиков и просмотр регистрации; во-первых, эта система знакомит с целью и значением исследования, текущим статусом исследований в стране и за рубежом, технологиями разработки и инженерными разработками. Затем проводится технико-экономическое обоснование и анализ потребностей системы, а также проектирование функционального модуля, проектирование базы данных, а также детальное проектирование и реализация интерфейсных и серверных модулей. Наконец, описаны методы тестирования, используемые в этой системе, и перечисленные тестовые примеры. С точки зрения проектирования системы, эта система основана на структуре B/S, использует базу данных MySQL и применяет технологию ASP.NET и язык C#.

Основные цели онлайн-системы управления информацией отеля включают в себя следующие моменты:

- повышение эффективности обслуживания: благодаря онлайн - управлению можно быстро реагировать на бронирования, запросы и отзывы клиентов, сокращая время ручных операций и частоту ошибок;
- повышение удовлетворенности клиентов: система может предостав-

лять более персонализированные и удобные услуги, такие как онлайн-бронирование, самостоятельная регистрация заезда/отъезда, обслуживание клиентов в режиме реального времени и т. д., чтобы улучшить качество обслуживания клиентов;

- оптимизация управления операциями: благодаря централизованному управлению номерами, питанием, клиентами и другой информацией, это помогает отелям лучше планировать ресурсы и повышать операционную эффективность;

- снизить эксплуатационные расходы: автоматизированное и информатизированное управление снижает потребление рабочей силы и материальных ресурсов, тем самым снижая эксплуатационные расходы;

- повышение конкурентоспособности рынка: можно использовать систему для анализа рыночных тенденций и разработки маркетинговых стратегий, которые помогут отелям более точно позиционировать рынок и привлекать целевые группы клиентов;

- содействие принятию научных решений: онлайн-система управления информацией в отеле предоставляет руководству отеля подробные инструменты отчетности и анализа данных для поддержки принятия решений на основе данных. Функция анализа данных системы в режиме реального времени позволяет отелям своевременно корректировать бизнес-стратегии, чтобы реагировать на изменения рынка;

- содействие инновациям и развитию: онлайн-система управления информацией в отеле поддерживает быстрый запуск новых услуг и улучшение старых услуг, способствуя постоянным инновациям в гостиничном бизнесе. Гибкость и масштабируемость системы дают возможность отелям внедрять новые технологии и модели обслуживания.

1 ОБЩИЕ ХАРАКТЕРИСТИКИ СИСТЕМЫ УПРАВЛЕНИЯ ИНФОРМАЦИЕЙ ОНЛАЙН-БРОНИРОВАНИЯ ОТЕЛЕЙ

1.1 Анализ процессов по бронированию отелей

Развитие гостиничного бизнеса включает в себя множество факторов, и его собственный потенциал развития огромен. Однако в то же время общее развитие гостиничного бизнеса также сталкивается с большой конкуренцией. В реальной работе гостиничного бизнеса уровень управления и обслуживания будет иметь большое влияние на качество всей гостиничной индустрии. Сегодня, с быстрым развитием модернизации, каждый менеджер отеля решает, как использовать информационные технологии и другие инструменты. Необходимо решить Важнейший вопрос – повышение качества обслуживания и мощности гостиницы.

Если отель хочет продолжать развиваться в будущем, он должен улучшать качество обслуживания, что требует от отеля инвестиций в высокие технологии и эффективной эксплуатации отеля. Применение информационных технологий в управлении гостиницами может помочь гостиницам сократить эксплуатационные расходы. Для развития отеля требуется единая система управления онлайн-бронированием. Функции этой системы не могут ограничиваться определенной функцией. Она должна иметь полный интерфейс данных и быть управляемой и контролируемой для удовлетворения потребностей управления отелем. Используя информационную систему управления отелем, можно хранить большой объем данных, улучшать качество обслуживания отеля и улучшать внутреннее управление отелем. Содействовать развитию отрасли на новом уровне и эффективно повышать экономическую выгоду отелей.

Отель – это услуга, и в этой отрасли гости – боги. Менеджеры отелей должны серьезно задуматься о том, сможем ли мы обслуживать гостей лучше. Помимо предоставления качественных услуг, качественная гостиница может также иметь относительно полное оснащение (оборудование) и гуманизиро-

ванные услуги, тем самым обеспечивая частоту повторения клиентов и удовлетворенность клиентов.

В то же время измерение рабочего состояния отеля в основном зависит от того, может ли он удовлетворить требования различных пользователей. Система управления отелем основана на Интернете. На основе этого она обеспечивает онлайн-функционирование гостиничного бизнеса и может не только обеспечить канал управления для гостиничных операторов, но и предоставить клиентам удобные и качественные услуги, связанные с отелем. Таким образом, значение исследования системы управления онлайн-бронированием отелей заключается в следующем: во-первых, оно может предоставить стандартизированные спецификации управления для традиционных отелей, тем самым улучшая стандартизированную работу отеля и тем самым улучшая качество и уровень обслуживания. Во-вторых, она обеспечивает безбумажный способ для стойки отеля добавлять, удалять, изменять и запрашивать информацию о клиентах, что делает систему удобной и быстрой с меньшим количеством ошибок.

Система управления информацией об онлайн-бронировании отелей – это автоматизированный инструмент, используемый для обработки и управления бронированием отелей. Такая система может помочь менеджерам отелей эффективно управлять бронированием номеров, информацией о клиентах, ценами и наличием мест и т. д. Кроме того, он может предоставлять различные отчеты и анализ, которые помогут менеджерам отелей принимать более обоснованные решения.

Описание системы может включать следующее:

- название компании или предприятия: например, «Система онлайн - бронирования отелей XYZ»;

- основная деятельность: Основной деятельностью этой системы является обработка и управление бронированием отелей. Это включает в себя получение запросов на бронирование, подтверждение бронирования, обработку отмены и изменений, а также управление наличием номеров и ценами;

– дополнительные действия. В дополнение к основным действиям система может также предоставлять другие функции, такие как создание отчетов и анализ, а также управление информацией о клиентах;

– юридический статус: это может быть система, разработанная и поддерживаемая внутри компании, или это может быть услуга, предоставляемая сторонним поставщиком;

– позиционирование на рынке. Система может позиционироваться как эффективное и простое в использовании решение для бронирования отелей, обеспечивающее комплексные функции управления;

– границы исследования. Система может сосредоточиться на обработке и управлении бронированием отелей, не принимая во внимание другие сопутствующие виды деятельности, такие как общественное питание или управление объектами;

– задачи, требующие автоматизации. Сюда могут входить получение и подтверждение бронирования, обработка отмены и изменения, управление наличием номеров и ценами, создание отчетов и аналитики, управление информацией о клиентах и интеграция с другими системами.

Организационная структура может включать в себя следующие компоненты:

– менеджмент: отвечает за общую стратегию и принятие решений;

– технический отдел: отвечает за разработку и обслуживание системы;

– отдел обслуживания клиентов: отвечает за обработку вопросов и запросов клиентов;

– отдел продаж и маркетинга: отвечает за продвижение системы и привлечение новых клиентов;

– финансовый отдел: отвечает за управление финансовым состоянием компании. Экономическая целесообразность. С точки зрения экономической выгоды, поскольку эта система предоставляет услуги онлайн-бронирования отелей, она имеет информационный процесс и платформу приложений, поддерживающую эту систему. Таким образом, нет необходимости тратить

больше денег на аренду другого оборудования. Просто можно добавить компьютер, сервер и обслуживающий персонал к исходной базе. Следовательно, с экономической точки зрения возможно внедрение системы управления онлайн-заказами отелей.

Социальная целесообразность. Социальная целесообразность системы анализируется с точки зрения пользователя. В традиционных операциях управление онлайн-бронированием отелей представляет собой очень сложную задачу. Как только возникает проблема с определенным фрагментом статистики, его необходимо реорганизовать, что повлияет на нормальную работу. Эта система позволяет персоналу быстро и точно получать данные.

Эксплуатационная осуществимость. Поскольку система разработана в режиме В/С, пока у пользователя есть возможность искать и просматривать информацию с помощью браузера, соответствующие операции в системе могут быть легко выполнены, и пользователям не нужно проходить соответствующее обучение перед использованием. Дизайн системы управления онлайн-бронированием отелей аналогичен дизайну популярных веб-сайтов, поэтому пользователи могут ознакомиться с системой за очень короткое время. Это оперативно осуществимо.

Эта система имеет функцию входа и регистрации. Когда новый пользователь впервые заходит на веб-сайт, ему необходимо завершить регистрацию, чтобы забронировать номер, столик в ресторане, онлайн - сообщения и комментарии. Администраторы могут управлять учетными записями пользователей. Для получения информации о номерах вы можете добавить фотографии номеров, цены, типы и удобства.

А посредством исследований и анализа разработчиков мы можем точно проанализировать потребности пользователей с точки зрения пользователя, понять производительность и характеристики пользователей и проектов, а также обратить внимание на то, могут ли разработчики также использовать потенциальные и глубокие потребности пользователей. После полного понимания схемы работы исходной системы мы также можем уточнить различные

потребности пользователей и определить функции новой системы. При этом, анализ требований может повысить эффективность системы. Цель состоит в том, чтобы сделать систему более плавной и предоставить пользователям лучший опыт, тем самым увеличивая привязку пользователей.

- требования к целостности системы. Эта система имеет полные функции, такие как вход в систему и регистрация, управление пользовательской информацией, управление сообщениями и комментариями, управление гостевыми комнатами, управление новостями и управление питанием;

- требования к безопасности системы. Эта система использует базу данных MySQL для хранения данных и подключается к базе данных через имена пользователей и пароли для обеспечения безопасности данных. Наконец, эта система будет развернута на VS, что сможет обеспечить безопасную работу проекта;

- требования к работоспособности системы. Эта система предназначена в основном для бронирования номеров в отелях и развертывается на VS для работы. Администратору необходимо управлять только справочной информацией. Пользователям нужно только зарегистрироваться и войти в систему, и они смогут комментировать комнаты и просматривать ответы.

Регистрационный модуль:

Пользователи могут зарегистрировать свои учетные записи на стойке регистрации системы. Информация, необходимая для регистрации учетной записи, включает в себя: имя пользователя, пароль, пароль подтверждения, адрес электронной почты, QQ и т. д. После отправки информации для входа система определит пользователя на основе JS. кодировки. Проверяется соответствие введенной информации спецификации. Далее добавляется запись о пользователе в базу данных системы. В целях безопасности используется алгоритм шифрования MD5 для шифрования пароля и последующего сохранения его в базе данных.

Модуль входа:

В процессе входа пользователя в систему, включая имя пользователя,

пароль и код подтверждения, система должна сначала определить, является ли код подтверждения правильным или неправильным. Если пользователь входит в систему с использованием кода подтверждения, то он должен не только ввести имя пользователя и пароль при входе в систему, а также код подтверждения. После определения правильности проверочного кода определяются имя пользователя и пароль. Поскольку пароль пользователя зашифрован с помощью MD5, при определении пароля пользователя пароль должен быть преобразован в MD5, прежде чем можно будет определить пароль пользователя; если имя пользователя или пароль неверны, на странице входа появится сообщение об ошибке.

Модуль управления сообщениями: модуль управления сообщениями включает в себя подмодули, такие как сообщение, просмотр сообщения и ответ на сообщение. Эта функция может использоваться пользователями и менеджерами в системе. Пользователи могут оставлять сообщения и просматривать ответы менеджеров, а менеджеры могут просматривать сообщения отзывов пользователей. Оставить сообщение и ответ.

Модуль управления комментариями: модуль управления комментариями включает в себя подмодули, такие как онлайн-комментарии, просмотр комментариев и управление комментариями. Эту функцию могут использовать пользователи и администраторы системы. Пользователи могут комментировать и просматривать комментарии других пользователей, а администраторы могут находить все записи комментариев, удалять или изменять их.

Модуль управления питанием: модуль управления питанием включает в себя подмодули записи питания и просмотра питания. Функциональный модуль могут использовать как пользователи, так и менеджеры. Только менеджеры могут добавлять, изменять или удалять информацию о питании, а другие пользователи могут только просматривать информацию о питании.

Модуль управления помещениями: модуль управления гостевой комнатой включает в себя подмодули входа в гостевую комнату и просмотра госте-

вой комнаты. Этот функциональный модуль могут использовать как пользователи, так и менеджеры. Только менеджеры могут добавлять, изменять или удалять информацию о комнате, а другие пользователи могут только просматривать информацию о комнате.

Модуль управления регистрацией: модуль управления регистрациями включает в себя подмодули записи регистрации и просмотра регистрации. Этот функциональный модуль могут использовать как пользователи, так и менеджеры. Только менеджеры могут добавлять, изменять или удалять информацию о регистрации, а другие пользователи могут только просматривать информацию о регистрации.

Модуль управления оформлением: модуль управления оформлением включает в себя подмодули записи оформления заказа и просмотра оформления заказа. Этот функциональный модуль могут использовать как пользователи, так и менеджеры. Только менеджеры могут добавлять, изменять или удалять информацию о оформлении заказа, а другие пользователи могут только просматривать данные о оформлении заказа.

Модуль управления сотрудниками: он включает в себя подмодули ввода данных о сотрудниках и просмотра информации о сотрудниках. Этот функциональный модуль могут использовать как пользователи, так и менеджеры. Только менеджеры могут добавлять, изменять или удалять информацию о сотрудниках, а другие пользователи могут только просматривать информацию о сотрудниках.

Проблемы, существующие в системах управления информацией онлайн-бронирования отелей, включают низкую эффективность управления информацией, отсутствие стратегии устойчивого развития и отсутствие талантов в области управления информацией. Общие методы решения этих проблем включают инновационные бизнес-модели, уделение внимания системам управления информацией и создание превосходной команды специалистов по управлению информацией.

Краткое описание онлайн-информационной системы отеля.

Проблемы:

- низкая эффективность управления информацией. хотя многие гостиницы внедрили системы управления информацией, они не могут эффективно использовать систему управления информацией в реальной работе, в результате чего многие функции не применяются, что влияет на эффективность управления;

- отсутствие стратегии устойчивого развития. некоторые гостиничные компании не включили управление информацией в число направлений развития и имеют недостаточные капитальные вложения, что повлияло на общий уровень управления информацией в гостиницах;

- отсутствие талантов в области управления информацией: в отеле не хватает профессиональных управленческих талантов в работе по управлению информацией, что приводит к неспособности предоставлять профессиональные решения при возникновении проблем;

- несовершенная информационная сетевая система. информационная сетевая система является низкоуровневой и несовершенной, что влияет на единое управление отраслью и стабильность информационной сети.

