

Министерство образования, науки и молодежной политики Нижегородской области
Государственное автономное профессиональное образовательное учреждение
«Перевозский строительный колледж»

Специальность: 09.02.07 Информационные системы и программирование
ПМ 01. Разработка программных модулей программного обеспечения для компьютерных систем

Форма обучения: очная

Группа: 4121

Номер зачетной книжки _____

Работа защищена с оценкой

« _____ »
(прописью)

_____ Грищенко А.А.
(подпись) (ФИО руководителя курсовой работы)

« _____ » _____ 2021 г.

КУРСОВАЯ РАБОТА

на тему Разработка программного модуля информационной системы гостиничного бизнеса

Исполнитель

(подпись)

_____ Купцова Ю.С.
(инициалы, фамилия)

« _____ » _____ 20__ г.

Руководитель

(подпись)

_____ Грищенко А.А.
(инициалы, фамилия)

« _____ » _____ 20__ г.

г. Перевоз,

2021 г.

ЗАДАНИЕ НА КУРСОВУЮ РАБОТУ

по ПМ 01. Разработка программных модулей программного обеспечения для компьютерных систем

студенту 4 курса, группы 4121

Купцовой Юлии Сергеевне
(фамилия, имя, отчество)

1. Тема Разработка программного модуля информационной системы гостиничного бизнеса

Утверждена приказом от « ___ » _____ 202_ г. № _____

2. Структура, содержание, объем курсовой работы

Введение

Глава 1. Теоретическое обоснование темы работы

Глава 2. Техническое задание

2.1. Техническое задание

Глава 3. Практическая часть

3.1 Технические средства разработки

3.2. Структура (модель) программы

3.3 Структура программного кода

3.4 Описание интерфейса пользователя

Заключение

Список используемых источников

Дата выдачи задания: « ___ » _____ 20__ г.

Срок окончания работы « ___ » _____ 20__ г.

Руководитель курсовой работы _____ А.А. Грищенко
(подпись) ФИО

Зав. ПЦК Информационных технологий _____ М.Н. Кокина
(подпись) ФИО

Содержание

Введение.....	4
Глава 1. Теоретическое обоснование темы работы.....	6
1.1 История становления и развития гостиничной индустрии	6
1.2 Организационная структура гостиниц и прочих средств размещения	9
1.3 Базовые службы гостиниц.....	11
Глава 2. Техническое задание	17
Глава 3. Практическая часть	19
3.1 Технические средства разработки.....	19
3.2. Описание предметной области	21
3.3 Структура (модель) программы.....	27
3.4 Описание интерфейса пользователя	30
Заключение	37
Список используемых источников.....	38
Приложение	40

Введение

С развитием общества росла потребность в широком обмене информацией. Работа человека с информацией при использовании компьютерных технологий предъявляет новые требования к обществу, и в связи с этим развивается информатизация.

Процесс бронирования начинается с сотрудничества клиента и отеля. Заявка поступает через Интернет, телефон, почту и другие средства лично от клиента или посредника - физического или юридического лица. Каждое бронирование должно быть зарегистрировано в отделе бронирования. В заявлении клиент должен указать фамилию, имя, отчество, дату прибытия, продолжительность пребывания и дату отъезда, категорию и количество номеров, форму оплаты.

Так что создание и существование такого рода программы многим упрощает работу. Тем более в нашем современном мире, мире технологий, это крайне необходимо.

Актуальность темы курсовой работы заключается в том, что на сегодняшний день любая гостиница нуждается в программном обеспечении, позволяющим автоматизировать процессы бронирования гостиничных номеров.

Целью данного курсовой работы является разработка программного модуля информационной системы гостиничного бизнеса, выполняющая автоматизацию процессов бронирования гостиничных номеров.

Объектом исследования является автоматизация процессов бронирования гостиничных номеров.

Предметом исследования – разработка собственной информационной системы.

Для достижения поставленной цели необходимо решение конкретных **задач:**

- Исследовать предметную область гостиничного комплекса;

- Осуществить постановку задачи;
- Построить функциональную модель предоставления номеров в гостинице;
- Спроектировать базу данных гостиницы;
- Разработать программный код;
- Протестировать работу разработанной системы;
- Ввод программного обеспечения в эксплуатацию.

ОБРАЗЕЦ

Глава 1. Теоретическое обоснование темы работы

1.1 История становления и развития гостиничной индустрии

На сегодняшний день гостиничный сервис - это наиболее масштабный и быстро развивающийся пласт экономики всех развитых стран.

Согласно словарю русского языка С. И. Ожегова, гостиница - дом для временного проживания приезжающих с одноместными или неодноместными номерами, с обслуживанием.

Первое упоминание о гостиницах можно встретить в древнегреческих манускриптах.

Гостиницы в те времена носили название таверн или постоялых дворов. Из-за того, что путешествия по стране занимали огромное время, пребывание в пути было крайне долгим, организация ночлега и питания для путешественников становилась все более и более актуальной задачей правительства тех времен.[1]

Особенности классификаций различных стран

Все гостиничные комплексы и прочие средства размещения делятся на категории в зависимости от комфортабельности и разнообразия предоставляемых услуг. Категория средства размещения обозначается звездами, что является наиболее распространенной классификацией:

- 5 звезд — гостиница класса люкс;
- 4 звезды — гостиница первого класса;
- 3 звезды — гостиница улучшенной средней категории;
- 2 звезды — гостиница средней категории;
- 1 звезда — гостиница туристской категории.

В свою очередь туристские гостиницы, насчитывающие более пяти номеров, делятся на шесть категорий от самой низкой (без звезды) до самой высокой (4 звезды luxury). Крупные гостиничные комплексы, состоящие из ста

и более номеров, имеют категории от одной до четырех звезд.

Требования, предъявляемые к гостиницам и туристским комплексам:

- наличие минимум пяти номеров;
- минимальная площадь холла (зависит от количества номеров и общей площади здания);
- наличие входа в гостиницу, независимого от входа в ресторан, наличие служебного входа (зависит от звездности гостиницы), наличие лифта (в многоэтажных зданиях), системы отопления и кондиционирования;
- оснащенность гостиницы телефонами, Интернетом, горячим и холодным водоснабжением;
- требования, предъявляемые к жилой зоне: обеспечение шумоизоляции, минимальная площадь жилых помещений (номеров), наличие кухонного уголка (характерно для гостиниц с одной или двумя звездами), наличие собственного санузла, площадь ванных комнат, уровень освещенности жилых помещений и ванных комнат и пр.;
- требования к персоналу: гостиницы с двумя звездами — знание одного иностранного языка; гостиницы с тремя звездами и выше — знание двух иностранных языков, включая английский;
- требования к системе питания: гостиницы с тремя звездами — возможность заказа еды в номера-апартаменты; гостиницы с четырьмя и более звездами — обязательное наличие ресторана на территории отеля;
- уровень доступности гостиницы для лиц с ограниченными возможностями.[3]

Классификация гостиниц

Пять звезд — крупные гостиницы или гостиничные комплексы, соответствующие международным стандартам размещения, с высоким уровнем обслуживания, питания и организацией досуга.

Четыре звезды — гостиницы с номерами площадью не менее 40 м², с высоким уровнем сервиса и питания. Обязательно наличие: многокомнатных

апартаментов, ванной и душевой кабины, обслуживания номеров и возможности заказать еду в номер.

Три звезды — отели среднего размера, круглосуточная работа персонала и всех подразделений, ресторан, бар в здании отеля. Индивидуальные ванны комнаты в каждом номере.

Две звезды — наличие индивидуальных ванны комнат минимум, чем в 50% номеров, наличие телефона и телевизора в каждом номере.

Одна звезда — маленький отель с минимальным набором необходимых услуг и хорошим (домашним) питанием. Не менее 25% номеров с индивидуальными ванными комнатами. Управляющими гостиниц чаще всего являются их же хозяева, проживающие на одном из этажей.[4]

Члены автомобильной ассоциации утверждают, что число звезд не является гарантом качества средств размещения, а просто указывает на размеры гостиницы и уровень оснащённости. Для более подробной оценки Автомобильная ассоциация использует *критерий процентное*, который показывает разницу в качестве обслуживания гостиниц с одинаковым количеством звезд. Процентная оценка является сугубо субъективной и ставится проверяющим анонимным инспектором. Оцениваются качество сервиса, профессионализм персонала, его приветливость, учтивость, вежливость, дружелюбие, опрятность, качество предметов интерьера в номере, чистота зон питания, общее впечатление от гостиницы.

Оценку в 50-59% получают гостиницы, уровень обслуживания в которых удовлетворяет минимальным запросам, которые обеспечивают небольшой, но достаточный набор услуг и удобств.

Оценка 60-69% — отели с более высоким классом обслуживания в некоторых областях.

Оценка 70—89% — гостиницы с хорошим сервисом, высоким уровнем обслуживания, питания и гостеприимства.

Гостиницы, получившие оценку выше 80%, отмечаются звездами красного цвета, вместо обычных желтых. [2]

Помимо классификации средств размещения на основе качества обслуживания и уровня комфорта, существует классификация гостиниц по их функциональному предназначению. Различают:

- деловые гостиницы — предназначены для лиц, путешествующих по роду профессиональной деятельности (семинары, конгрессы, выставки, специальные курсы и пр.). К таким отелям относятся: бизнес-отели, конгресс-отели, конгресс-центры и ведомственные отели;
- туристические гостиницы — основной целью таких гостиниц является размещение туристов, приезжающих отдохнуть или получить профилактическое и восстановительное лечение. К таким гостиницам относятся: курортные отели, пансионаты, дома отдыха, санатории, профилактории, казино-отели;
- транзитные гостиницы — необходимы для кратковременного обслуживания проезжающих мимо туристов. Это привокзальные гостиницы, гостиницы при аэропортах, придорожные мотели;
- гостиницы для постоянного проживания. Предоставляются клиентам, находящимся в длительных командировках (от 1,5 месяца);
- гостиницы различной формы собственности. К ним относятся: кондоминиумы — гостиничные комплексы, определенное количество номеров, проданных владельцам, которые перепродают или сдают в аренду; таймшерные отели — в данном случае продается не сам отель, а право на долгосрочное в нем проживание. [3]

1.2 Организационная структура гостиниц и прочих средств размещения

Структурная организация любой гостиницы зависит от ее функционального назначения, географического расположения, специфики приезжающих постояльцев (туристы, семьи с детьми, бизнесмены и пр.) и других факторов.

Кроме этого, именно организационная структура отражает обязанности и

полномочия каждого из работников сферы гостиничного сервиса.

В условиях современного рынка структура управления любым объектом должна быть конкурентоспособной. Для этого она должна быть как можно более простой и гибкой (рис. 1). [2]

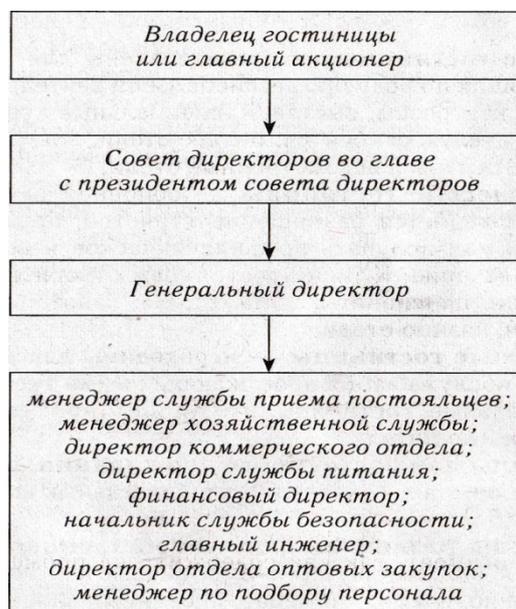


Рис. 1. Примерный план организационной структуры гостиничного предприятия

Служба размещения постояльцев, служба бронирования номеров и служба консьержа подчиняется менеджеру службы приема гостей.

Горничные и уборщица подчиняются менеджеру по хозяйственной части.

Менеджеры по продажам и служба организации банкетов в подчинении у директора коммерческого отдела.

Шеф-повара, повара, бармены и главный бармен, а также директор ресторана и официанты подчиняются директору службы питания. В некоторых гостиницах служба по организации банкетов также подчиняется директору службы питания.

Отдел кадров и отдел обучения персонала находятся в подчинении у менеджера по работе с персоналом.

Иногда отдел закупок может входить в состав финансового отдела и подчиняться его директору.

Во многих странах есть свои особенности структур управления гостиничным бизнесом. Так, например, в Европе во главе большинства подразделений стоят старшие сотрудники: старший носильщик, старшая горничная, старший сотрудник службы бронирования, старший телефонист, старший сотрудник информационной службы и т. д.

Чем меньше отель, тем проще он организован, однако основные структуры остаются неизменными. Чем масштабнее гостиничное предприятие, тем большую структурную организацию оно имеет, так как содержит ряд промежуточных ступеней, основная функция которых — контроль за исполнением поручений.

Весь менеджментный состав имеет четкую иерархию (рис.2), и все сотрудники занимают определенное положение, согласно своему статусу. В самом низу иерархической структуры находятся отдельные сотрудники — работники и исполнители — обслуживающий персонал гостиницы. [3]

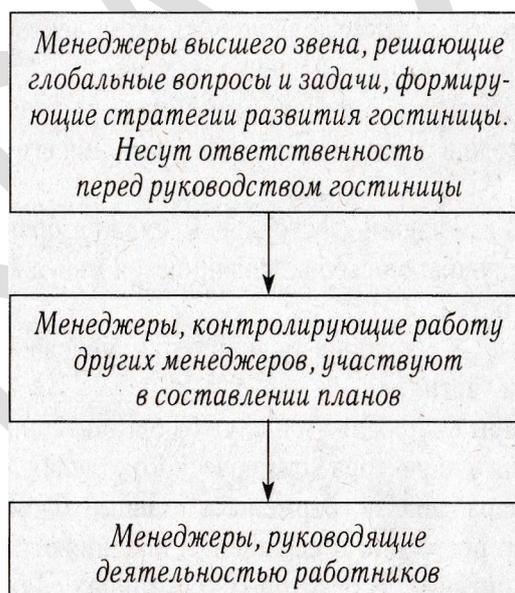


Рис 2. Иерархическая система менеджмента среднестатистического отеля

1.3 Базовые службы гостиниц

К основным структурным подразделениям любой гостиницы относятся:

- служба контроля за номерным фондом;

- административная служба;
- служба питания;
- финансовая служба;
- технические службы;
- дополнительные службы.

Служба контроля за номерным фондом. Основными задачами данной структуры являются: реализация бронирования номеров, прием прибывающих гостей, регистрация и размещение гостей в соответствующие номера, обеспечение. Трансфера после окончания пребывания в гостинице, трансфер к следующему пункту маршрута, обслуживание гостей в номерах (удовлетворение всех потребностей), поддержание санитарно-гигиенического состояния номера на должном уровне, обеспечение комфорта в жилых зонах, а также оказание любых услуг бытового характера (стирка, химчистка, глажение, чистка обуви и пр.).

Во главе данной службы стоит менеджер по приему и размещению гостей, или менеджер по эксплуатации номеров. В состав данной структуры входят: служба приема и размещения; клининговая служба (горничные); сервисная служба (гардеробщики, коридорные и пр.); служба консьержей; служба посыльных; служба портье; инспекция по уборке номеров и служба безопасности отеля. [3]

Потенциальный клиент может бронировать (резервировать) номер различными способами: по телефону, факсу, электронной почте, телексу или с помощью компьютерного терминала.

Как только запрос поступает в службу бронирования номеров отеля, в программе или журнале сразу фиксируются следующие персональные данные:

- фамилия, имя, отчество клиента;
- адрес проживания и контактный телефон;
- сколько ожидается гостей;
- предполагаемая дата въезда / выезда;
- характеристики номера (тип номера, тип кровати);

- гарантия оплаты и кто оплачивает предоставляемые услуги (клиент или фирма);
- дополнительные пожелания (вид из окна, наличие в номере цветов, гипоаллергенные средства личной гигиены и пр.). [5]

Административная служба

Административная служба — это управленческая структура, в обязанность которой входит контроль за всеми подразделениями отеля, решение финансовых проблем, кадровых вопросов. Кроме того административная служба обеспечивает поддержание условий труда, контролирует соблюдение норм и стандартов режима труда и отдыха персонала, отвечает за соблюдение техники безопасности, а также противопожарной и экологической безопасности. [3]

Служба питания и напитков

Это подразделение гостиничного сервиса находится на втором месте по прибыльности, после номерного фонда. Наибольшую прибыль приносят банкеты и корпоративные обеды, проводимые на территории гостиницы. В большинстве гостиниц есть несколько точек по обеспечению постояльцев питанием и напитками: ресторан (иногда несколько), кафе, бары.