За последние десятилетия отели в Китае быстро развивались. Частично это связано с экономическим развитием, а частично с тем, что люди становятся все более непредубежденными, их материальная жизнь становится богаче, а индустрия туризма также становится все более открытой. все более и более изощренной. По сравнению с гостиницами других стран уровень информатизации отелей Китая значительно ниже. Система управления отелями в моей стране только началась с компьютерных технологий. В 21 веке, с широким распространением крупномасштабных рынков электронной коммерции, все больше отелей высокого класса начали использовать системы управления отелями для управления своим основным бизнесом. С ростом популярности компьютерного программного обеспечения и снижением его стоимости многие отели начали использовать компьютеры для автоматизированного управления,

что открывает широкие возможности для разработки программного обеспечения для управления гостиницами. Судя по данным отечественной гостиничной индустрии, отелей много, а используемые системы управления онлайн-бронированием отелей еще более разнообразны. Различные типы отелей не идеальны при использовании и не могут быть использованы в соответствии с личными потребностями; пользователи, даже если система будет разработана по индивидуальному заказу, она не сможет выполнять функции и эффекты, которые должна иметь информационная система. В настоящее время на рынке систем управления онлайн-заказами отелей моей страны мы можем провести углубленное обсуждение системы онлайн-управления отелями в соответствии с развитием сети.

Когда информационная система управления гостиницей только зарождалась, зарубежные страны уделяли больше внимания сочетанию управления гостиницей и информационными технологиями. Соединенные Штаты были первыми, кто принял новую систему управления гостиницей, а ЕСИ была первой, кто принял ее. Информационная система управления отелем. Эта система управления отелем в основном включает в себя управление клиентами, бронирование номеров и статистику данных. Чтобы сэкономить затраты на исследование и разработки, многие отели будут покупать приложения у других поставщиков или проводить вторичные разработки.

Во многих отраслях промышленности, особенно в тех, что представлены Соединенными Штатами, управление компьютерной информацией и сетевые приложения уже в то время опережали многие отрасли промышленности. Кроме того, в 1970-х годах зарубежные страны начали применять компьютеры в повседневной работе отелей. В ходе последующей популяризации и применения компьютерных технологий многие зарубежные отели предоставили гостям услуги глобального заказа, обеспечив круглосуточную функцию онлайн-заказа номеров.

Инновация бизнес-модели. В полной мере использовать информационные технологии для улучшения рабочего процесса управления, улучшения

управления и контроля сотрудников, а также повышения уровня обслуживания и экономических выгод. Анализ и прогноз группы клиентов, состояние развития отелей и будущие ситуации с помощью системы управления информацией, а также получены научные данные для поддержки разработки стратегий устойчивого развития.

1.2 Обзор существующих методов решения для онлайн бронирования отелей

Онлайн-система управления информацией отеля – это программное решение, которое помогает отелям управлять своими бизнес-операциями, отношениями с клиентами и ресурсами. Вот три примера распространенных онлайн-систем управления информацией в отелях, а также их плюсы и минусы.

Система управления недвижимостью Opera (PMS) (Как показано на рисунке 1)

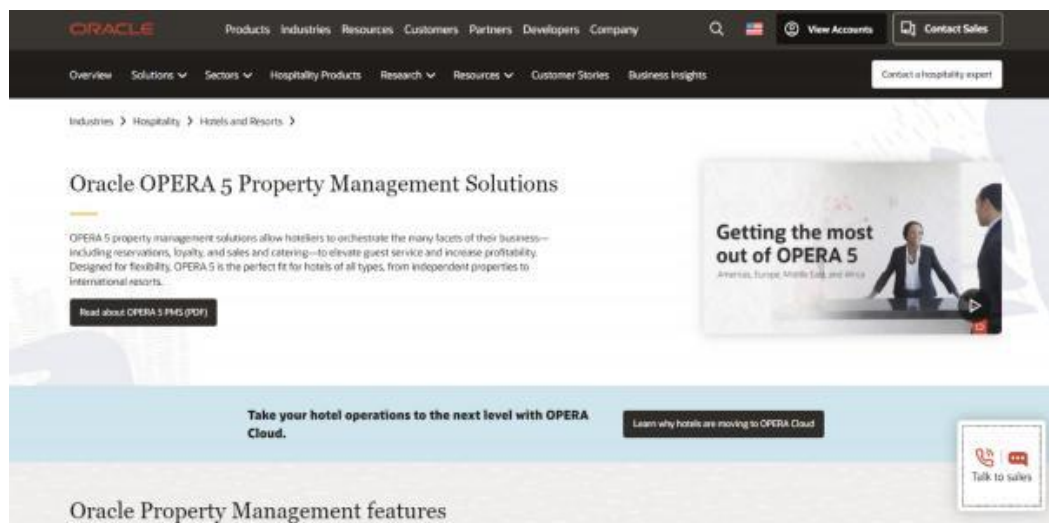


Рисунок 1 – Система управления недвижимостью Опера (PMS)

Преимущество:

- комплексность: опера pms предоставляет комплексные функции, включая управление стойкой регистрации, управление состоянием номеров, управление взаимоотношениями с клиентами, управление финансами и т. д;
- интеграция: может быть легко интегрирована с другими системами (такими как финансовые системы, системы продаж и маркетинга) для повышения общей операционной эффективности;

– функция отчетности. мощные функции анализа данных и отчетности помогают принимать решения на основе данных.

Недостатки:

– сложность: из-за обширного набора функций новым пользователям может быть сложно начать работу и потребуются длительное время на обучение;

– высокая стоимость: затраты на внедрение и обслуживание высоки, особенно подходят для крупных отелей или гостиничных сетей;

– зависимость от сети: для облачных решений нестабильные сетевые соединения могут влиять на использование системы.

Следующая система Хотелогикс (Как показано на рисунке 2)



Рисунок 2 – Хотелогикс

Преимущества:

– удобство для пользователя: интерфейс прост и интуитивно понятен, что позволяет новым пользователям легко начать работу;

– облачные решения. благодаря облачным системам пользователи могут получить доступ к системе в любое время и в любом месте, используя только подключение к интернету;

– гибкость: подходит для отелей разного размера: от небольших независимых отелей до крупных сетевых отелей.

Недостатки:

– функциональные ограничения. Некоторые расширенные функции могут быть не такими мощными, как у более зрелых систем на рынке (таких как Opera);

– служба поддержки клиентов. Время ответа службы поддержки клиентов может быть медленнее в часы пик;

– проблемы интеграции. Интеграция с другими несовместимыми системами может представлять определенные проблемы.

Комната Енота показана на рисунке 3.



Рисунок 3 – Комната Енота

Преимущества:

– автоматизация. Оснащен мощными функциями автоматизации, такими как интеллектуальное ценообразование и управление резервированием, для сокращения ручного вмешательства;

– обновления в режиме реального времени: статус номера, цена, доступность и другая информация обновляются в режиме реального времени для обеспечения точности данных;

– анализ данных. Встроенные инструменты анализа данных помогают

отелям оптимизировать стратегии ценообразования и маркетинговую деятельность.

Недостатки:

- функциональная сложность. несмотря на мощные функции, некоторым пользователям интерфейс и настройки функций могут показаться сложными;

- зависит от интернета. поскольку это онлайн-система, она должна полагаться на стабильное сетевое соединение;

- ограничения рынка. в некоторых областях проникновение на рынок может быть низким, а поддержка и ресурсы сообщества относительно ограничены.

Проектирование и внедрение системы управления информацией об онлайн-бронировании отелей на базе ASP.NET

Преимущества:

- это отечественное программное обеспечение, которое в определенной степени решает проблему недостаточной локализации и повышает адаптивность и настраиваемость отеля;

- плата за использование является относительно низкой, что удобно для отелей с ограниченным бюджетом и в определенной степени снижает экономическое давление на отели.

Недостатки:

- хотя эта система улучшилась по сравнению с вышеописанной, существует еще много областей для повышения эффективности управления информацией;

- отсутствие талантов в области управления информацией: в отеле не хватает профессиональных управленческих талантов в работе по управлению информацией, что приводит к неспособности предоставлять профессиональные решения при возникновении проблем.

Анализ существующих программных решений в гостиничной сфере выявил ряд значимых преимуществ у отечественных разработок:

– высокий уровень локализации и адаптируемости. Благодаря тому, что система разработана с учётом специфики отечественного рынка, она эффективно решает проблемы, связанные с локализацией. Программное обеспечение учитывает законодательные, языковые и культурные особенности, что повышает точность настройки и удобство использования в условиях локальной гостиничной индустрии. Это особенно важно для региональных отелей, где универсальные решения зарубежных производителей часто оказываются недостаточно гибкими;

– экономическая эффективность. Стоимость использования отечественной системы значительно ниже по сравнению с аналогами. Это делает её особенно привлекательной для малых и средних гостиниц с ограниченным бюджетом, снижая финансовую нагрузку и позволяя более рационально распределять ресурсы. Кроме того, отсутствие валютных рисков и зависимости от зарубежных поставщиков обеспечивает дополнительную стабильность в условиях экономической неопределённости;

– гибкость в масштабировании и интеграции. Система легко интегрируется с другими сервисами (CRM, системы управления персоналом, платёжные шлюзы), что позволяет отелям выстраивать единое цифровое пространство для управления бизнесом. Это особенно важно в условиях растущей конкуренции и необходимости оперативно реагировать на изменения рынка;

– поддержка и обслуживание. Наличие службы поддержки, регулярные обновления и адаптация под изменяющееся законодательство обеспечивают надёжную и долгосрочную эксплуатацию системы без дополнительных затрат на перевод или адаптацию интерфейса.

2 АЛГОРИТМИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ РЕШЕНИЯ ЗАДАЧИ

2.1 Общий алгоритм управления онлайн-бронирования

Система онлайн-бронирования отелей требует точного управления ресурсами, предотвращения конфликтов и обеспечения высокой производительности. В основе работы лежит ряд ключевых алгоритмов, которые решают фундаментальные задачи, включая обнаружение временных пересечений, управление инвентарем, обработку пользовательских запросов, контроль параллельных операций и защиту данных.

Для предотвращения двойного бронирования система реализует алгоритм проверки пересечения временных интервалов. Каждый новый запрос на бронирование сравнивается с существующими данными, и в случае обнаружения пересечений заявка отклоняется. В малонагруженных системах применяется последовательная проверка всех бронирований для данного номера, тогда как для масштабируемых решений используются структуры данных с упорядоченными интервалами для эффективного поиска пересечений.

Обработка пользовательских запросов, таких как фильтрация и сортировка номеров, обеспечивается алгоритмами быстрой сортировки и сортировки слиянием, что позволяет мгновенно упорядочивать данные по цене, рейтингу или другим параметрам. Для ускоренного поиска в отсортированных списках используется двоичный поиск, а для мгновенной выборки номеров по атрибутам (например, «номер с видом на море») применяется хэш-таблица.

Управление запасами номеров выполняется с учетом приоритетов и динамики спроса. Данный алгоритм позволяет оптимизировать распределение, ориентируясь на прибыльность и продолжительность бронирования. Контроль доступных номеров ведется с помощью алгоритма подсчета, обновляющего данные при каждом новом бронировании, что предотвращает избыточные продажи.

Для повышения производительности применяется механизм кэширования, где алгоритмы LRU (Least Recently Used) и FIFO (First In, First Out) управляют хранением данных, удаляя наименее востребованную информацию. Это ускоряет доступ к часто запрашиваемым отелям и улучшает пользовательский опыт.

Обработка больших объемов данных реализуется через пагинацию. Используются SQL-запросы с параметрами «LIMIT» и «OFFSET», а в случае высокой нагрузки – постраничное перемещение курсора, что снижает затраты на выборку данных.

Параллельные запросы бронирования требуют механизмов синхронизации. Оптимистическая блокировка применяется для предотвращения конфликтов путем проверки версий данных перед обновлением, тогда как пессимистическая блокировка блокирует строки базы данных («SELECT... FOR UPDATE»), исключая возможность одновременного изменения информации несколькими пользователями.

Проверка и обработка введенных пользователем данных обеспечивается механизмами регулярных выражений, которые валидируют адреса электронной почты, номера телефонов и форматы дат, предотвращая ошибки при регистрации и оформлении бронирования.

Организация работы персонала, включая уборку номеров и распределение задач, реализуется через алгоритмы поиска кратчайшего пути, такие как поиск в ширину (BFS), который определяет оптимальные маршруты для сотрудников, и метод кругового распределения, равномерно распределяющий задачи. Безопасность данных обеспечивается алгоритмами шифрования. Хеширование паролей (SHA-256) исключает возможность их расшифровки, а симметричное шифрование (AES) защищает конфиденциальные данные, такие как информация о платежах.

Таким образом, комплекс алгоритмов, задействованный в системе управления онлайн-бронированием отелей, гарантирует точность обработки

заявок, высокую производительность, надежное управление ресурсами и защиту пользовательских данных.

2.2 Функциональные модули и методы их управления

Функциональные модули системы представлены на рисунке 4. Система принимает дизайн разделения front-end и back-end, и вся система единообразно связана системным интерфейсом. Front-end пользовательская часть и back-end управление могут быть введены соответственно через функцию входа/регистрации. Архитектура имеет белый фон, черные линии разделяют модули, текст четко обозначен, а функциональные модули и таблицы базы данных соответствуют друг другу посредством соединительных линий.

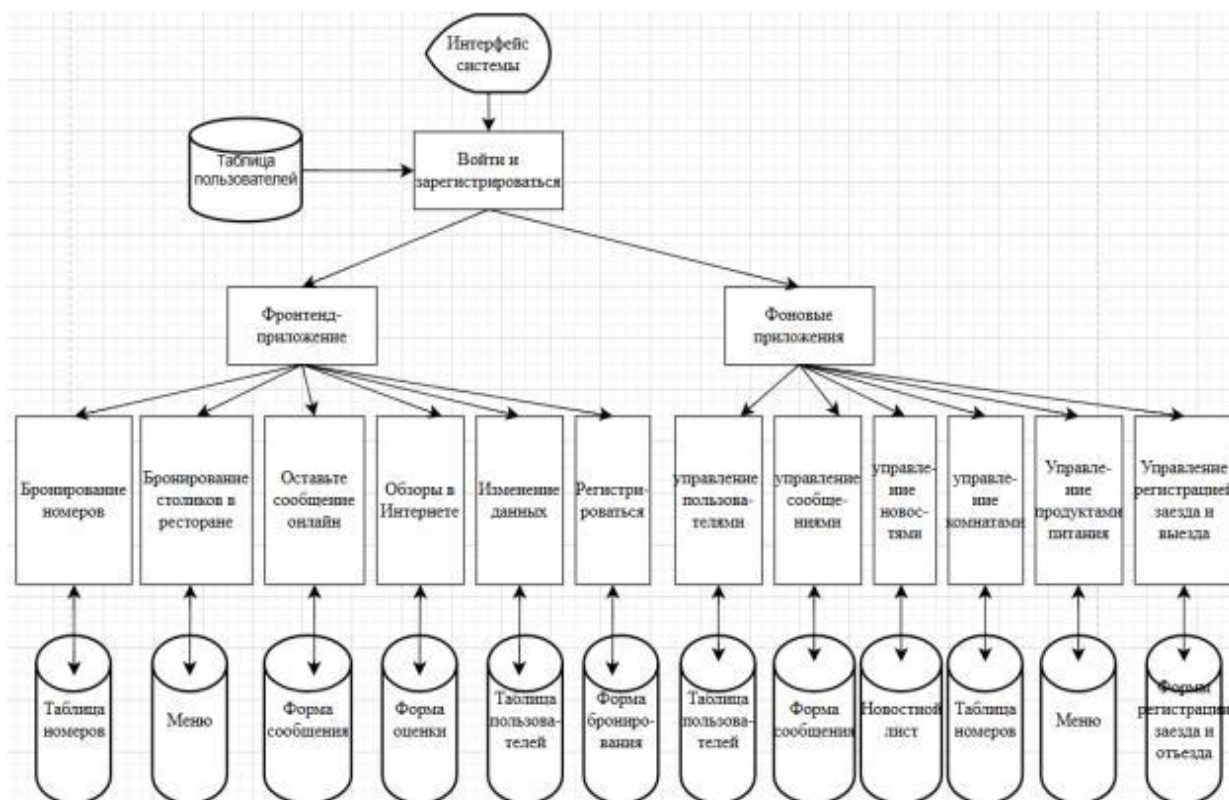


Рисунок 4 – Функциональные модули системы

Фронтенд-приложение для онлайн-бронирования отелей состоит из нескольких ключевых функций, связанные с управлением номерами, бронированием, пользовательскими взаимодействиями и обработкой данных.