Помимо непосредственно обеспечения постояльцев пищей, служба питания отвечает за проведение банкетов и корпоративов.

Состав службы:

- кухня во главе с шеф-поваром;
- бары во главе с главным барменом;
- кафе;
- служба банкетинга;
- ресторан.

Задачи руководителя подразделения общественного питания:

- составление меню ресторанов и кафе вместе с шеф-поваром;

- распределение обслуживающего персонала по участкам работы;
- контроль за качеством пищи и напитков;
- контроль за уровнем сервиса.

Обслуживанием в номерах занимается отдельное подразделение службы.

Его сотрудники обеспечивают:

- прием заказов от постояльцев;
- регистрацию заказов с указанием времени желаемой доставки и дополнительных пожеланий;
- передачу заказов на кухню;
- контроль за приготовлением пищи;
- доставку готового продукта постояльцу через службу консьержей. [1]

Отдел маркетинга и продаж **Задачи отдела:**

- проведение маркетинговых исследований (анализ ситуации на рынке услуг, анализ уровня спроса и предложений, анализ ценовой политики, установление тарифа и систем скидок с целью привлечения клиентов и увеличения доходов);
- поиск оптовых покупателей гостиничных услуг (туристические агентства, фирмы-организаторы конгрессов и конференций, фирмы, приглашающие много командированных и пр.);
- поддержание контакта с уже знакомыми клиентами, а также поиск новой клиентуры.

Кроме этого служба маркетинга и продаж занимается непосредственно продажей гостиничного продукта, организацией различных массовых предприятий с целью привлечения

дополнительной клиентуры, а также рекламой предоставляемых услуг с целью их продвижения.

Непосредственной задачей службы является организация масштабных рекламных кампаний и составление рекомендаций о наиболее эффективных направлениях капиталовложений.

Отдел маркетинга занимается связями с общественностью — как за пределами предприятия (отеля), так и в его стенах. Иными словами, регулярно проводит анализ степени удовлетворенности постояльцев оказываемыми услугами.

С этой целью гостей отеля просят заполнить анкеты, в которых предлагается оценить по пятибалльной шкале такие показатели, как:

- качество уборки номеров;
- оснащенность номеров как техническими, так и индивидуально-гигиеническими средствами;
- приветливость персонала;
- качество питания и пр.

По результатам анализа анкет сотрудники отдела маркетинга формулируют предложения по улучшению качества обслуживания и направляют их руководству гостиницы. [2]

Инженерная служба

Данная служба является одной из самых главных, хоть и наиболее незаметной в структуре гостиничного сервиса. Благодаря ей, поддерживаются условия правильного функционирования всех систем жизнеобеспечения клиентов и сотрудников (теплоснабжение, кондиционирование, санитарно-техническое оборудование, электроснабжение помещений, службы ремонта и строительства и пр.). В состав этого подразделения входят службы ремонта и строительства, службы по обслуживанию телевизионного, кабельного и компьютерного оборудования, службы по обеспечению работы лифтов, службы, контролирующей работу кухонного оборудования (газовых и электроплит), служба по благоустройству территории, служба связи, служба по контролю за расходом электроэнергии, воды, газа и прочих ресурсов. Во главе службы стоит главный инженер. [3]

Вывод к 1 главе:

В первой главе курсовой работы были описаны подробные теоретические данные о гостиничном комплексе, его значимость. Были даны классификации гостиниц. Описана организационная структура гостиничного комплекса. Были описаны базовые службы гостиничного комплекса, их разновидности, а также дана подробная информация каждой из них.

ОБРАЗЕЦ

Глава 2. Техническое задание

2.1. Техническое задание

1. Введение

Предметом разработки является написание программы гостиничного комплекса. Программное обеспечение должно быть достаточным для выполнения всех реализованных функций системы, а также иметь средства для организации всех необходимых процессов обработки данных, позволяющих своевременно выполнять все функции во всех режимах работы системы.

2. Основание для разработки

Наименование работы: «Разработка программного модуля информационной системы гостиничного бизнеса»

Исполнитель: Купцова Юлия Сергеевна

Соисполнители: отсутствуют

3. Назначение разработки

Программа предназначена для автоматизации работ по заказу номеров и услуг, заполнению анкет, отслеживанию свободных и зарезервированных номеров, вводу в базу данных новых данных о клиентах.

4. Технические требования

4.1 Требования к функциональным характеристикам

Программа должна выполнять следующие функции:

- Регистрация нового клиента;
- Поиск клиентов в базе данных;
- Редактирование информации о клиентах;
- Бронирование гостиничных номеров;
- Выявление в базе данных занятых и свободных номеров.

4.2 Требования к надежности

Надежность системы в целом зависит от надежности используемой операционной системы.

4.3 Требования к составу и параметрам технических средств

Приложение должно работать на персональных компьютерах.

Минимальные системные требования:

- Тип процессора: Intel Celeron
- Объем оперативной памяти: 1024 Мб
- Объем свободного места на жестком диске: 100 Мб

4.4 Требования к программной совместимости

Программа работает на персональных компьютерах под управлением операционных систем семейства Windows (7, 8, 10).

5. Требования к программной документации

Разрабатываемые программные модули должны быть самодокументированы, т.е. тексты программ должны содержать все необходимые комментарии.

Вывод к главе 2:

В данной главе представлено техническое задание на разработку программы гостиничного комплекса. Описано назначение. Технические характеристики, требования к разрабатываемому программному продукту, требования к совместимости, а также перечислена программная документация, которая будет разработана в процессе создания программы.

Глава 3. Практическая часть

3.1 Технические средства разработки

Перед разработкой программного модуля информационной системы гостиничного бизнеса нужно было сделать выбор среди инструментальных средств для разработки. Мною были выбраны следующие средства:

- Среда разработки Visual Studio 2019;
- Язык программирования C#;
- СУБД MS SQL Management Studio 2017.

C# (произносится как "си шарп") — простой, современный объектно-ориентированный и типобезопасный язык программирования. C# относится к широко известному семейству языков C, и покажется хорошо знакомым любому, кто работал с C, C++, Java или JavaScript.

C# является объектно-ориентированным языком, но поддерживает также и компонентно-ориентированное программирование. Разработка современных приложений все больше тяготеет к созданию программных компонентов в форме автономных и самоописательных пакетов, реализующих отдельные функциональные возможности. Важная особенность таких компонентов — это модель программирования на основе свойств, методов и событий. Каждый компонент имеет атрибуты, предоставляющие декларативные сведения о компоненте, а также встроенные элементы документации. C# предоставляет языковые конструкции, непосредственно поддерживающие такую концепцию работы. Благодаря этому C# отлично подходит для создания и применения программных компонентов.

Вот лишь несколько функций языка C#, обеспечивающих надежность и устойчивость приложений: сборка мусора автоматически освобождает память, занятую уничтоженными и неиспользуемыми объектами; обработка исключений предоставляет структурированный и расширяемый способ выявлять и обрабатывать ошибки; строгая типизация языка не позволяет

обращаться к неинициализированным переменным, выходить за пределы индексированных массивов или выполнять неконтролируемое приведение типов.

В C# существует единая система типов. Все типы C#, включая типы-примитивы, такие как `int` и `double`, наследуют от одного корневого типа `object`. Таким образом, все типы используют общий набор операций, и значения любого типа можно хранить, передавать и обрабатывать схожим образом. Кроме того, C# поддерживает пользовательские ссылочные типы и типы значений, позволяя как динамически выделять память для объектов, так и хранить упрощенные структуры в стеке.