Модуль бронирования номеров. Этот модуль обеспечивает работу с системой резервирования и представление доступных номеров пользователям.

Он взаимодействует с «Таблицей номеров» в реляционной базе данных, содержащей сведения о категориях номеров, их стоимости, статусе доступности и связанных удобствах.

Функциональность включает:

- отображение списка доступных номеров с возможностью фильтрации по дате, цене, количеству мест и другим параметрам;
- обработку запросов на бронирование с проверкой пересечений дат и подтверждением резервирования;
- интеграцию с системой платежей для оформления предоплаты или полной оплаты проживания.

Модуль резервирования столиков в ресторане. Для управления бронированием столиков в ресторане используется логика, схожая с модулем номеров. Однако в данном случае система работает со «Списком блюд», который представляет меню ресторана и позволяет пользователям предварительно заказывать еду или формировать предпочтения.

Функциональность включает:

- просмотр доступных столиков с возможностью выбора времени бронирования;
- отображение меню ресторана с возможностью выбора блюд во время бронирования;
- интеграцию с системой заказов и оплат.

Модуль онлайн-коммуникаций. Функционал онлайн-общения реализуется через «Форму сообщения», которая позволяет пользователям отправлять вопросы в службу поддержки, общаться с администрацией отеля или делать запросы на дополнительные услуги.

Функциональность включает:

- чат с администрацией для оперативного взаимодействия;
- отправку и обработку сообщений через централизованную систему поддержки;
- взаимодействие с бэкендом для хранения истории переписки.

Модуль онлайн-отзывов и комментариев. Система отзывов реализуется через «Форму оценки», которая позволяет пользователям оставлять комментарии о проживании, оценивать уровень сервиса и оставлять рекомендации для других гостей.

Функциональность включает:

- оставление отзывов и выставление рейтингов номерам, ресторану и другим услугам отеля;
- модерацию комментариев перед публикацией;
- подключение к аналитической системе для сбора данных о пользовательском опыте.

Модуль управления пользователями. Этот модуль отвечает за аутентификацию, регистрацию и управление аккаунтами клиентов. Взаимодействует с «Таблицей пользователей», в которой хранятся данные о зарегистрированных клиентах, их бронированиях, истории заказов и предпочтениях.

Функциональность включает:

- регистрацию новых пользователей и верификацию учетных данных;
- восстановление паролей и управление персональными настройками;
- синхронизацию с системой лояльности и бонусных программ.

Модуль управления регистрацией заезда и выезда. Контроль процессов заселения и выселения гостей обеспечивается через «Таблицу регистрации», а также взаимодействует с «Формой регистрации заезда» и «Формой регистрации выезда».

Функциональность включает:

- автоматическую регистрацию при заезде с возможностью предварительной онлайн-регистрации;
- генерацию цифровых ключей для доступа в номер;
- автоматическое списание оплаты или расчет дополнительных услуг при выезде.

Модуль управления закулисными процессами. Этот раздел системы от-

вечает за внутренние процессы отеля, включая администрирование пользователей, управление контентом и обработку заказов.

Управление пользователями. Администраторы отеля используют систему управления клиентскими учетными записями, взаимодействующую с «Таблицей пользователей», для настройки прав доступа, мониторинга активности клиентов и управления бронированиями.

Управление сообщениями. Обработка клиентских запросов через «Форму сообщения» позволяет персоналу быстро реагировать на обращения, модерировать сообщения и отслеживать уровень удовлетворенности клиентов.

Управление новостями и объявлениями. Система взаимодействует с «Таблицей новостей», где хранятся актуальные объявления, акции и обновления. Контент автоматически публикуется в интерфейсе приложения и отправляется клиентам через уведомления.

Управление номерным фондом. Связано с «Таблицей номеров», где хранятся данные обо всех номерах, их статусах и бронированиях. Включает функции обновления информации, автоматического расчета загрузки отеля и перераспределения номеров в случае непредвиденных ситуаций.

Управление ресторанным сервисом. Система управления меню и заказами работает через «Список продуктов питания», позволяя ресторанному персоналу обновлять блюда, контролировать запасы и получать предварительные заказы от гостей. Блок-схема входа пользователя показана на рисунке 5.



Рисунок 5 – Процесс входа пользователя в систему

Чтобы обеспечить безопасность системы, пользователи должны ввести имя пользователя, пароль, разрешения и код подтверждения на странице входа, а затем нажать кнопку «Войти», чтобы войти на главную страницу системы. Интерфейс онлайн-сообщений показан на рисунке 6.

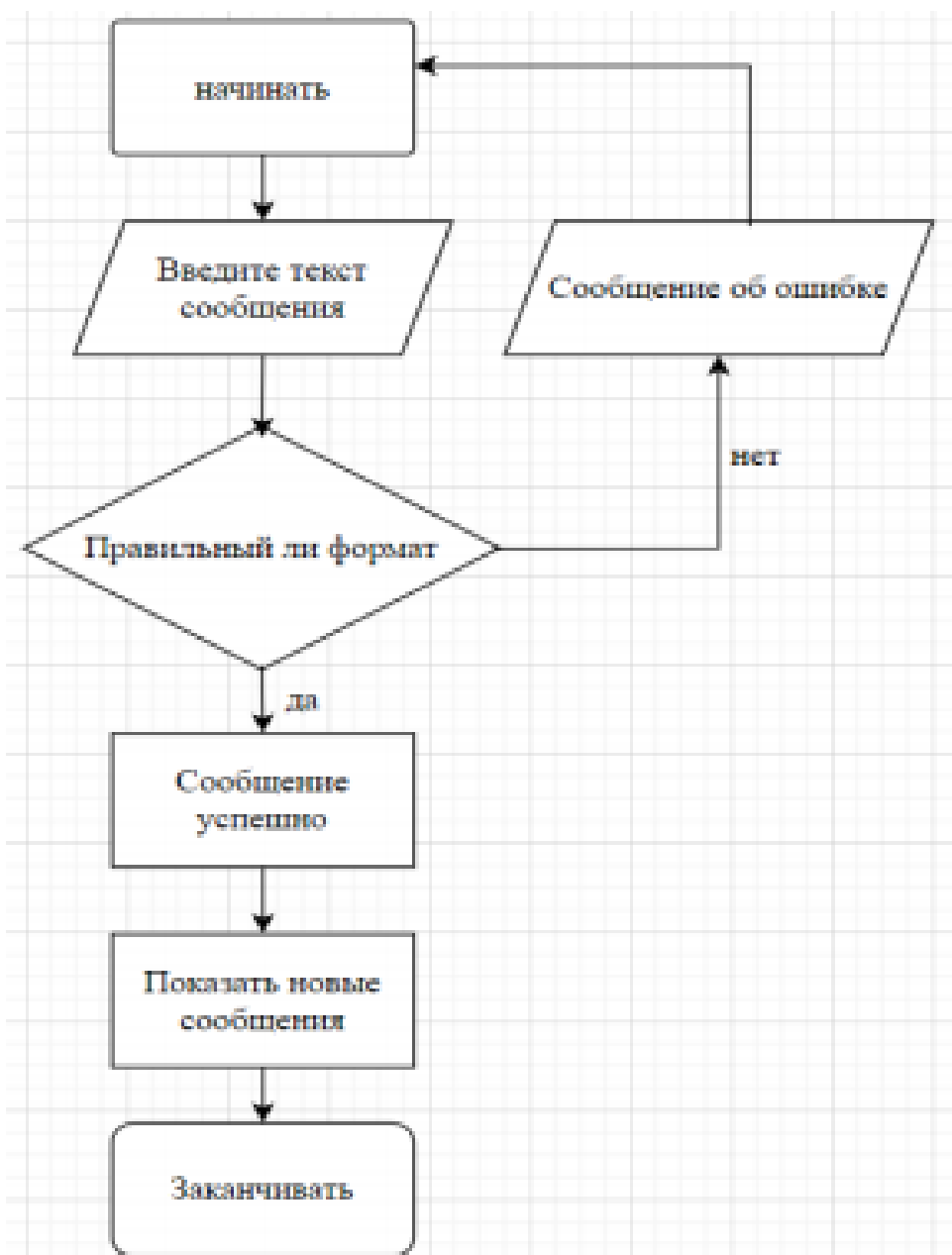


Рисунок 6 – Процесс онлайн-сообщения

Через «Доску сообщений» вы можете видеть все комментарии, вы можете оставлять комментарии через «Опубликовать сообщение», а администратор может поддерживать все комментарии через «Управление сообщениями». Процесс резервирования питания показан на рисунке 7.



Рисунок 7 – Процесс бронирования питания

После того, как менеджер объявит меню, меню появится перед системой, позволяя пользователям просмотреть меню, название меню, тип, цену; характеристики, характеристики, инвентарь; фотографии, подробную информацию и выбрать соответствующий ресторан для бронирования. используйте идентификатор для передачи параметров и поиска ресторанов, отправьте форму бро-

нирования еды и создайте запись о резервировании еды. Процесс бронирования номера показан на рисунке 8.



Рисунок 8 – Процесс бронирования номера

После того, как администратор отправит информацию о комнате, ком-

ната появится перед системой, и пользователи смогут просмотреть номер комнаты, расположение комнаты, тип комнаты и т. д. Цена, конфигурация оборудования, описание, необходимо выбрать соответствующее бронирование комнаты, использовать идентификатор для передачи параметров для поиска комнаты, отправить форму бронирования в add.aspx и создать запись резервирования. Блок-схема управления пользователями показана на рисунке 9.



Рисунок 9 – Схема управления пользователями

Администратор нажимает кнопку «Управление пользователями», чтобы войти в систему, удалить, изменить, использовать SESSION для получения новой учетной записи, затем ввести новую учетную запись и синхро-

низировать ее с базой данных для обновления пароля. Процесс управления питанием показан на рисунке 10.

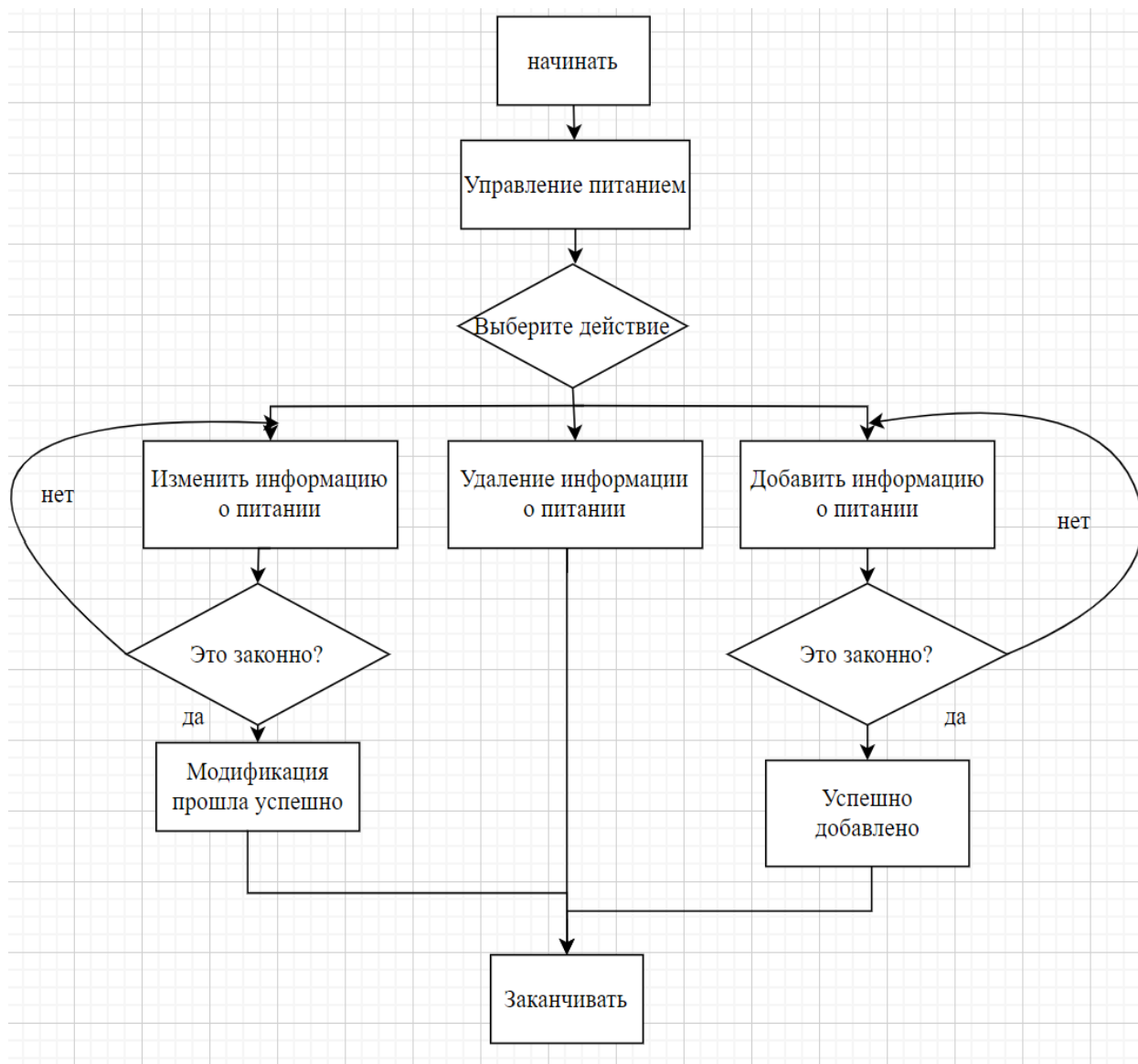


Рисунок 10 – Схема управления питанием

Менеджеры могут напрямую войти на веб-страницу публикации ресторана, нажав кнопку «Выпуск общественного питания», и загрузить соответствующую информацию о ресторане. Содержимое в основном включает в себя: название блюда, тип, цену; характеристики, характеристики, инвентарь и т. д., прежде чем отправить его в Ресторан.

После отправки ресторана таблица информации о ресторане в базе данных обновит данные, и система отобразит информацию о ресторане. Интерфейс добавления гостевой комнаты показан на рисунке 11.

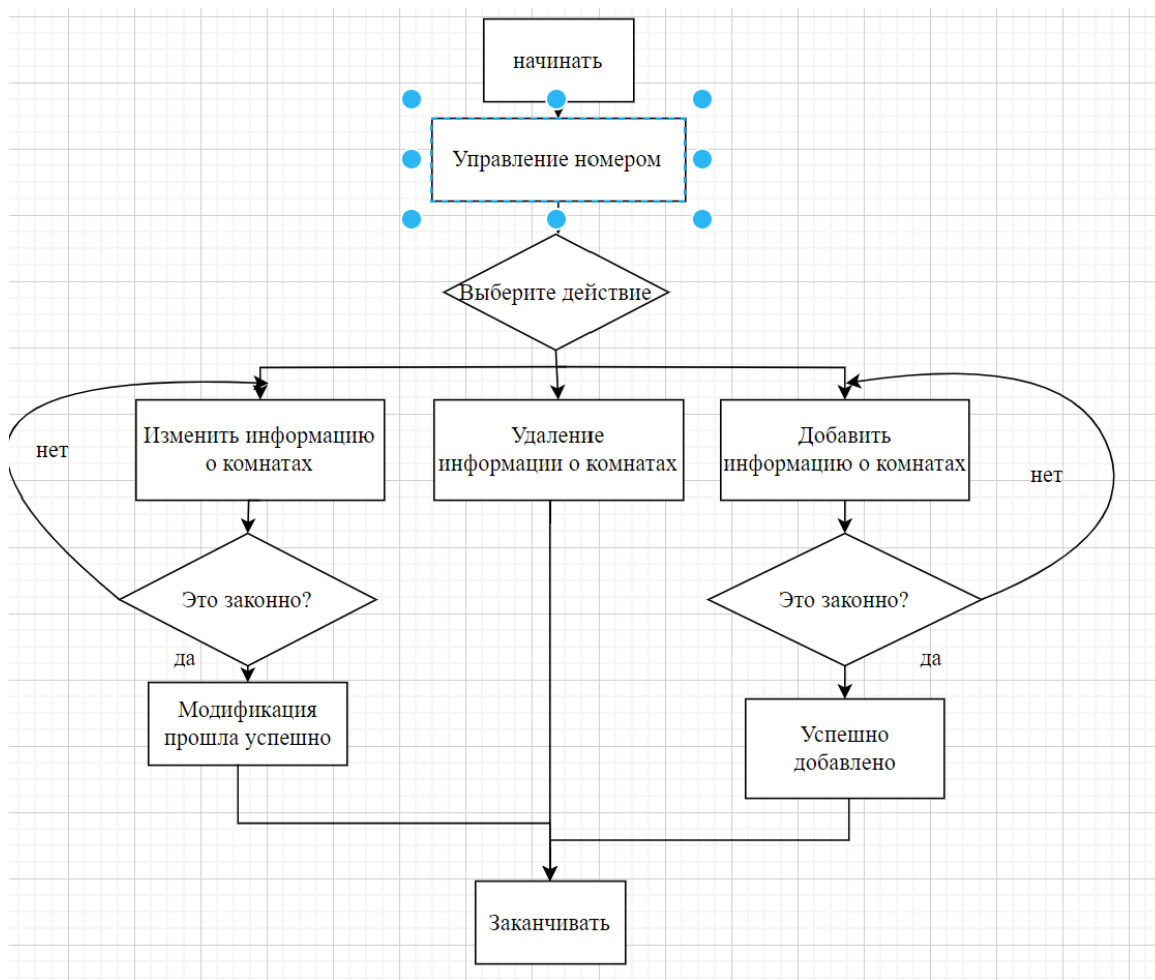


Рисунок 11 – Процесс управления помещением

Управление комнатами в основном включает в себя добавление, удаление и изменение данных о комнатах; запрос списка доступных комнат и другие операции, связанные с обновлением информации. К числу таких операций относятся, например, обновления, требующие синхронизации с пользовательским интерфейсом (обновления представления), а также обратная связь от пользователей или других компонентов системы.

Кроме того, процесс управления комнатами может включать в себя проверку доступности, назначение параметров (таких как вместимость, статус, тип), управление бронированием, а также взаимодействие с другими подсистемами, такими как система уведомлений или расписания.

Процесс управления регистрацией и обновлением комнат подробно представлен на рисунке 12.

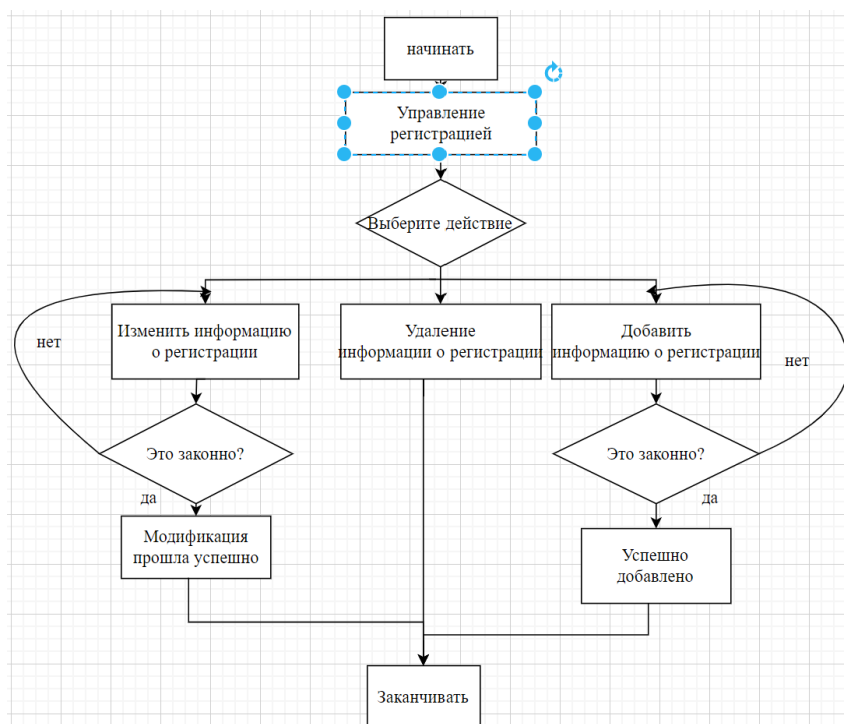


Рисунок 12 – Процесс управления регистрацией

С помощью кнопки «Регистрация» сотрудники могут заполнить информацию о регистрации на главной странице, включая номер комнаты, время регистрации, имя пользователя и т. д. После заполнения информация о регистрации в базе данных будет обновиться, и отобразится соответствующая информация о регистрации. Процесс управления выездом показан на рисунке 13.

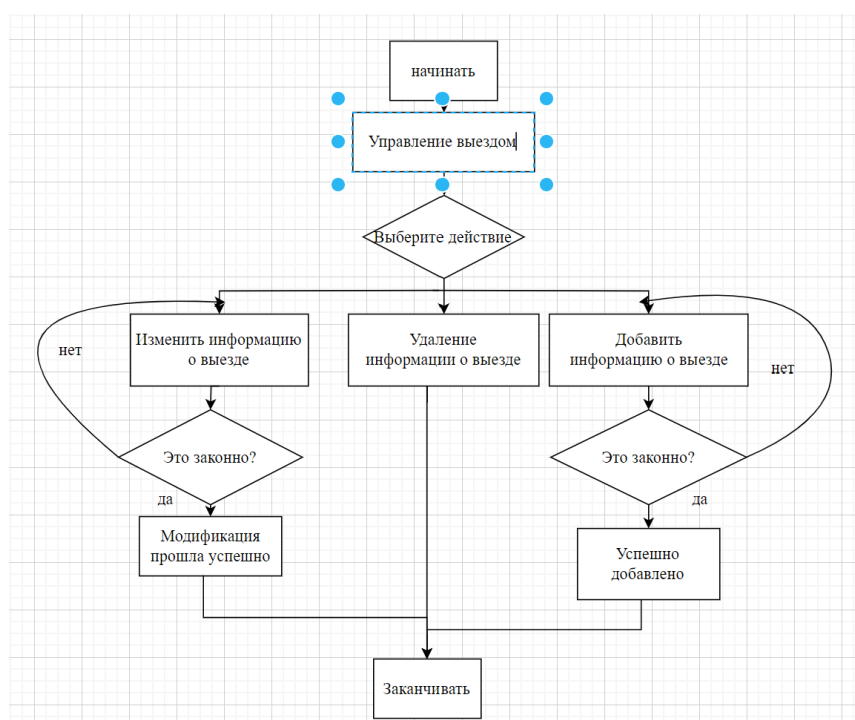


Рисунок 13 – Процесс управления выездом

После нажатия кнопки «регистрация выезда» сотрудники могут войти на веб-страницу, где размещено уведомление о выезде, и указать номер комнаты, пользователя, время выезда и другое содержимое; после завершения процедур выезда, оплата и т. д., информация в базе данных будет. Информация о выезде будет обновлена, и отобразится информация о выезде системы.

2.3 Обзор возможностей профильного программного обеспечения

Тестирование черного ящика: процесс тестирования обычно выглядит так: проект работает как нерушимый черный ящик, и вам нужно просто запускать его шаг за шагом. Чтобы судить о нормальной работе программы, необходимо проверить правильность входной и выходной информации и полноту выходной информации. Однако результаты таких испытаний постоянно меняются, и для получения наиболее эффективных результатов испытаний необходимо выбирать репрезентативную входную информацию.

Тестирование методом «белого ящика»: все, что нужно сделать, это ввести информацию, понаблюдать, правильно ли информация выполняется в соответствии с программой, проверить, что каждый узел приложения может правильно отражать передачу информации, а исчерпывающее тестирование белого ящика может передавать информацию по разным путям.

Тестирование методом «белого ящика» может показаться надежным, но на самом деле оно таит в себе скрытые опасности по трем причинам. Во-первых, программа просто неправильная, и ошибок в программе вообще не видно. Во-вторых, пути в белом поле могут быть опущены, что приведет к полной неполноте, и, в-третьих, невозможно узнать об ошибках, связанных с данными.

Язык программирования C#. Язык прост и легок для изучения. Язык C# не сложен и его несложно изучить. Но это не значит, что язык C# не имеет коннотации. Только те, кто действительно его изучает, могут понять, что язык C# такой же глубокий и глубокий, как китайская культура. Людям, изучившим C++, будет проще изучить C#, поскольку многие базовые знания остаются теми же, например, операторы выбора, массивы и т. д. Можно сказать, что язык C# избавился от недостатков языка C++ и сохранил его суть.

Объектно-ориентированный. Это одно из самых привлекательных преимуществ языка C#, поскольку объектно-ориентированная модель ближе к мышлению человеческого мозга и более удобна для программирования. На самом деле, в реальной жизни мы тоже имеем дело с классами. Например, человек – это класс, конкретный студент – это конкретный экземпляр человека, а способность студента есть, говорить и т. д. – это метод. И это объектно-ориентированный подход.

Как и в других реляционных базах данных, иерархия хранения базы данных представляет собой сумму данных в трех столбцах: база данных, таблица и база данных. Система MySQL может управлять и обслуживать несколько репозиториев. Каждый репозиторий состоит из множества таблиц.

Каждая таблица хранит несколько записей. Каждая запись должна быть реализована на основе взаимосвязи между несколькими полями. Можно предположить, что каждая таблица представляет собой двумерный массив, каждый элемент в массиве является записью, и каждый элемент в каждой записи является записью. Таблицу также можно представить как плоскую таблицу, в которой каждая запись – это строка, а каждое поле – это столбец. Отображает логическую связь между базами данных, таблицами и столбцами.

IIS бесплатен. Для мест с небольшим количеством посетителей более уместен ситуативный подход. То есть после установки IIS на свой компьютер вы можете использовать его для анализа требований к доступу к предыдущим веб – страницам HTML.

Первоначально IIS был разработан корпорацией Microsoft, и по мере развития сервера IIS в последней версии были внесены существенные изменения и корректировки в сервер IIS, что сделало его работу более удобной. В этой статье представлен алгоритм балансировки нагрузки на основе IIS. Система не требует дополнительных инвестиций в оборудование и проста в внедрении, что делает ее особенно подходящей для разработки небольших коммерческих веб - сайтов. На основе текущей реальной ситуации он может служить справочным материалом для аналогичных приложений.

NET основан на общем языке и запускает программы на сервере. Базовая база данных, механизм сообщений и обработка интерфейса данных общего языка могут быть хорошо интегрированы в сеть.

Преимуществами .NET являются высокая эффективность, возможность повторного использования компонентов, безопасность и относительно небольшой объем кода. Платформа обеспечивает удобную поддержку объектно-ориентированного программирования, облегчает масштабирование и интеграцию различных сервисов.

В программировании ASP (классическом) кодирование осуществляется с использованием скриптового языка, при этом весь код встраивается непосредственно в HTML. Это приводит к значительному увеличению объема кода, снижению его читаемости и затрудняет сопровождение.

ASP.NET эффективнее и проще в использовании, чем ASP. Для той же функции ASP.NET также требует меньше кода.

Эта система использует структуру B/S, использует MySQL в качестве фона и использует компоненты команд для вызова и хранения SQL, чтобы реализовать вызов и хранение SQL. SQL это широко используемый инструмент связи с базами данных.

ANSI определяет SQL как стандартизированный язык работы с реляционными базами данных и использует его в качестве инструмента для выполнения различных операций, таких как обновление информации, извлечение данных, вставка новых записей и удаление существующих. SQL отличается простотой синтаксиса, высокой эффективностью и широким применением во всех современных СУБД. Он является универсальным средством взаимодействия с базами данных, что делает его неотъемлемой частью большинства информационных систем.

C# – это объектно-ориентированный язык программирования, разработанный Microsoft и используемый преимущественно в рамках платформы .NET. Он обеспечивает строгую типизацию, высокую безопасность

кода и автоматическое управление памятью (через сборщика мусора), но при этом не позволяет напрямую управлять памятью, как, например, C++.

C# и Java обладают всеми основными характеристиками объектно-ориентированного программирования, включая инкапсуляцию, наследование и полиморфизм, что делает их удобными для построения модульных, масштабируемых и сопровождаемых приложений.

C# хорошо подходит для разработки приложений с различными архитектурами: от клиент-серверных решений и настольных приложений до распределённых облачных систем и мобильных платформ, особенно в связке с такими технологиями, как Windows Forms, WPF, ASP.NET и Xamarin.