Чтобы обеспечить совместимость программ и библиотек C# при дальнейшем развитии, при разработке C# много внимания было уделено управлению версиями. Многие языки программирования обходят вниманием этот вопрос, и в результате программы на этих языках ломаются чаще, чем хотелось бы, при выходе новых версий зависимых библиотек. Вопросы управления версиями существенно повлияли на такие аспекты разработки C#, как отдельные модификаторы `virtual` и `override`, правила разрешения перегрузки методов и поддержка явного объявления членов интерфейса.

База данных SQL — это реляционная база данных общего назначения (управляемая служба) в Microsoft Azure, которая поддерживает такие структуры, как реляционные данные, JSON, пространственные данные и XML. База данных SQL предоставляет возможности динамического масштабирования производительности в двух разных моделях приобретения: на основе виртуальных ядер и DTU. База данных SQL также предоставляет такие компоненты, как индексы `columnstore` для экстремального аналитического анализа и отчетности и выполняющуюся в памяти OLTP для экстремальной обработки транзакций. Майкрософт быстро и эффективно обрабатывает исправления и обновления базы кода SQL, а также упрощает обслуживание базовой инфраструктуры.

База данных SQL Azure предоставляет следующие возможности

развертывания для Базы данных SQL Azure:

- Отдельная база данных с собственным набором ресурсов, управляемая с помощью сервера Базы данных SQL. Отдельная база данных аналогична автономным базам данных на SQL Server.
- Эластичный пул — это коллекция баз данных с общим набором ресурсов, управляемая с помощью сервера Базы данных SQL. Отдельные базы данных можно перемещать в эластичный пул и обратно.
- Управляемый экземпляр, который представляет коллекцию системных и пользовательских баз данных с общим набором ресурсов. Управляемый экземпляр аналогичен экземпляру ядра СУБД Microsoft SQL Server.

3.2. Структура (модель) программы

При проектировании разрабатываемой системы предполагается использование языка UML — унифицированного графического языка моделирования для описания, визуализации, проектирования и документирования объектно-ориентированных (ОО) систем. UML призван поддерживать процесс моделирования программных средств на основе ОО подхода, организовывать взаимосвязь концептуальных и программных понятий, отражать проблемы масштабирования сложных систем.

Мной были разработаны следующие диаграммы: диаграмма прецедентов, диаграмма последовательности, диаграмма компонентов, информационно-логическая модель базы данных.

Диаграмма прецедентов - применяется для моделирования вида системы с точки зрения прецедентов (или вариантов использования). Диаграммы прецедентов имеют большое значение для визуализации, специфицирования и документирования поведения элемента. Диаграммы прецедентов важны для тестирования исполняемых систем в процессе проектирования.

Диаграмма последовательности - такая диаграмма, на которой

показаны взаимодействия объектов, упорядоченные по времени их проявления.

Диаграмма компонентов - статическая структурная диаграмма, показывает разбиение программной системы на структурные компоненты и связи (зависимости) между компонентами. В качестве физических компонент могут выступать файлы, библиотеки, модули, исполняемые файлы, пакеты и т. п.

3.2.1 Диаграмма прецедентов

Диаграмма прецедентов используется для представления функциональной модели информационной системы.

Прецеденты (варианты использования — Use Cases) — это подробные процедурные описания вариантов использования системы всеми заинтересованными лицами, а также внешними системами, т. е. всеми, кто (или что) может рассматриваться как актёры (actors) — действующие лица. По сути, это своего рода алгоритмы работы с системой с точки зрения внешнего мира.

Прецеденты являются основой функциональных требований к системе, позволяют описывать границы проектируемой системы, ее интерфейс, а затем выступают как основа для тестирования системы заказчиком с помощью приемочных тестов.

Диаграмма прецедентов программного модуля информационной системы гостиничного бизнеса представлена на рисунке 3:

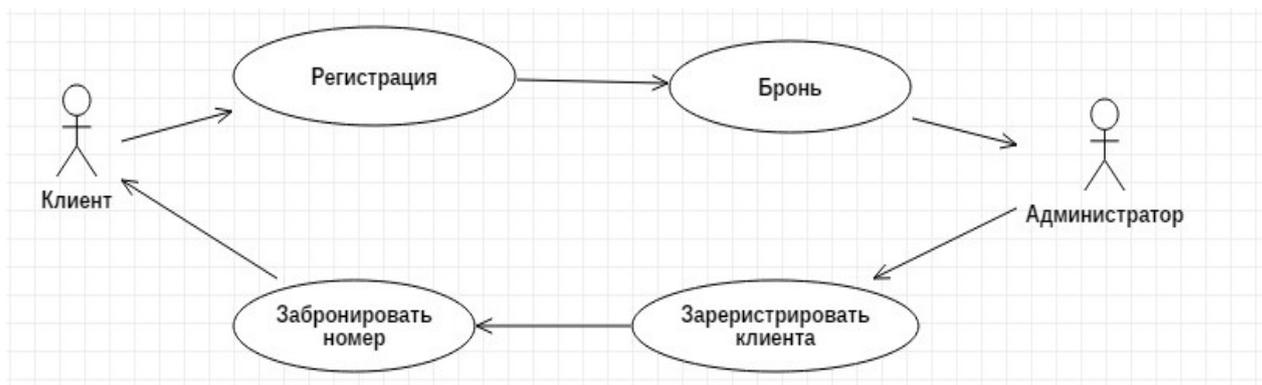


Рис.3 Диаграмма прецедентов

3.2.2 Диаграмма последовательности

Диаграмма последовательности — диаграмма, на которой показано взаимодействие объектов (обмен между ними сигналами и сообщениями), упорядоченное по времени, с отражением продолжительности обработки и последовательности их проявления.

Спроектированная мной диаграмма последовательности системы гостиничного комплекса представлена на рисунке 4:

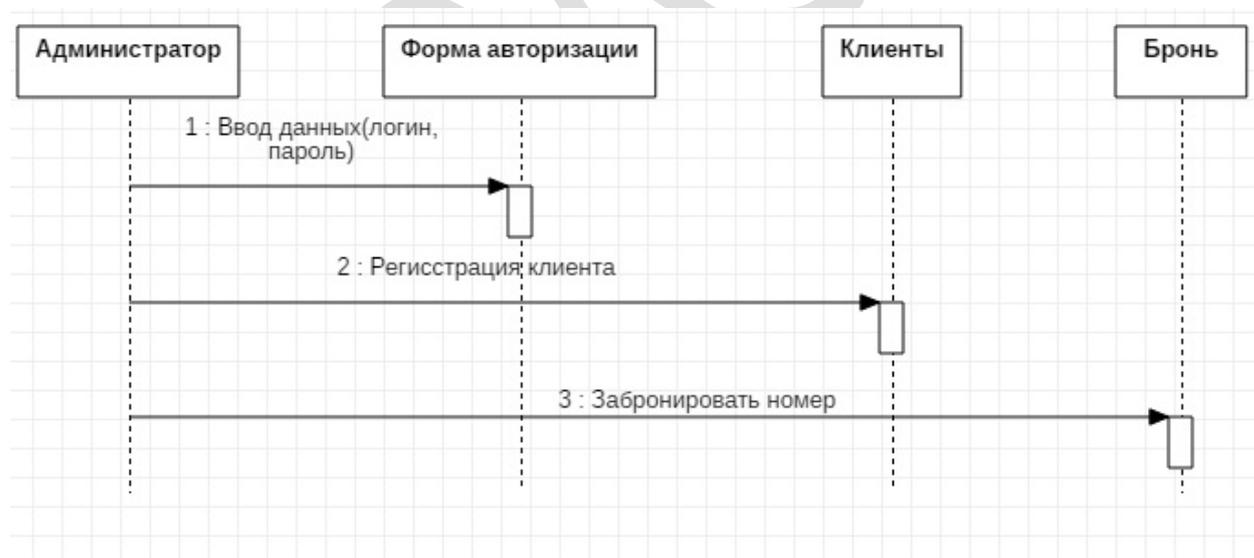


Рис.4 Диаграмма последовательности

3.2.3 Диаграмма компонентов

Диаграмма компонентов (Component diagram) — статическая структурная диаграмма, показывает разбиение программной системы на структурные

компоненты и связи (зависимости) между компонентами. В качестве физических компонент могут выступать файлы, библиотеки, модули, исполняемые файлы, пакеты и т. п.

Диаграммы компонентов применяются для моделирования статического вида системы с точки зрения реализации. Сюда относится моделирование физических сущностей, развернутых в узле, например исполняемых программ, библиотек, таблиц, файлов и документов. По существу, диаграммы компонентов - это не что иное, как диаграммы классов, сфокусированные на системных компонентах.

Ниже, на рисунке 5 представлена диаграмма компонентов:

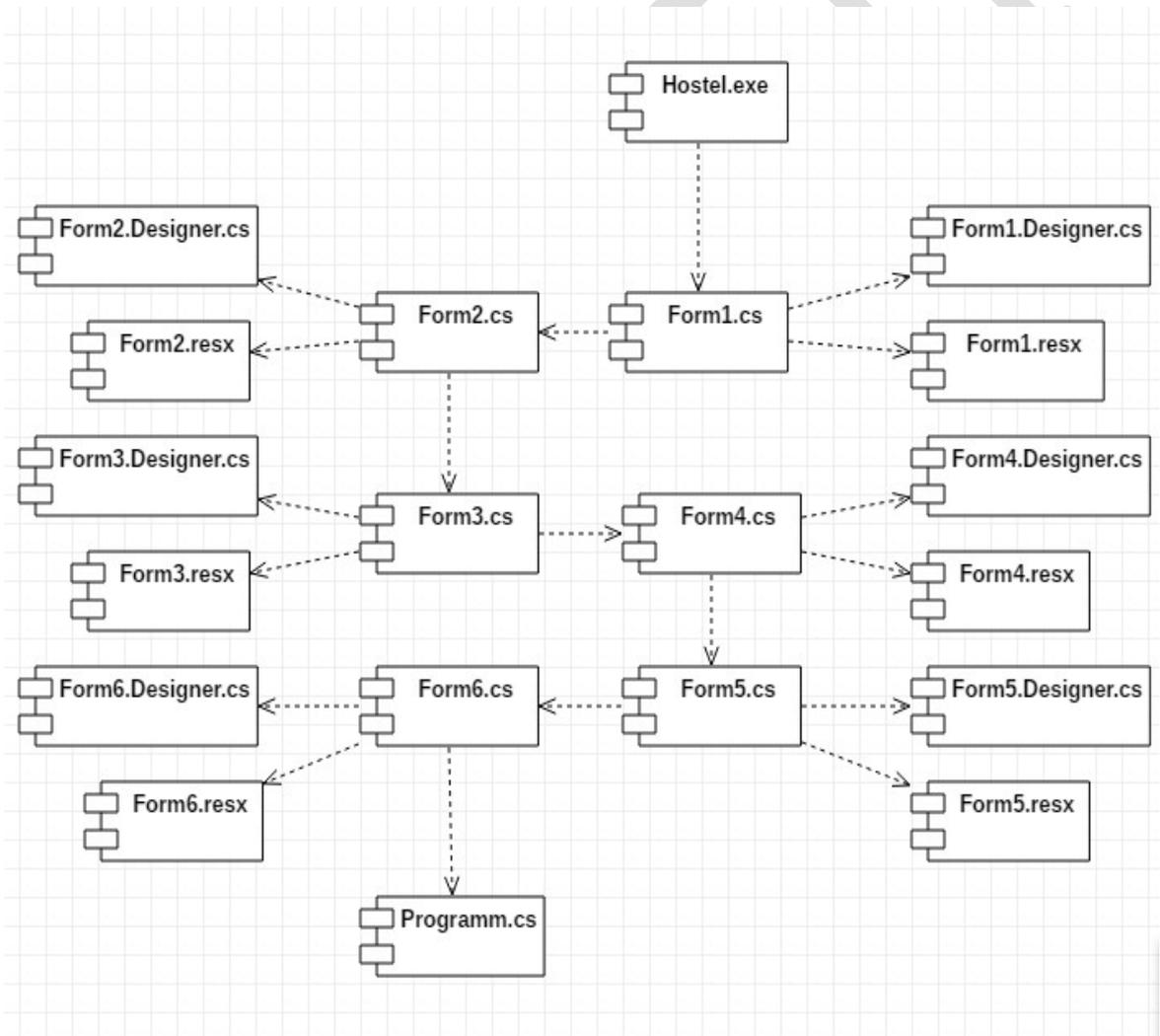


Рис.5 Диаграмма компонентов

3.2.4 Информационно-логическая модель базы данных

Информационно-логическая модель отображает данные предметной области в виде совокупности информационных объектов и связей между ними. Эта модель представляет данные, подлежащие хранению в базе данных.

Информационный объект - это информационное описание некоторого реального объекта, процесса, явления или события. Информационный объект образуется совокупностью взаимосвязанных реквизитов, представляющих качественные и количественные характеристики предметной области. Примерами информационных объектов могут быть Сотрудник, Ученик, Учитель, Методическая комиссия и т. п. Каждому информационному объекту нужно присвоить уникальное имя, соответствующее этому объекту.

Информационно-логическая модель разрабатываемой базы данных представлена на рисунке 6:

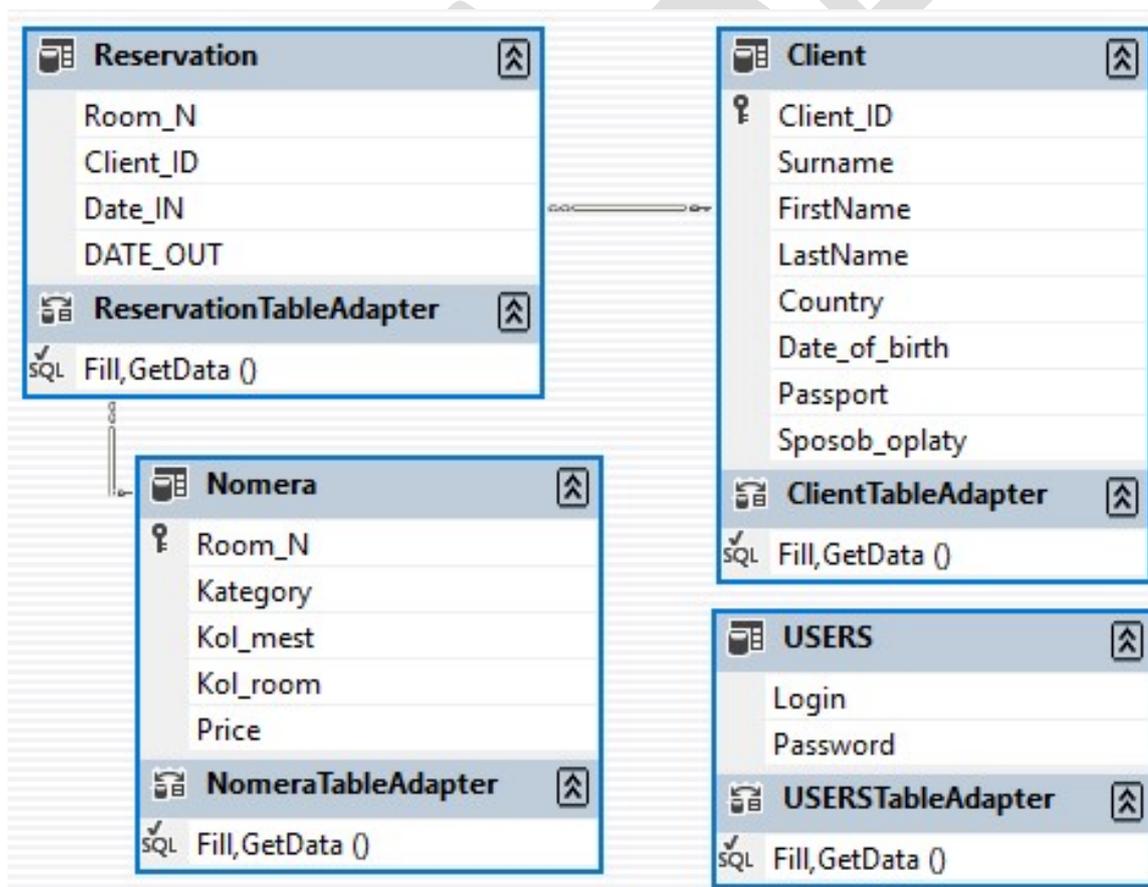


Рис.6 Информационно-логическая модель базы данных

3.2.5 Описание структуры базы данных

Назначение таблиц в разрабатываемой структуре данных следующее:

Таблица Users хранит логины и пароли сотрудников гостиничного комплекса, и служит для проверки правильности введенных данных при авторизации.