Таблица 1 – Преимущества ASP.NET

Преимущества	Содержание
1	2
Высокая производительность	Поддерживая модели асинхронного программирования, сервер Kestrel ASP.NET Core оптимизирует сценарии с высоким параллелизмом, а его производительность значительно превосходит производительность традиционных фреймворков.
Эффективность разработки	Интегрируется с мощной средой IDE Visual Studio, предоставляет автоматическое завершение кода, инструменты отладки и визуальные конструкторы, а также поддерживает быструю разработку.
Кроссплатформенный	ASP.NET Core поддерживает Windows, Linux и macOS, подходит для команд со смешанным технологическим стеком.
Безопасность	Встроенная аутентификация, защита от межсайтового скриптинга и подделки запросов в соответствии со стандартами безопасности корпоративного уровня.
Кроссплатформенный	ASP.NET Core поддерживает Windows, Linux и macOS, подходит для команд со смешанным технологическим стеком.
Экосистема	Огромная библиотека пакетов NuGet и официальная поддержка от Microsoft, обеспечивающая долгосрочную гарантию обслуживания.

Таблица 2 – Почему ASP.NET стоит использовать

Причины использования	Объяснение
1	2
Требования корпоративного уровня	Подходит для таких сценариев, как банки и государственные учреждения.
Стек технологий Microsoft	Если команда знакома с экосистемой C# и.NET, синергия цепочки инструментов может быть maximизирована.
Долгосрочная поддержка	ASP.NET Core – ключевой проект поддержки Microsoft, предоставляющий LTS (версию с долгосрочной поддержкой)
Высокопроизводительные сценарии	Должны обрабатывать запросы с высокой степенью параллелизма (например, приложения для связи в реальном времени) и могут сочетать SignalR и архитектуру микросервисов для достижения отклика с малой задержкой.
Стек технологий Microsoft	Если команда знакома с экосистемой C# и.NET, синергия цепочки инструментов может быть maximизирована.

Таблица 3 – Преимущества MySQL

Преимущества	Содержание
1	2
Открытый исходный код и бесплатно	Community Edition (MySQL Community Edition) полностью бесплатен и подходит частным лицам и малому бизнесу для сокращения расходов на разработку.
Кроссплатформенная поддержка	Поддерживает несколько операционных систем, таких как Windows, Linux, macOS и т. д., с высокой степенью адаптивности.
Высокая производительность	Механизм запросов оптимизирован (например, InnoDB поддерживает блокировки на уровне строк).
Простота использования	Синтаксис близок к стандартному SQL, а стоимость обучения низкая.
Безопасность данных	Поддерживает передачу данных с шифрованием SSL.
Масштабируемость	Поддерживает вертикальное расширение (обновление оборудования) и горизонтальное расширение и может обрабатывать данные на уровне ТБ.

Таблица 4 – Преимущества Visual Studio

Преимущества	Содержание
1	2
Полнофункциональная интегрированная среда разработки	Поддерживает несколько языков программирования (C#, C++, Python, JavaScript и т. д.) и типов разработки (веб, настольные, мобильные, облачные).
Эффективное написание кода	Интеллектуальное завершение кода IntelliSense, рефакторинг кода и сообщения об ошибках в реальном времени
Цепочка инструментов корпоративного уровня	Встроенная среда модульного тестирования, инструменты управления базами данных для удовлетворения потребностей сложных проектов.
Интеграция облака и ИИ	Глубокая интеграция облачных сервисов Azure для поддержки разработки приложений на основе ИИ.

Таблица 5 – Почему Visual Studio стоит использовать

Преимущества	Содержание
1	2
Разработка приложений корпоративного уровня	Подходит для крупномасштабных приложений, систем разработки настольных ERP или финансового программного обеспечения, полагающихся на экосистему C#/.NET.
Требования к кроссплатформенности	Если вам необходимо одновременно создавать приложения для Windows, Linux и мобильных терминалов, вы можете использовать Xamarin, .NET MAUI или ASP.NET Core для повторного использования кода.
Разработка в облаке	Разрабатывайте микросервисы (с поддержкой Docker и Kubernetes) или развертывайте их в облаке Azure, напрямую используя встроенные цепочки инструментов для упрощения процесса.
Эффективное взаимодействие в команде	Используйте Azure DevOps для управления спросом, проверки кода и автоматизированного развертывания, что подходит для гибких групп разработки.
Обучение и поддержка сообщества	Официальная документация Microsoft завершена, а сообщество активно, что позволяет разработчикам учиться и решать проблемы.

Системы онлайн-бронирования, предназначенные для корпоративного сегмента и крупных веб-приложений, должны обеспечивать высокую степень интеграции с внешними сервисами, поддерживать безопасность транзакций и эффективно справляться с нагрузкой. В подобных системах часто используются технологии, связанные с экосистемой Microsoft, такие как ASP.NET Core, поскольку они обеспечивают гибкость, масштабируемость и высокую производительность. Современные тенденции в разработке подтверждают, что ASP.NET Core стал доминирующим стандартом, в то время как традиционный ASP.NET постепенно утрачивает актуальность.

При проектировании систем бронирования важно учитывать особенности хранения и обработки данных. Для управления транзакциями и поддержки высоконагруженных систем предпочтительно использовать реляционные базы данных, такие как MySQL 8.0. Однако в сценариях, связанных с анализом больших объёмов данных, обработкой сложных JSON-структур или геоинформационных систем, более целесообразно рассматривать альтернативные решения, такие как Hadoop, ClickHouse или PostgreSQL с PostGIS.

Разработка веб-приложений для онлайн-бронирования требует эффективных инструментов программирования, особенно когда проект ориентирован на командную работу, облачную инфраструктуру или кроссплатформенные решения. В этом контексте Visual Studio 2022 выступает в качестве мощного средства разработки, предлагая интеграцию с системами контроля версий, поддержку многопоточной обработки и оптимизацию кода на базе искусственного интеллекта. Однако для лёгкого редактирования или написания небольших скриптов более предпочтительны альтернативные инструменты, такие как Visual Studio Code.

Внедрение автоматизированных алгоритмов в систему бронирования играет ключевую роль в управлении ресурсами. Использование оптимизированных механизмов сортировки, быстрой проверки доступности номеров и предсказательной аналитики помогает улучшить пользовательский опыт и предотвратить конфликты при бронировании. Современные тенденции показывают,

что онлайн-бронирование становится всё более сложной системой, требующей гибкости, интеграции с облачными сервисами и поддержки распределённых вычислений.

Дополнительно, всё большую значимость приобретает внедрение алгоритмов машинного обучения, позволяющих анализировать поведенческие паттерны пользователей и предлагать персонализированные рекомендации. Это способствует увеличению коэффициента конверсии и повышает лояльность клиентов. Интеграция с внешними API – такими как платёжные шлюзы, системы управления отелями (PMS), и платформы отзывов – позволяет создавать единое цифровое пространство, в котором данные синхронизируются в реальном времени.

Не менее важным аспектом является обеспечение безопасности: шифрование персональных данных, защита от DDoS-атак и соответствие требованиям стандартов, таких как GDPR и PCI DSS, становятся неотъемлемыми элементами системы. В условиях высокой конкуренции на рынке важным конкурентным преимуществом становится способность системы масштабироваться и адаптироваться под индивидуальные потребности бизнеса – от малых отелей до крупных сетевых операторов.

3 ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ПРЕДЛАГАЕМОГО АЛГОРИТМА РЕШЕНИЯ ЗАДАЧИ

3.1 Основные этапы практической разработки программного продукта

Анализ осуществимости. Экономическая осуществимость: производится оценка затрат на разработку системы (например, бесплатность базы данных MySQL и стека технологий ASP.NET) и ожидаемые выгоды (повышение эффективности управления гостиницей и снижение затрат на рабочую силу).

Социальная осуществимость: необходимо проанализировать принятие пользователем (на основе простоты использования структуры B/S) и ценность оптимизации бизнес-процессов.

Эксплуатационная осуществимость: проверка порогового значения действий пользователя (доступ через браузер не требует обучения).

Определение требований. Функциональные требования: разделение функциональных модулей на три роли: администратор, сотрудник и пользователь (например, бронирование номеров, управление отзывами, управление регистрацией заезда и выезда и т. д.). Требования к производительности: это требования к безопасности (шифрование хранилища MySQL), целостности (модуль замкнутого цикла) и работоспособности (на основе развертывания VS2010).

Выходные данные: спецификация требований, блок-схема системы (описывающая процесс работы пользователя).

Фаза проектирования системы. Проектирование архитектуры системы. Применение многоуровневой архитектуры (архитектуру B/S) для уточнения иерархии системы.

Уровень представления: создание пользовательского интерфейса (веб - стороны) на основе платформы ASP.NET и поддержка доступа через браузер.

Уровень бизнес-логики: использование C# для разработки базовой биз-

нес-логики (например, проверки регистрации пользователя, правил бронирования номеров).

Уровень доступа к данным: подключение к базе данных MySQL через ADO.NET или Entity Framework для добавления, удаления, изменения и запроса данных.

Архитектура развертывания: использование IIS (Internet Information Services) в качестве веб-сервера для поддержки одновременного доступа и балансировки нагрузки.

В рамках реализации системы была применена концепция модульного проектирования, что позволило обеспечить её логичную структуру, масштабируемость и удобство сопровождения. Вся функциональность системы распределена по ключевым подсистемам, каждая из которых отвечает за определённый сегмент бизнес-логики.

Пользовательская подсистема включает в себя модуль регистрации и входа, обеспечивающий аутентификацию и авторизацию пользователей. Также в состав данной подсистемы входит модуль сообщений, предоставляющий возможность публикации текстовых сообщений с автоматической фиксацией временных меток. Дополнительно реализован модуль отзывов, позволяющий пользователям оставлять оценки и комментарии относительно номеров или предоставляемых услуг.

Подсистема управления гостевыми комнатами отвечает за организацию информации о жилом фонде и его бронирование. Модуль информации о номерах обеспечивает управление категориями и ценами, включая, например, стандартные и улучшенные номера. Модуль бронирования реализует функциональность онлайн-бронирования и отслеживания статуса заказов, включая информацию об их оплате.

Управление питанием реализовано в рамках отдельной подсистемы, в которую входят два модуля. Первый отвечает за ведение справочника вариантов питания и их стоимости, второй – за реализацию процедуры бронирования

и отслеживание статуса оплаты, по аналогии с подсистемой управления номерами.

Бэкэнд-подсистема предназначена для администрирования и анализа. В неё входит модуль статистики, предназначенный для формирования отчётов о поведении пользователей, уровне загрузки помещений и других ключевых показателях. Также реализован модуль управления правами доступа, предусматривающий классификацию административных пользователей на различные уровни, включая суперадминистраторов и администраторов с ограниченными правами.

Наряду с функциональным моделированием была разработана концептуальная структура базы данных, обеспечивающая логичную организацию хранимой информации. Далее она была преобразована в логическую модель, соответствующую требованиям выбранной системы управления базами данных и предполагающей эффективную реализацию операций чтения, записи и обработки данных.

Модуль регистрации: пользователи могут регистрировать свои учетные записи на стойке регистрации системы. Информация, необходимая для регистрации учетной записи, включает: имя пользователя, пароль, пароль подтверждения; адрес электронной почты, QQ и т. д. После отправки информации для входа система определит, соответствует ли введенная пользователем информация спецификации на основе кода JS. Если она соответствует, то все в порядке; добавьте запись пользователя в базу данных системы. В целях безопасности алгоритм шифрования MD5 используется для шифрования пароля и последующего сохранения его в базе данных.

Модуль входа: в процессе входа пользователя, включая имя пользователя, пароль и код подтверждения, система должна сначала определить правильность кода подтверждения. Если пользователь входит в систему с помощью кода подтверждения, то пользователь должен ввести не только имя пользователя и пароль, но и код подтверждения при входе. После определения пра-

вильности кода подтверждения определяются имя пользователя и пароль. Поскольку пароль пользователя был зашифрован с помощью MD5, при определении пароля пользователя пароль должен быть преобразован в MD5 перед определением пароля пользователя; если имя пользователя или пароль неверны, на странице входа появится сообщение об ошибке.

Модуль управления сообщениями включает в себя подмодули, такие как сообщение, просмотр сообщения и ответ на сообщение. Пользователи и менеджеры в системе могут использовать эту функцию. Пользователи могут оставлять сообщения и просматривать ответы менеджеров, в то время как менеджеры могут просматривать сообщения обратной связи пользователей и отвечать на сообщения.

Модуль управления комментариями включает в себя подмодули, такие как онлайн-комментарии, просмотр комментариев и управление комментариями. Пользователи и менеджеры в системе могут использовать эту функцию. Пользователи могут оставлять комментарии и просматривать комментарии других пользователей, в то время как менеджеры могут находить все записи комментариев и удалять или изменять их.

Модуль управления питанием включает в себя подмодули ввода питания и просмотра питания. Пользователи и менеджеры могут использовать этот функциональный модуль. Только менеджеры могут добавлять, изменять или удалять информацию о питании, в то время как другие пользователи могут только просматривать информацию о питании.

Модуль управления гостевыми комнатами включает в себя подмодули ввода питания и просмотра питания. Пользователи и менеджеры могут использовать этот функциональный модуль. Только менеджеры могут добавлять, изменять или удалять информацию о гостевых комнатах, в то время как другие пользователи могут только просматривать информацию о гостевых комнатах.

Модуль управления регистрацией включает подмодули входа и просмотра регистрации. Функциональный модуль могут использовать и пользо-

ватели, и менеджеры. Только менеджеры могут добавлять, изменять или удалять информацию о регистрации, а другие пользователи могут только просматривать информацию о регистрации.

Модуль управления регистрацией включает подмодули входа и просмотра регистрации. Функциональный модуль могут использовать и пользователи, и менеджеры. Только менеджеры могут добавлять, изменять или удалять информацию о регистрации, а другие пользователи могут только просматривать информацию о регистрации.

Модуль управления сотрудниками включает подмодули входа и просмотра регистрации сотрудников. Функциональный модуль могут использовать и пользователи, и менеджеры. Только менеджеры могут добавлять, изменять или удалять информацию о регистрации сотрудников, а другие пользователи могут только просматривать информацию о регистрации сотрудников.

Три наиболее важными элементами диаграммы ER являются сущности, атрибуты и отношения. Диаграмма ER состоит из этих трех точек. Диаграмма ER – лучшая система для связи между различными сущностями, включая атрибуты каждой сущности.

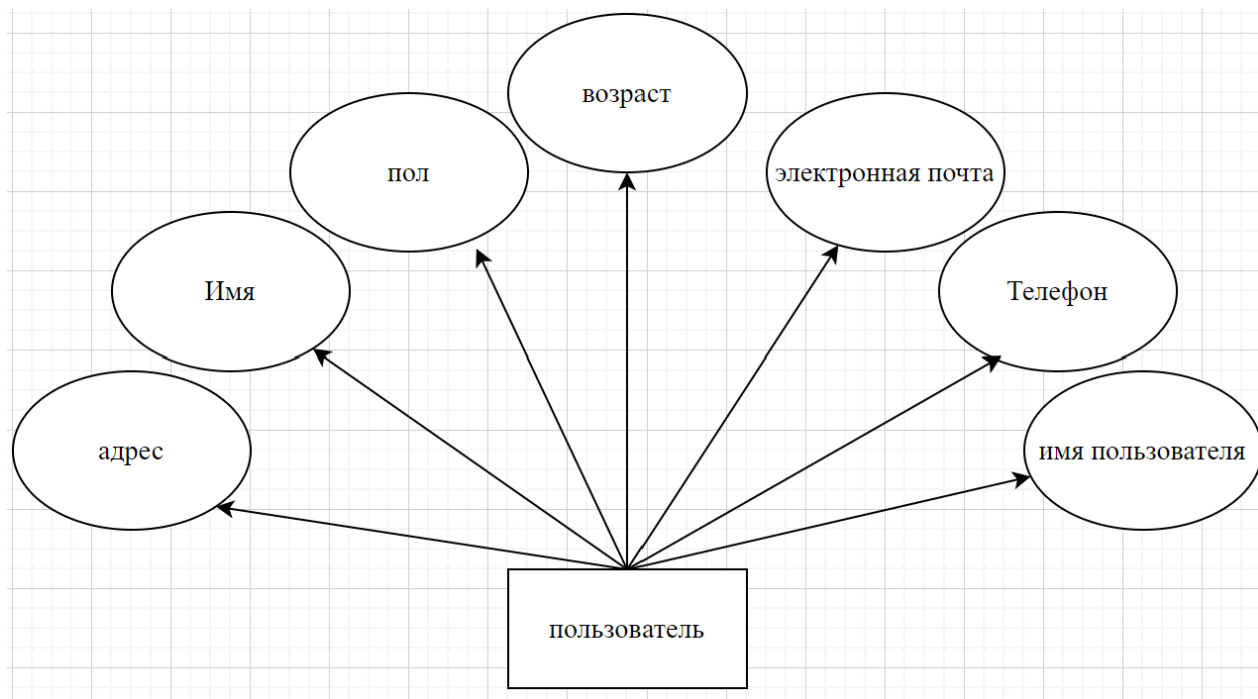


Рисунок 14 – Диаграмма атрибутов объекта пользователя



Рисунок 15 – Диаграмма атрибутов объекта гостевой комнаты



Рисунок 16 – Диаграмма атрибутов предприятия общественного питания

Рисунок 17 – Интерфейс входа пользователя

Пользователь должен ввести имя пользователя, пароль, полномочия, проверочный код на странице входа, а затем нажать кнопку «Войти» для входа на главную страницу системы.

Пользователь: Лицо, управляющее процессом входа в систему.

Страница входа: интерфейс, где пользователи вводят информацию.

Процесс взаимодействия: Пользователь последовательно вводит имя пользователя, пароль, разрешение и проверочный код на странице входа.

После того, как пользователь нажимает кнопку «Войти», страница входа отправляет информацию в систему для проверки.

Система возвращает информацию на основе результата проверки: если проверка прошла успешно, страница входа перенаправит пользователя на главную страницу системы.

The screenshot shows a web interface for a restaurant reservation system. At the top, there is a header with the title "餐饮预定" (Restaurant Reservation) in purple. Below the header, there is a navigation bar with five links: "修改个人信息" (Modify Personal Information), "修改登录密码" (Modify Login Password), "我的预订信息" (My Reservation Information), "我的入住信息" (My Check-in Information), and "我预定的餐饮" (My Reserved Restaurant). The main content area contains a reservation form. It starts with "预约日期:" (Reservation Date) followed by a text input field. Below that is "时间段:" (Time Period) followed by another text input field. To the left of a list of food items is the label "选择菜品:" (Select Dish). The list consists of ten items, each with a checkbox and a label: "餐品1" through "餐品10".

Рисунок 18 – Диаграмма модуля бронирования ресторана

Администратор публикует информацию о комнатах: Администратор публикует информацию о комнатах, включая номер комнаты, местоположение, тип, цену, конфигурацию оборудования и описание и т. д.

Информация о комнате отображается на интерфейсе системы: Интерфейс системы получает информацию о комнате, опубликованную администратором, и отображает ее пользователю.



Рисунок 19 – Интерфейс бронирования номеров

Пользователи просматривают информацию о комнате: пользователи могут просматривать подробную информацию о комнате на интерфейсе системы, включая номер комнаты, местоположение, тип, цену, конфигурацию оборудования и описание и т. д.

Выберите номер для бронирования: пользователь выбирает номер и нажимает кнопку «Забронировать», а система ищет соответствующую информацию о номере по идентификатору номера.

Используйте идентификатор для поиска комнаты: система ищет соответствующую информацию о комнате в бэкенде на основе идентификатора комнаты, выбранного пользователем.

Отправьте форму бронирования на add.aspx: Пользователь заполняет

форму бронирования (например, время заезда, время выезда, контактную информацию и т. д.) и отправляет ее на страницу `add.aspx` на внутренней стороне через интерфейс системы.

Создание записей о бронировании: после того, как внутренняя служба получает форму бронирования, она создает записи о бронировании и сохраняет их в базе данных.

Отображение информации об успешном бронировании: интерфейс системы отображает пользователю информацию об успешном бронировании.

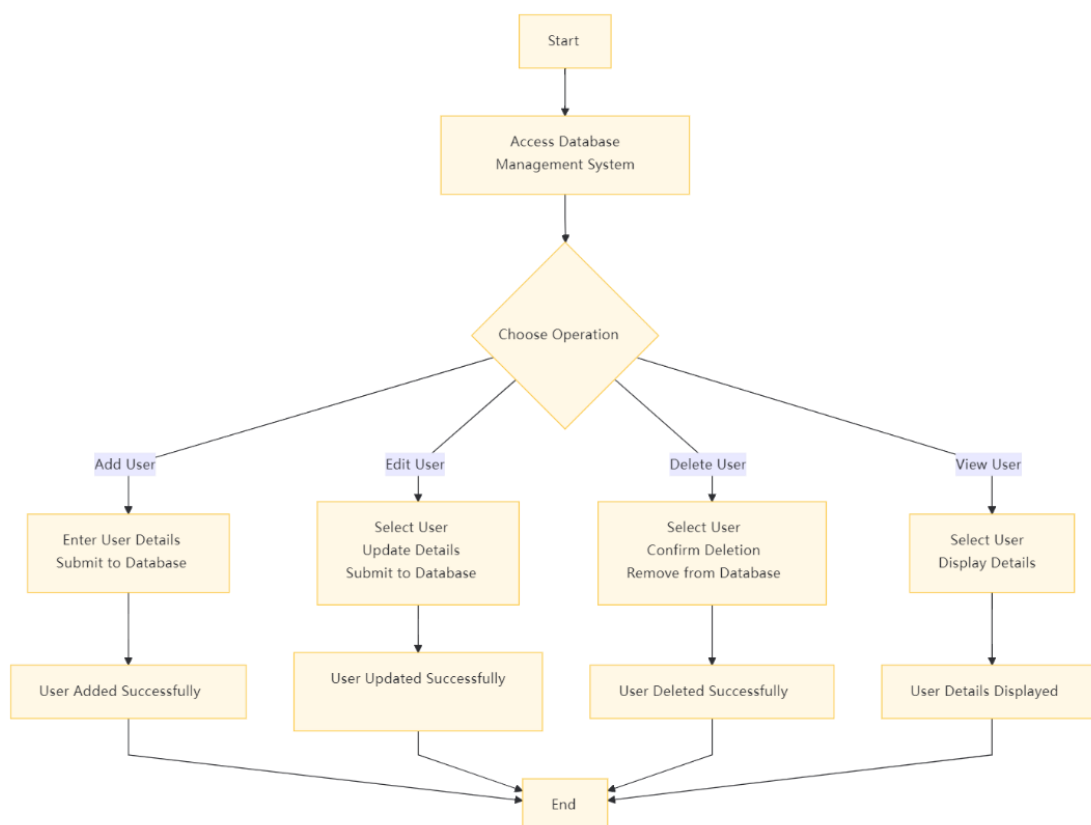


Рисунок 20 – Диаграмма активности бронирования номеров

Класс `Admin`: содержит свойства имени пользователя и пароля администратора; предоставляет методы для входа в систему, удаления пользователей, изменения пользователей, создания пользователей и обновления паролей.

Класс `UserManagementSystem`: содержит список пользователей и используется для управления пользователями; предоставляет администраторам методы входа в систему, удаления пользователей, изменения пользователей, создания пользователей и обновления паролей, а также управления сессиями.

Класс сеанса: содержит идентификатор сеанса и имя пользователя; предоставляет методы для создания и уничтожения сеансов.



Рисунок 21 – Модуль управления пользователями UML-диаграмма

Класс Admin: содержит свойства имени пользователя и пароля администратора; предоставляет методы для входа в систему и загрузки информации о ресторане.

Класс RestaurantInfo: содержит информацию, связанную с рестораном, такую как названия блюд, типы, цены, спецификации, особенности, инвентарь, фотографии и подробности.

Класс интерфейса: предоставляет методы для отображения страницы

входа, страницы загрузки и информации о ресторане.

Бэкэнд-класс: предоставляет методы для аутентификации администратора, обработки загруженной информации и обновления базы данных.

Класс базы данных: предоставляет методы для обновления информации о ресторане.

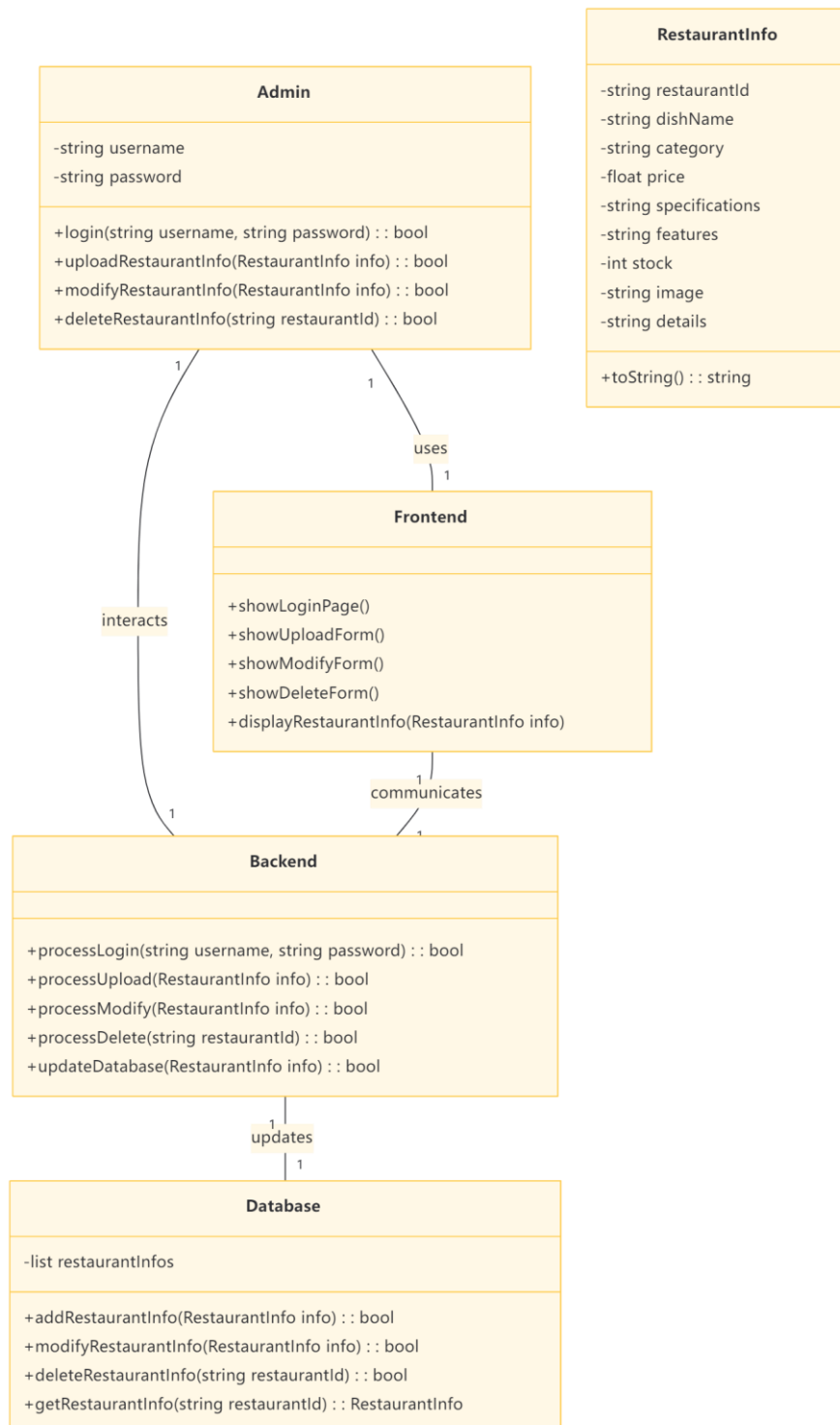


Рисунок 22 – Модуль управления пользователями UML-диаграмма

Класс RoomManager: предоставляет основные функции управления комнатами, включая добавление комнат, удаление комнат, изменение комнат, запрос комнат и обновление комнат; предоставляет функцию обратной связи для отображения результатов работы пользователям.

Класс номера: представляет собой основную информацию о номере, включая идентификатор номера, тип номера, местоположение, цену, инвентарь и статус.

Класс BusinessLogicLayer: обрабатывает бизнес-логику управления комнатами, такую как проверка входных данных, обработка бизнес-правил и т. д.; предоставляет интерфейс доступа к слою логики данных.

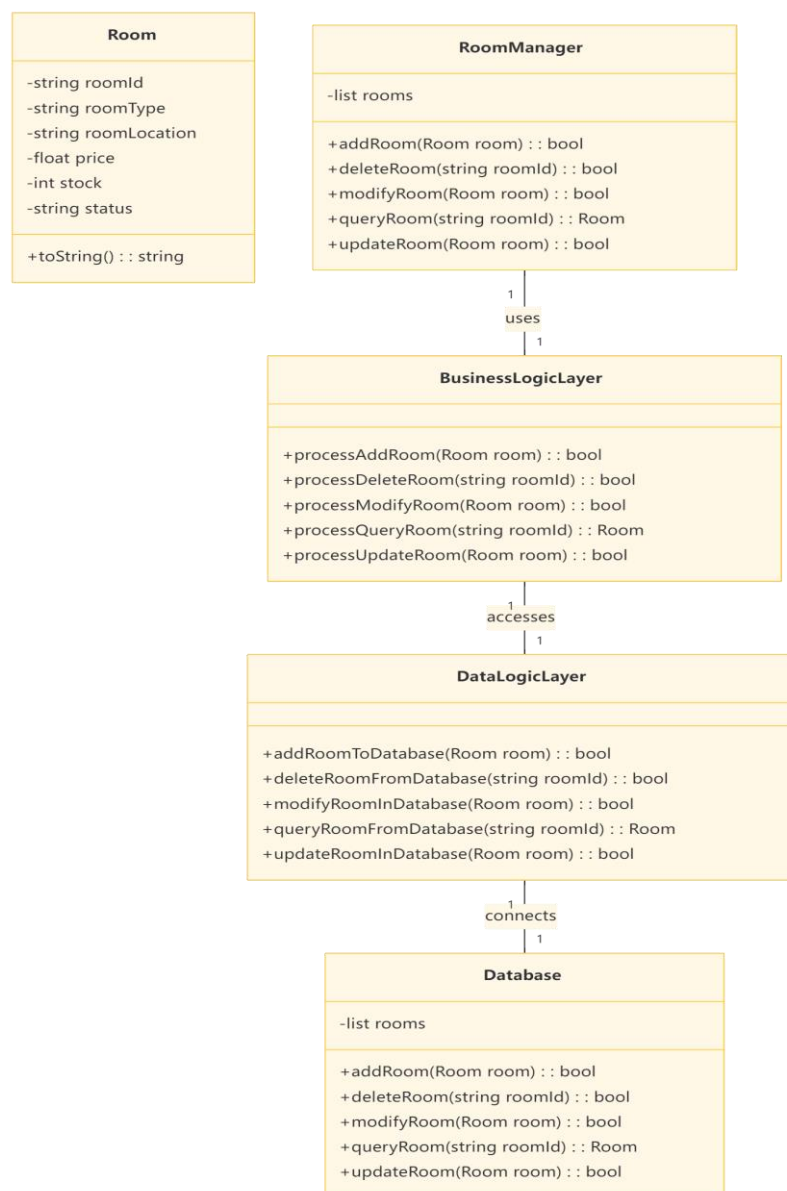


Рисунок 23 – Модуль управления гостевыми комнатами UML-диаграмма

Класс Staff: представляет персонал, содержит атрибуты имени пользователя и пароля; предоставляет методы входа и регистрации.