Таблица Clients содержит все необходимую информацию о клиентах (ФИО, страна, дата рождения, паспортные данные, способ оплаты)

Таблица Nomera содержит всю информацию о номерах (номер, категория, количество мест, количество комнат, цена).

Таблица Reservation служит для отображения свободных и забронированных номеров. Запись в нее осуществляется при бронировании номера.

Ниже представлен SQL запрос на создание базы данных:

```
CREATE table Client
(Client_ID INT NOT NULL Primary KEY Identity(1,1),
[Surname] NVARCHAR(30) NOT NULL,
[FirstName] NVARCHAR(30) NOT NULL,
[LastName] NVARCHAR(30) NOT NULL,
[Country] NVARCHAR(30) NOT NULL,
Date_of_birth DATE NOT NULL,
Passport NVARCHAR(11) NOT NULL,
Sposob_oplaty NVARCHAR(10) NOT NULL
)
```

```
INSERT INTO Client VALUES
('Чернов', 'Дмитрий', 'Сергеевич', 'РФ', '1995.02.14', '2212
443211', 'наличные')
```

```
CREATE TABLE Nomera
(Room_N NVARCHAR(3) NOT NULL PRIMARY KEY,
Kategory NVARCHAR(20) NOT NULL,
Kol_mest NVARCHAR(2) NOT NULL,
Kol_room NVARCHAR(2) NOT NULL,
Price MONEY NOT NULL)
```

```
INSERT INTO Nomera VALUES
('202', 'стандарт', '1', '1', '2000'),
```

```
(N'203',N'полулюкс',N'2',N'2','7000'),  
(N'301',N'люкс',N'2',N'1','9000'),  
(N'302',N'люкс',N'4',N'2','25000'),  
(N'303',N'стандарт',N'2',N'1','4000')
```

```
CREATE TABLE Reservation  
(Room_N NVARCHAR(3) NOT NULL,  
Client_ID INT NOT NULL,  
Date_IN DATE NOT NULL,  
DATE_OUT DATE NOT NULL,  
CONSTRAINT K1 FOREIGN KEY (Client_ID) REFERENCES Client  
(Client_ID) ON DELETE CASCADE ON UPDATE CASCADE,  
CONSTRAINT K2 FOREIGN KEY (Room_N) REFERENCES Nomera  
(Room_N) ON DELETE CASCADE ON UPDATE CASCADE  
)
```

```
INSERT INTO Reservation VALUES  
(N'202', '3', '2021.02.09', '2021.02.11'),  
(N'301', '3', '2021.02.13', '2021.02.22')
```

```
CREATE TABLE USERS(  
[Login] NVARCHAR(20) NOT NULL,  
[Password] NVARCHAR(20) NOT NULL)
```

```
INSERT INTO USERS VALUES (N'admin',N'admin')
```

3.3 Структура программного кода

3.3.1 DB.cs

Класс DB.cs содержит в себе методы для работы с базой данных и позволяет сократить код остальных форм, путем вызова методов из данного класса.

Класс DB.cs включает следующие методы:

public void OpenConnection() – открывает подключение к базе данных,
public void CloseConnection() – закрывает подключение с базой данных,
public SqlConnection GetConnection() – возвращает строку подключения к

базе данных

public bool ExistInDB(string s) – вот «Истина» если искомое значение

найденно в базе данных

`public string ReturnFromDB(string s)` – возвращает искомое значение в базе данных в виде строки

`public void NonQueryDB(string s)` – выполняет следующие запросы к базе данных: INSERT, UPDATE, DELETE.

Полный код данного класса и классов других форм представлен в приложении данной пояснительной записки.

3.3.2 Avtorization.cs

Данный класс относится к начальной форме, которая отвечает за авторизацию в системе работников гостиничного комплекса.

Форма содержит следующие методы:

`private void btn_vhod_Click(object sender, EventArgs e)` – обработчик события нажатия на кнопку «Вход», при данном событии происходит обращение к БД и в случае верно введенных данных пользователя, происходит открытие главного меню.

`private void txt_log_KeyDown(object sender, KeyEventArgs e)` – обработчик события нажатия клавиши Enter в поле для ввода логина. Вызывает те же методы, что и клавиша «Вход».

`private void Avtorization_FormClosing(object sender, FormClosingEventArgs e)` – обработчик события закрытия формы.

3.3.3 Main_menu.cs

Класс `Main_menu.cs` содержит обработчики событий нажатия на кнопки: «Клиенты», «Номера», «Забронировано», «Забронировать» - которые открывают соответствующие формы.

Так же на форме `Main_menu` присутствует метод который проверяет номера на занятость в выбранную дату, и окрашивающий таблички номеров в зеленый цвет если они свободны, и в красный – если заняты:

`private void CheckNum(TextBox text)`

```

{
    DB db = new DB();
    db.OpenConnection();
    SqlCommand com = new SqlCommand("SELECT Room_N FROM
Reservation WHERE Date_IN <= " + date1.Value.ToString("yyyy.MM.dd") + "
AND DATE_OUT>" + date1.Value.ToString("yyyy.MM.dd")+""",
db.GetConnection());

    SqlDataReader reader = com.ExecuteReader();
    string[] data = new string[9] { "", "", "", "", "", "", "", "", "" };
    int c = 0;
    while (reader.Read())
    {
        data[c] = reader[0].ToString();
        c++;
    }
    int count = 0;
    for (int i=0; i < 9; i++)
    {
        if (text.Text == data[i]) count++;
    }
    if (count>0) text.BackColor = Color.Red;
    else text.BackColor = Color.Green;
}

```

3.3.4 Reservation.cs

Данная форма позволяет выбрать данные клиента и номера, даты заселения и выселения, а также забронировать номер на указанные даты.

Форма содержит следующие методы:

private void date2_ValueChanged(object sender, EventArgs e) – обработчик события изменения дат в календарях, при изменении которых подсчитывается

количество дней проживания в номере

`private void cbx_nomer_SelectedIndexChanged(object sender, EventArgs e)` – обработчик события выбора номера из списка. При данном событии рассчитывается стоимость проживания в выбранном номере в указанные строки

`private void txt_bron_Click(object sender, EventArgs e)` – обработчик события нажатия на кнопку забронировать. При нажатии на кнопку происходит запись в таблицу `Reservation` и номер тем самым бронируется выбранным пользователем.

`private void cbx_client_SelectedIndexChanged(object sender, EventArgs e)` – обработчик события выбора клиента из списка. После выбора клиента, становится доступной функции выбора дат и номера.

3.4 Описание интерфейса пользователя

После запуска приложения открывается форма авторизации. Администратор системы должен заполнить поля формы авторизации, затем нажать войти. Система осуществляет обращение к базе данных и проверку введенных учетных данных, если данные не корректны выходит сообщение об этом, если же данные введены верно, то происходит открытие формы «Главное меню»:

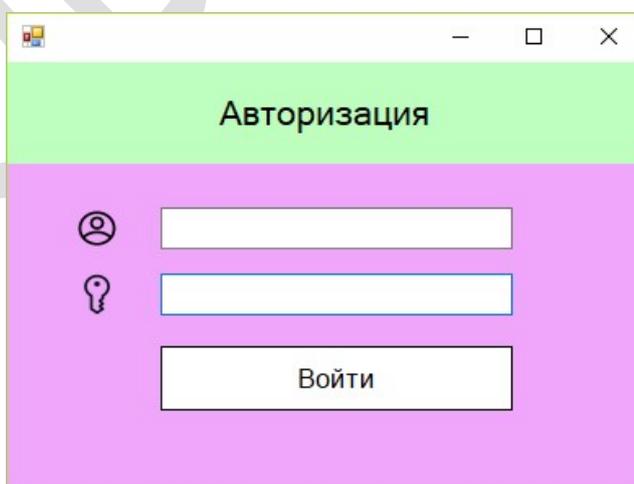


Рис.7 Авторизация администратора

На форме «Главное меню» присутствуют кнопки: «Клиенты», «Номера»,

«Забронировано», «Забронировать», открывающие соответствующие формы. Так же имеются таблички номеров гостиницы и календарь. Изменяя дату календаря можно узнать о свободных и занятых номерах на эту дату. Свободные номера подсвечиваются зеленым цветом, занятые – красным:



Рис.8 Окно главной формы «Гостиница»

При нажатии кнопки «Клиенты», открывается форма «Список клиентов», на которой имеется таблица с данными всех зарегистрированных клиентов, а так же имеются кнопки «Добавить» и «Назад»:

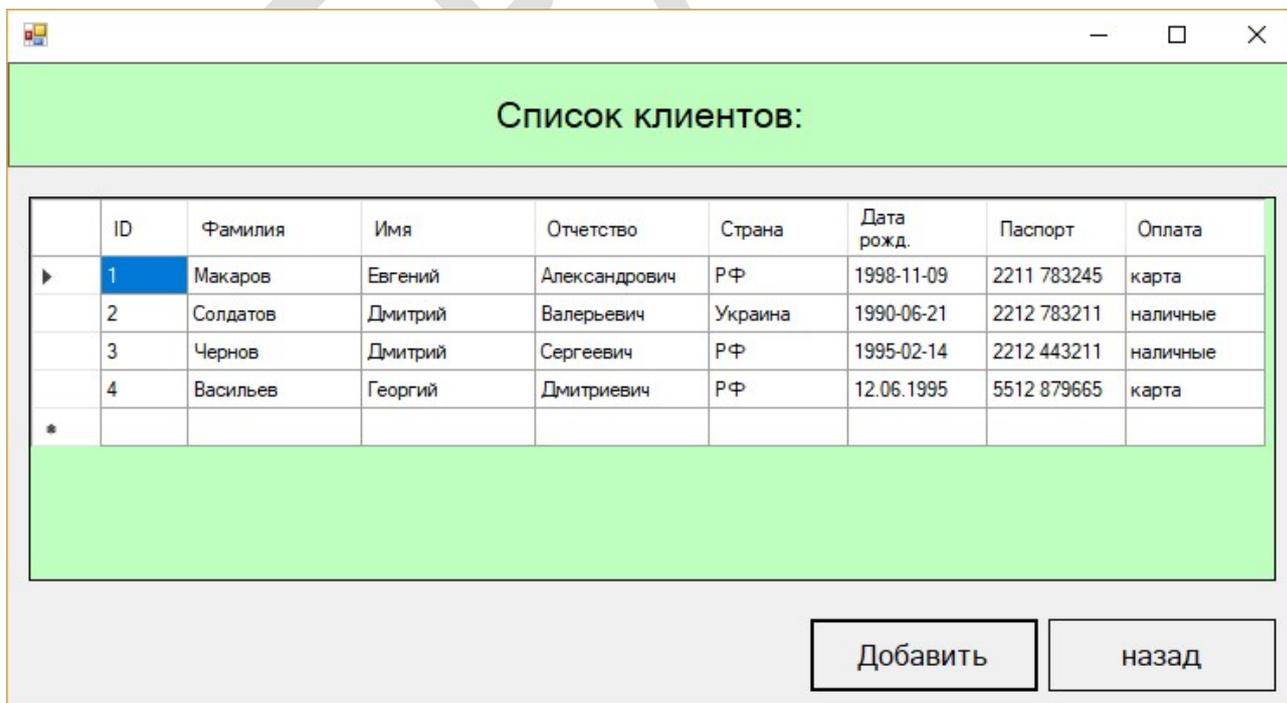


Рис.9 Окно главной формы «Гостиница»

При нажатии на кнопку «назад», произойдет переход к главному меню. При нажатии на кнопку «добавить», произойдет открытие формы «Добавление нового клиента». Данная форма содержит поля для заполнения личных данных клиента и кнопку «Регистрация». При нажатии на кнопку «Регистрация», происходит проверка заполненности всех полей, а так же проверка на существование такого пользователя в системе, и если все поля заполнены и такой пользователь ранее добавлен не был, то происходит запись данных клиента в базу данных. Добавим нового клиента:

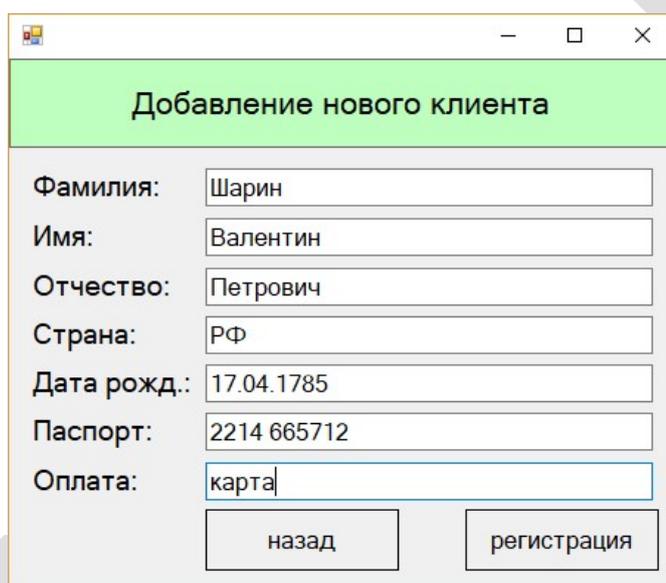


Рис.10 Заполнение данных на форме «Добавление нового клиента»

После добавления клиента, он отображается в списке клиентов:



ID	Фамилия	Имя	Отчество	Страна	Дата рожд.	Паспорт	Оплата
1	Макаров	Евгений	Александрович	РФ	1998-11-09	2211 783245	карта
2	Солдатов	Дмитрий	Валерьевич	Украина	1990-06-21	2212 783211	наличные
3	Чернов	Дмитрий	Сергеевич	РФ	1995-02-14	2212 443211	наличные
4	Васильев	Георгий	Дмитриевич	РФ	12.06.1995	5512 879665	карта
5	Шарин	Валентин	Петрович	РФ	17.04.1785	2214 665712	карта

Рис.11 Добавление нового клиента

Форма «Номера» содержит таблицу в которой отображены все

необходимые сведения о номерах гостиницы (номер, категория, количество мест, количество комнат и стоимость):

	№	Категория	Мест	Комнат	Стоимость
▶	101	люкс	3	2	9000
	102	полулюкс	2	1	5000
	103	стандарт	1	1	2000
	201	люкс	2	1	8000
	202	стандарт	1	1	2000
	203	полулюкс	2	2	7000
	301	люкс	2	1	9000
	302	люкс	4	2	25000

назад

Рис.12 Форма «Номера»

Форма «Забронированные номера» содержит таблицу с информацией о забронированных номерах на дату, выбранную в календаре, расположенном на данной форме:

	Номер	Фамилия	Имя	Отчество	Дата с:	Дата по:
▶	101	Макаров	Евгений	Александрович	13.02.2021 0:00:...	15.02.2021 0:00:...
	301	Чернов	Дмитрий	Сергеевич	13.02.2021 0:00:...	22.02.2021 0:00:...
	203	Васильев	Георгий	Дмитриевич	14.02.2021 0:00:...	26.02.2021 0:00:...
*						

14 февраля 2021 г.