Класс CheckInInfo: представляет информацию о регистрации, включая номер комнаты, время регистрации, имя пользователя и другую дополнительную информацию.

Класс интерфейса: предоставляет методы для отображения формы регистрации и информации о регистрации.

Бэкэнд-класс: обрабатывает запросы на регистрацию и обновляет информацию о регистрации в базе данных.

Класс базы данных: предоставляет методы для обновления о регистрации

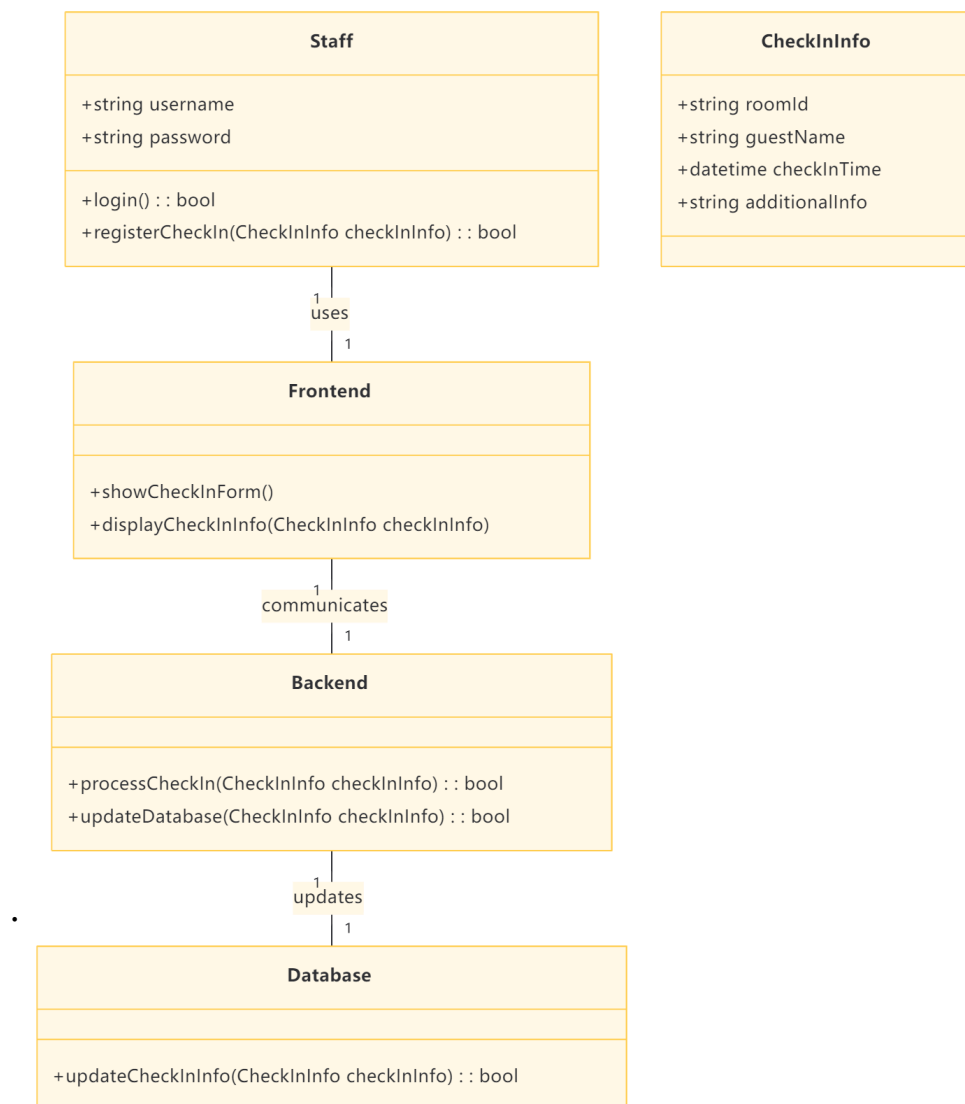


Рисунок 24 – UML-диаграмма управления регистрацией

Класс Staff: представляет персонал, содержит атрибуты имени пользователя и пароля; предоставляет методы для входа в систему и выхода из нее.

Класс CheckOutInfo: представляет информацию о выезде, включая номер комнаты, имя пользователя, время выезда, сумму платежа, способ оплаты и т. д.

Класс интерфейса: предоставляет методы для отображения формы регистрации при оформлении заказа и отображения информации о оформлении заказа.

Бэкэнд-класс: обрабатывает запросы на регистрацию выписки и обновляет информацию о выписке в базе данных.

Класс базы данных: предоставляет методы для обновления информации о выписке.



Рисунок 25 – UML-диаграмма управления кассой

Дизайн данных полей таблицы данных оказывает большое влияние на работу системы, поэтому очень важно хорошо спроектировать поля таблицы данных. Посредством анализа функциональных требований к системе существует восемь основных информационных таблиц, которые должна использовать система управления онлайн-бронированием отеля, а именно таблица пользователя, таблица комментариев, таблица новостей, таблица сообщений, таблица общественного питания, таблица бронирования питания, таблица гостевой комнаты, таблица гостей. Стол в номере Форма бронирования, форма заезда, форма выезда, форма персонала.

В таблице меню в основном хранится основная информация о блюдах. В название типа приема пищи, изображение приема пищи, цена приема пищи, описание приема пищи, дата приема пищи, стиль приема пищи, особенности приема пищи и рекомендации.

Таблица номеров для гостей включает номер комнаты, название комнаты, номер типа, название типа, время добавления, статус, цену за единицу, конфигурацию, фотографии и клики. Стол в комнате для гостей показан в Таблице 6 ниже.

Таблица 6 – Стол в гостевой комнате

Имя поля	Значение	Тип поля	Длина или диапазон
1	2	3	4
ID	Номер комнаты	Char	4
Name	Название комнаты	Varchar	50
TypeID	Введите имя	Varchar	4
TypeName	Введите имя	Varchar	50
Content	Введите номер	Varchar	16
Addtime	Добавить время	DateTime	8
ZhuangTai	Состояние	Varchar	50
PeiZhi	Конфигурация	Text	16

Продолжение таблицы 6

Photo	Фото	Image	50
Click	Нажмите	Int	4
DanJia	цена за единицу товара	Float	8

Таблица пользователей включает номер пользователя, скидку, имя, пол, возраст, адрес электронной почты, адрес, номер телефона, имя пользователя, пароль и баллы. Сущность пользователей показана в таблице 7.

Таблица 7 – Таблица пользователей

Имя поля	значение	Тип поля	Длина или диапазон	Может ли он быть пустым
1	2	3	4	5
ID	ID пользователя	Char	4	основной ключ
Zhekou	Скидка	Float	8	Null
XingMing	Имя	Varchar	50	Null
Sex	пол	Varchar	50	Null
Age	возраст	Varchar	50	Null
Ds	описывать	Text	16	Null
JiFen	интеграл	Float	8	Null
UserName	имя	Varchar	50	Null
Pwd	пароль	Varchar	50	Null

В таблице меню в основном хранится основная информация о блюдах. В том числе номер приема пищи, название приема пищи, номер типа приема пищи, название типа приема пищи, изображение приема пищи, цена приема

пищи, описание приема пищи, дата приема пищи, стиль приема пищи, особенности приема пищи и рекомендации. Список приемов пищи показан в таблице 8.

Таблица 8 – Список приемов пищи

Имя поля	Значение	Тип поля	Длина или диапазон	Может ли он быть пустым
1	2	3	4	5
CanPinID	Номер еды	Char	4	основной ключ
CanPinName	Название блюда	Varchar	50	Null
CanPinTypeID	Номер типа питания	Char	4	Null
CanPinTypeName	Название типа питания	Varchar	50	Null
CanPinPhoto	Фотографии еды	Image	50	Null
CanPinIntroduce	Знакомство с едой	Text	16	Null
CanPinPrice	Цены	Float	8	Null
CanPinDate	Дата еды	DateTime	8	Null
CanPinKuanshi	Стиль питания	Varchar	50	Null
CanPinTeSe	Особенности питания	Varchar	50	Null
TuiJian	рекомендовать	Int	4	Null

Детальное проектирование модуля. Модуль входа: реализация проверки имени пользователя/пароля и проверки кода проверки (проверка на стороне JS и взаимодействие на стороне ASP.NET).

Подсистема пользователя. Модуль регистрации: пароль хранилища, зашифрованный MD5, проверка легитимности поля. Модуль бронирования: запрос статуса номера и логика формирования заказа.

Подсистема администратора. Модуль управления данными: реализация CRUD-операций с информацией о гостевых комнатах/питании (элементы управления ASP.NET привязаны к данным MySQL).

Технические детали реализации. Интерфейс: ASP.NET Web Forms создает интерактивный интерфейс, а JavaScript помогает в динамической проверке.

Бэкэнд: Уровень бизнес-логики написан на языке C#, а для подключения к базе данных MySQL используется ADO.NET.

Меры безопасности: Session управляет статусом пользователя и предотвращает параметризованные запросы с внедрением SQL.

Вывод: работоспособная система, основные фрагменты кода.

Фаза тестирования системы. Стратегия тестирования: тестирование методом черного ящика: охватывает все функциональные варианты использования (например, регистрацию пользователя, процесс бронирования номеров). Тестирование методом белого ящика: проверка логики кода (например, реализация алгоритма шифрования MD5).

Дизайн тестового случая пример использования. Вход тестового пользователя: если вы введете неверный пароль, система сообщит, что верификация не пройдена. Добавлено в тестовую комнату: сообщение о том, что добавление не удалось из-за ввода неверного поля. Новое дополнение к тестовым блюдам: сообщение о том, что добавление не удалось из-за ввода неверного поля.

3.2. Примеры фактического тестирования программного продукта

В программном обеспечении тестирование программного обеспечения является очень важным шагом. Во время работы написанного вами программного обеспечения вы также должны тестировать программное обеспечение, чтобы убедиться в его корректности и надежности. Поэтому отладка очень важна. Только путем многократных корректировок можно привести систему в идеальное состояние, позволяющее пользователям использовать программу в безопасной и защищенной среде. Поэтому отладка является наиболее важной частью.

Тестирование программного обеспечения позволяет обнаружить недостатки в проекте, а также является основным методом предотвращения рисков. Это очень важный шаг во всей системе. В наших предыдущих тестах, чем более детальной и совершенной была наша работа, тем более удовлетворительной она была. Поэтому чем критичнее программное обеспечение, тем тщательнее оно тестируется. Система управления заказами отелей онлайн – это система с большим количеством личной конфиденциальной информации. Как только база данных будет украдена, это неизбежно приведет к большим проблемам для пользователей. Поэтому ее тестирование – очень важный шаг.

Таблица 9 – Тестовые случаи входа в систему

Имя	Шаги теста	Ожидаемый результат
1	2	3
Войдите в систему как обычно	Введите правильное зарегистрированное имя пользователя и пароль и нажмите «Войти».	Переход на домашнюю страницу и отображение приветственного сообщения.
Ошибка имени пользователя	Введите незарегистрированное имя пользователя или неправильное имя пользователя и правильный пароль.	Отображается сообщение «Неверное имя пользователя или пароль».
Неправильный пароль	Введите правильное имя пользователя, неправильный пароль.	Сообщение «Неверное имя пользователя или пароль»
Пустой ввод	Имя пользователя или пароль пусты, нажмите «Войти».	Запрос «Пожалуйста, введите имя пользователя/пароль» и выделение пустых полей.
Чувствительность к регистру	Имя пользователя или пароль вводятся в смешанном регистре (например, User123 vs user123)	Регистр чувствителен, вход невозможен (спроектировано в соответствии с требованиями).

Таблица 10 – Тестовые случаи входа в систему

Имя	Шаги теста	Ожидаемый результат
1	2	3
Обычная регистрация	Введите требуемое имя пользователя, пароль, подтвердите пароль и нажмите «Зарегистрироваться»	Сообщение «Регистрация выполнена успешно», переход на страницу входа или на домашнюю страницу.
Имя пользователя уже существует	Введите зарегистрированное имя пользователя	Сообщение «Имя пользователя уже существует, измени того».
Недостаточная сложность пароля	Введите простой пароль (например, 123456)	Подсказка «Пароль должен содержать заглавные и строчные буквы, цифры и специальные символы и должен быть длиной не менее 8 символов».

Таблица 11 – Новые тестовые случаи для общественного питания

Имя	Шаги теста	Ожидаемые результаты
1	2	3
T3	Откройте интерфейс добавления ресторана, информацию о ресторане, чтобы завершить добавление ресторана.	Появится сообщение «Добавлено успешно»
T4	Откройте интерфейс добавления ресторана , введена неверная информация о ресторане.	Подсказка «Добавить не удалось»
T5	Откройте интерфейс добавления ресторана , введена неверная информация о ресторане.	Подсказка «Добавить не удалось»

酒店评价

评价信息

评价人: 李飞 时间: 2023/4/25

内容: sfdsd sdf

回复内容: 等待回复

评价人: 李飞 时间: 2023/4/25

内容: 斯蒂芬斯蒂芬的是

回复内容: 等待回复水电费第三方胜多负少的

共【1】页 当前第【1】页 首页 上一页 下一页 尾页

评价类型:

好评

评价

Рисунок 26 – Интерфейс добавления комментариев

Когда пользователь успешно входит в систему, он или она может просмотреть веб-страницу отеля на веб-странице, ввести сообщение и добавить сообщение; изменить комментарий, чтобы вернуться к интерфейсу, соответствующему комментарию. Интерфейс добавления комментариев показан на рисунке.

酒店在线预订管理系统						
菜品信息管理						
菜品名称	菜品类别	购买单价	菜品数量	发布日期	修改	删除
菜品1	小吃	25	98	2023/4/25	修改	删除
菜品2	小吃	25	98	2023/4/25	修改	删除
菜品3	小吃	25	100	2023/4/25	修改	删除
菜品4	小吃	25	100	2023/4/25	修改	删除
菜品5	小吃	25	100	2023/4/25	修改	删除
菜品6	小吃	25	100	2023/4/25	修改	删除
菜品7	小吃	25	98	2023/4/25	修改	删除
菜品8	小吃	25	100	2023/4/25	修改	删除
菜品9	小吃	25	99	2023/4/25	修改	删除
菜品10	小吃	25	100	2023/4/25	修改	删除

Рисунок 27 – Добавить интерфейс комнаты

Управление комнатами в основном включает в себя добавление, удаление и изменение комнат; запрос комнат и другие обновления комнат, такие как обновления комнат, требующие обновлений комнат в представлении и обратной связи; доступ к уровню бизнес-логики и доступ к базовому уровню логики данных для получить связи с базой данных.

酒店在线预订管理系统	预定入住	登记入住
预定入住		
预定的客房: 109客房 房屋单价: 168 入住天数: 2 入住日期: 2023-05-06 退房日期: 入住时间: 20:30 入住姓名: 姚春 联系电话: 15946213650		
预定入住		

Рисунок 28 – Интерфейс добавления регистрации

С помощью кнопки «Регистрация» сотрудники могут заполнить информацию о регистрации на главной странице, включая номер комнаты, время регистрации, имя пользователя и т. д. После заполнения информация о регистрации в базе данных будет обновиться, и отобразится соответствующая информация о регистрации.

酒店在线预订管理系统		菜品管理		菜品信息管理			
菜品名称	菜品类别	购买单价	菜品数量	发布日期	修改	删除	
菜品1	小吃	25	98	2023/4/25	修改	删除	
菜品2	小吃	25	98	2023/4/25	修改	删除	
菜品3	小吃	25	100	2023/4/25	修改	删除	
菜品4	小吃	25	100	2023/4/25	修改	删除	
菜品5	小吃	25	100	2023/4/25	修改	删除	
菜品6	小吃	25	100	2023/4/25	修改	删除	
菜品7	小吃	25	98	2023/4/25	修改	删除	
菜品8	小吃	25	100	2023/4/25	修改	删除	
菜品9	小吃	25	99	2023/4/25	修改	删除	
菜品10	小吃	25	100	2023/4/25	修改	删除	

Рисунок 29 – Интерфейс добавления еду

Менеджеры могут напрямую войти на веб-страницу публикации ресторана, нажав кнопку «Выпуск общественного питания», и загрузить соответствующую информацию о ресторане. Содержимое в основном включает в себя: название блюда, тип, цену; характеристики, характеристики, инвентарь и т. д., прежде чем отправить его в Ресторан После отправки ресторана таблица информации о ресторане в базе данных обновит данные, и система отобразит информацию о ресторане.

ЗАКЛЮЧЕНИЕ

Систему управления информацией о бронировании отелей онлайн можно разделить на три модуля, а именно, модуль администратора, модуль персонала и модуль пользователя.

Функции модуля администратора включают управление пользователями, управление сообщениями, управление новостными данными, управление регистрацией и выездом, управление товарами, управление размещением, управление номерами и управление меню.

Функции модуля управления персоналом включают управление регистрацией и выездом, управление номерами, управление размещением, управление меню и управление клиентами.

Функции модуля пользователя включают бронирование номеров, изменение данных, онлайн-сообщение, онлайн-комментарий, бронирование питания и регистрация.

В работе рассмотрены текущие состояния отечественных и зарубежных исследований, изучены технологии разработки и внедрения инженерии. Выполнен анализ осуществимости и анализ спроса системы, а также выполнено проектирование функциональных модулей, проектирование баз данных и детальное проектирование и реализация каждого модуля фронтальной и бэкэндов. Также описываются методы тестирования, используемые в этой системе, и перечисленные тестовые случаи. С точки зрения проектирования системы, данная система основана на структуре B/S, использует базу данных MySQL, а также технологию ASP.NET и язык C#.

В процессе разработки этой системы было выполнено: поиск соответствующей литературы и материалов, чтение соответствующей технической литературы для постепенного прояснения концепций и идей, анализ данных, улучшение навыков рисования и дизайна, проведение опросов пользователей, моделирование базы данных, изменение системных ошибок, отладка кода и

другие процессы обучения и исследования. Интерфейсом этой системы являются бизнес-операции пользователя, включая просмотр информации, обратную связь по сообщениям, бронирование номеров, бронирование столиков в ресторане, отзывы об отелях, проверки регистрации и т. д. Администраторы могут добавлять, удалять, проверять и изменять информацию в этой системе в фоновом режиме управления, например: управление сотрудниками, управление клиентами, управление новостями, управление номерами, управление размещением, управление питанием, управление статистикой и т. д.

Основные преимущества разработанной системы:

- общий интерфейс этой системы прост, с быстрыми запросами и хорошей практичностью;
- система практична и может удовлетворить как потребности развития отеля, так и потребности клиентов;
- система имеет хорошую надежность и высокий уровень безопасности;
- система проста в обслуживании и удобна в использовании.

Основные недостатки:

- в модуле новостей нельзя изменять картинки по желанию. можно только перевыбирать картинки или не вносить никаких изменений, что повлияет на скорость изменения данных сотрудниками бэкэнда;
- функции интерфейса не выделяются, внутренняя страница управления слишком проста, а общий дизайн пользовательского интерфейса системы недостаточно красив.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

- 1 Агафонова, А. А. Роль цифровизации в повышении качества сервиса в российской гостиничной индустрии / А. А. Агафонова // Журнал прикладных исследований. – 2024. – №9. – С. 89-125.
- 2 Аркабаев, Н. К. Разработка Web-серверных приложений на базе .NET CORE в примере интернет-магазина [Электронный ресурс] / Н. К. Аркабаев, З. Ж. Алымова // ВОГУ. – 2024. – №1. – Режим доступа : <https://cyberleninka.ru/article/n/razrabotka-web-servernyh-prilozheniy-na-baze-net-core-v-primere-internet-magazina>. – 15.05.2025.
- 3 Баранова, Н. В. Якименко М. В. Трансформационные процессы в отечественном гостиничном бизнесе / Н. В. Баранова, М. В. Якименко // Вестник науки. – 2024. – №6. – С. 75.
- 4 Володченко, В. С. Разработка web-сайта [Электронный ресурс] / В. С. Володченко, Д. С. Ланцова, Т. А. Миронова, Е.В. Сапунова, К. А. Бышок // Достижения науки и образования. – 2019. – №12 – С. 53. – Режим доступа : <https://cyberleninka.ru/article/n/razrabotka-web-sayta>. – 15.05.2025.
- 5 Давыдова, М. А. Web-технологии в бизнесе [Электронный ресурс] / М. А. Давыдова // E-Scio. – 2018. – №6. – С. 21. – Режим доступа : <https://cyberleninka.ru/article/n/web-tehnologii-v-biznese>. – 15.05.2025.
- 6 Допира, Р. И. Технология разработки интернет-магазина [Электронный ресурс] / Р. И. Допира, Н. В. Попова, К. М. Базикова // Научный журнал. – 2016. – №1. – С. 2. – Режим доступа : <https://cyberleninka.ru/article/n/tehnologiya-razrabotki-internet-magazina>. – 11.03.2025.
- 7 Журавлев, А. А. Разработка приложения для просмотра кулинарных рецептов с помощью ASP.NET Core [Электронный ресурс] / А. А. Журавлев // StudNet. – 2021. – №4. – Режим доступа :

<https://cyberleninka.ru/article/n/razrabotka-prilozheniya-dlya-prosmotra-kulinarny-h-retseptov-s-pomoschyu-asp-net-core>. – 15.05.2025.

8 Журавлев, А. А. Создание социальной сети с помощью среды разработки Visual Studio и платформы ASP.NET Core MVC [Электронный ресурс] / А. А. Журавлев, А. В. Немтинов // StudNet. – 2020. – №12. – Режим доступа : <https://cyberleninka.ru/article/n/sozдание-sotsialnoy-seti-s-pomoschyu-sredy-razrabotki-visual-studio-i-platformy-asp-net-core-mvc>. – 15.05.2025.

9 Журавлев, Н. Е. Взаимодействие web-сервера и web-приложения через Web-Socket [Электронный ресурс] / Н. Е. Журавлев // Мировая наука. – 2019. – №12. – С. 33. – Режим доступа : <https://cyberleninka.ru/article/n/vzaimodeystvie-web-servera-i-web-prilozhenie-cherez-web-socket>. – 15.05.2025.

10 Елуферьева, Ю. С. Корпоративные web-серверы [Электронный ресурс] / Ю. С. Елуферьева // Экономика и социум. – 2018. – №7. – С. 50. – Режим доступа : <https://cyberleninka.ru/article/n/korporativnye-web-servera>. – 15.05.2025.

11 Кульков, Д. А. Создание web-сервера [Электронный ресурс] / Д. А. Кульков, Е. Ю. Дедова, С. Г. Бедняк // Экономика и социум. – 2018. – №5. – С. 48. – Режим доступа : <https://cyberleninka.ru/article/n/sozдание-web-servera>. – 15.05.2025.

12 Лаврова, А. П. Основные тенденции развития в сфере гостиничного бизнеса / А. П. Лаврова // Гуманитарные, социально-экономические и общественные науки. – 2025. – №3. – С. 55-90.

13 Марьясова, Е. А. Эволюция гостиничной индустрии: тенденции и перспективы на будущее / Е. А. Марьясова, Д. О. Кирсанова, Д. Н. Кузнецова, А. Р. Рыжова, А. В. Шишнякова // Инновационная экономика: информация, аналитика, прогнозы. – 2025. – №2. – С. 23-32.

14 Мурадова, Г. Б. Основы web-программирования [Электронный ресурс] / Г. Б. Мурадова, Л. С. Минич // Academy. – 2021. – №2. – С. 65. – Режим

доступа : <https://cyberleninka.ru/article/n/osnovy-web-programmirovaniya>. – 15.05.2025.

15 Негодин, В. А. Использование платформы ASP.NET при разработке приложения для моделирования аварийного разлива нефти [Электронный ресурс] / В. А. Негодин // Форум молодых ученых. – 2019. – №8. – С. 36. – Режим доступа : <https://cyberleninka.ru/article/n/ispolzovanie-platformy-asp-net-pri-razrabotke-prilozheniya-dlya-modelirovaniya-avariynogo-razliva-nefti>. – 15.05.2025.

16 Нелюбов, Р. П. Безопасность в веб-приложениях ASP.NET Core [Электронный ресурс] / Р. П. Нелюбов // E-Scio. – 2022. – №5. – С. 68. – Режим доступа : <https://cyberleninka.ru/article/n/bezopasnost-v-veb-prilozheniyah-asp-net-core>. – 15.05.2025.

17 Нуриев, Г. Ю. Web-разработка [Электронный ресурс] / Г.Ю. Нуриев, А.П. Багаева // Актуальные проблемы авиации и космонавтики. – 2022. – Режим доступа : <https://cyberleninka.ru/article/n/web-razrabotka>. – 15.05.2025.

18 Першукова, С. А. Реинжиниринг бизнес-процессов предприятий гостиничного и ресторанного бизнеса на основе конкурентных / С. А. Першукова // Гостиничный бизнес. – 2025. – №5. – С. 14-15.

19 Пищенко, Д. А. Особенности организации и функционирования ресторанного бизнеса при отеле / Д. А. Пищенко // Вестник науки. – 2025. – №1. – С. 82.

20 Родионов, В. В. Разработка web-приложения для оптимизации организационных задач на языке программирования C# ASP.NET Core [Электронный ресурс] / В. В. Родионов, Е. Ю. Воронкин // Интерэкспо Гео-Сибирь. – 2021. – №2. – Режим доступа : <https://cyberleninka.ru/article/n/razrabotka-web-prilozheniya-dlya-optimizatsii-organizatsionnyh-zadach-na-yazyke-programmirovaniya-c-asp-netcore>. – 15.05.2025.

21 Селяков, М. А. Трехуровневая архитектура микросервиса в ASP.NET [Электронный ресурс] / М. А. Селяков // Вестник науки. – 2020. – №4. – С. 25.

– Режим доступа : <https://cyberleninka.ru/article/n/trehurovnevaya-arhitektura-mikroservisa-v-asp-net>. – 15.05.2025.

22 Смирнова, А. С. Информационные технологии в гостиничном бизнесе [Электронный ресурс] / А. С. Смирнова // Вестник ПГУ им. Шолом-Алейхема. – 2021. – №4. – С. 45. – Режим доступа : <https://cyberleninka.ru/article/n/informatsionnye-tehnologii-v-gostinichnom-biznese>. – 15.05.2025.

23 Спасов, Д. Ю. Веб-приложение система онлайн-бронирования для гостиничного бизнеса [Электронный ресурс] / Д. Ю. Спасов // Вестник науки. – 2024. – №6. – С. 75. – Режим доступа : <https://cyberleninka.ru/article/n/veb-prilozhenie-sistema-onlayn-bronirovaniya-dlya-gostinichnogo-biznesa>. – 15.05.2025.

24 Степанова, В. М. Автоматизация хозяйственно преимуществ, гибкости и инновации в условиях социально-экономических изменений / В. М. Степанова // Инновационная экономика: информация, аналитика, прогнозы. – 2025. – №1. – С. 44-46.

25 Таран, В. Н. Кроссплатформенная разработка web-сайтов [Электронный ресурс] / В. Н. Таран, Р. С. Горшар, В. Е. Осыка // Технические науки – от теории к практике. – 2016. – №3. С. 51. – Режим доступа : <https://cyberleninka.ru/article/n/krossplatformennaya-razrabotka-web-saytov>. – 15.05.2025.

26 Федотов, В. А. Разработка интернет-магазина [Электронный ресурс] / В. А. Федотов // Форум молодых ученых. – 2020. – №8. – С. 48. – Режим доступа : <https://cyberleninka.ru/article/n/razrabotka-internet-magazina-1>. – 15.05.2025.

27 Шарапов, Н. Р. Архитектура технологии разработки веб-приложений Asp.Net Core MVC [Электронный ресурс] / Н. Р. Шарапов // Вопросы науки и образования. – 2018. – №13. – С. 25. – Режим доступа : <https://cyberleninka.ru/article/n/arhitektura-tehnologii-razrabotki-veb-prilozheniy-asp-net-core-mvc>. – 15.05.2025.

28 Шарапов, Н. Р. Эффективность применения технологии Asp.Net Core для разработки веб-приложений [Электронный ресурс] / Н. Р. Шарапов // Вопросы науки и образования. – 2018. – №13. – С.25. – Режим доступа : <https://cyberleninka.ru/article/n/effektivnost-primeneniya-tehnologii-asp-net-core-dlya-razrabotki-veb-prilozheniy>. – 15.05.2025.