назад

Рис.13 Форма «Номера»

Форма «Бронирование номера» служит для бронирования номеров

ГОСТИНИЧНОГО КОМПЛЕКСА:

№	Категория	Мест	Комнат	Стоимость
*				

Рис.14 Форма «Бронирование номера»

Изначально большинство полей данной формы скрыты. Администратор первым делом должен указать регистрировался ли данный клиент ранее в информационной системе гостиничного комплекса. Если нажать кнопку «нет» откроется форма добавления клиента, если нажать да – станет доступен выпадающий список с паспортными данными клиента. После выбора паспортных данных, станут также доступны календари, в которых имеется возможность указания дат заселения и выселения клиента. Указав даты, система рассчитывает количество дней проживания в гостинице, а так же отобразятся свободные номера на указанную дату:

№	Категория	Мест	Комнат	Стоимость
▶ 101	люкс	3	2	9000
102	полулюкс	2	1	5000
103	стандарт	1	1	2000
201	люкс	2	1	8000
203	полулюкс	2	2	7000
301	люкс	2	1	9000
302	люкс	4	2	25000
*				

Рис.15 Выбор клиента и сроков проживания

После прохождения описанных шагов, необходимо выбрать номер для

заселения. Когда номер выбран система рассчитывает стоимость проживания в данном номере в указанные сроки, а так же становится активна кнопка «Забронировать»:

The screenshot shows a web application window titled "Бронирование номера". On the left is a table with columns: №, Категория, Мест, Комнат, and Стоимость. The row for room 101 is highlighted. On the right is a form with the following fields:

- Client registration: "Клиент ранее регистрировался?" with "Да" and "Нет" buttons.
- Passport: "Паспорт клиента:" with a dropdown menu showing "2214 665712".
- Date selection: "Выбор даты" with "Дней:" set to "3", "С:" (From) set to "9 февраля 2021 г.", and "По:" (To) set to "12 февраля 2021 г.". There are calendar icons next to the dates.
- Room selection: "Выбор номера" with a dropdown menu showing "103" and "Стоимость:" set to "6000".
- Buttons: "Забронировать", "Назад", and "Сброс".

Рис.16 Заполнение всех полей формы бронирования номеров

После нажатия кнопки забронировать появляется сообщение об успешном бронировании номера:

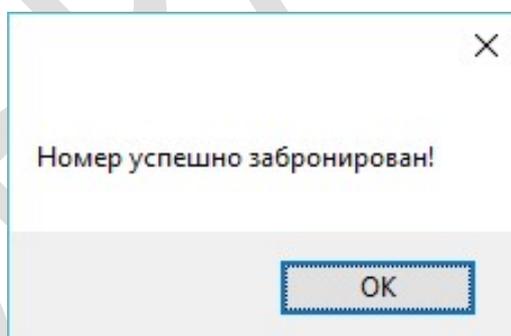


Рис.17 Сообщение о бронировании номера

Теперь, если выбрать на главном меню дату 11 февраля 2021 года, то забронированный ранее номер подсветиться красным:

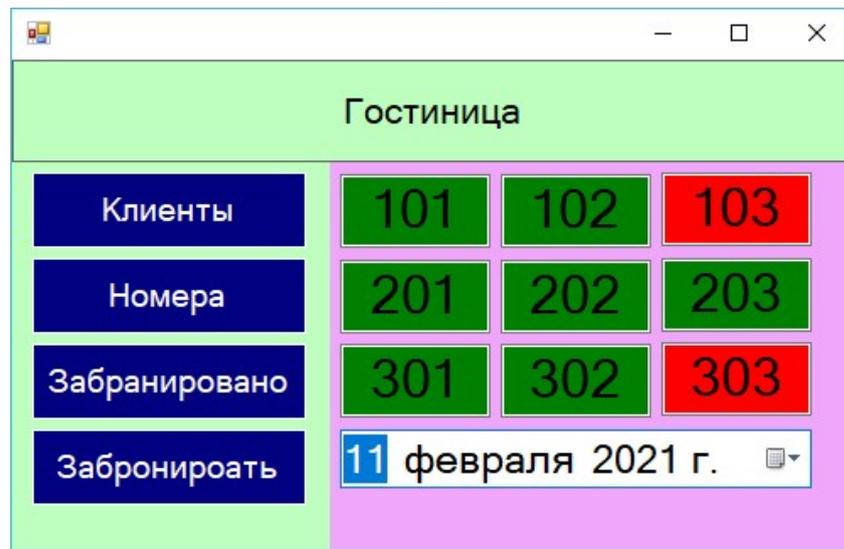


Рис.17 Отображение забронированных номеров

Вывод по главе 3:

Третья глава посвящена описанию среды программирования и разработки информационной системы и базы данных гостиничного комплекса. Описаны технические средства разработки; структура и модель создаваемого программного обеспечения; разработка интерфейса и описание работы программы.

Заключение

При выполнении курсовой работы был разработан программный модуль информационной системы гостиничного бизнеса, в целях ознакомления пользователей с историей гостиницы, а также гостиничного сервиса и использование информационных технологий в нем. Так же были выполнены задачи научной работы, а именно:

- Исследована предметная область гостиничного комплекса;
- Осуществлена постановка задачи;
- Построена функциональная модель предоставления номеров в гостинице;
- Спроектирована база данных гостиницы;
- Разработан программный код;
- Произведено тестирование работы разработанной системы;

Разработанная система гостиничного комплекса обеспечивает выполнение всего необходимого функционала и удовлетворяет требованиям, предъявляемым к ней перед началом разработки. Программное обеспечение выполняет следующие функции: регистрацию администратора гостиницы в системе, просмотр информации о номерах, клиентах, забронированных номерах на указанную дату, бронирование номера определенным клиентом на определенные даты, автоматический расчет стоимости проживания в указанном номере и другие.

На мой взгляд, мне удалось выполнить все задачи, поставленные передо мной в ходе выполнения курсовой работы, а так же овладеть всеми необходимыми компетенциями.

Список используемых источников

1. Албахари Джозеф , Албахари Бен С# 6.0. Карманный справочник; Вильямс - М., 2016. - 318 с
2. Албахари Джозеф , Албахари Бен С# 6.0. Справочник. Полное описание языка; Вильямс - М., 2016. - 623 с.
3. Аллен, Г. Тейлор SQL для чайников / Аллен Г. Тейлор. - М.: Диалектика, Вильямс, 2016. - 416 с.
4. Альфред В. Ахо, Моника С. Лам, Сети Рави , Джеффри Д. Ульман Компиляторы. Принципы, технологии и инструментарий; Вильямс - М., 2018. - 503 с
5. Баумгартен Л.В. Маркетинг Гостиничного предприятия. М. : Юрайт, 2016.
6. Бишоп Дж. С# в кратком изложении; Бином. Лаборатория знаний - М., 2017. - 234 с.
7. Гэри Маклин Холл Адаптивный код на С#. Проектирование классов и интерфейсов, шаблоны и принципы SOLID; Вильямс - М., 2015. - 432 с.
8. Дэвис Алекс Асинхронное программирование в С# 5.0; ДМК Пресс - М., 2016. - 120 с.
9. Дунаев, В. В. Базы данных. Язык SQL для студента / В.В. Дунаев. - М.: БХВ-Петербург, 2017. - 288 с.
10. Федцов В.Г. Культура сервиса. Учебное пособие / В.Г. Федцов. - М.: Приор, 2016. - 208 с.

Интернет ресурсы:

1. Функции службы Базы данных SQL - URL
<https://docs.microsoft.com/ru-ru/azure/sql-database/sql-database-technical-overview>
2. Проектирование информационной системы гостиницы - URL
<https://www.studsell.com/view/133290>

3. Основы языка SQL - URL

<http://citforum.ru/programming/32less/les44.shtml>

ОБРАЗЕЦ

Код программы:

DB.cs

```
using System;
using System.Collections.Generic;
using System.Data.SqlClient;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Hotel_Complex
{
    class DB
    {
        SqlConnection con = new SqlConnection("Data
Source=(localdb)\\mssqllocaldb;Initial Catalog=HotelDB;Integrated
Security=True;");
        public void OpenConnection()
        {
            if (con.State == System.Data.ConnectionState.Closed)
                con.Open();
        }
        public void CloseConnection()
        {
            if (con.State == System.Data.ConnectionState.Open)
                con.Close();
        }
        public SqlConnection GetConnection()
        {
            return con;
        }
        public bool ExistInDB(string s)
        {
            OpenConnection();
            SqlCommand com1 = new SqlCommand(s, GetConnection());
            SqlDataReader reader = com1.ExecuteReader();
            string target = "";
            while (reader.Read())
            {
                target = reader[0].ToString();
            }
            reader.Close();
            if (target != "") return true;
            else return false;
        }
    }
}
```

```

public string ReturnFromDB(string s)
{
    OpenConnection();
    SqlCommand com1 = new SqlCommand(s, GetConnection());
    SqlDataReader reader = com1.ExecuteReader();
    string target = "";
    while (reader.Read())
    {
        target = reader[0].ToString();
    }
    reader.Close();
    CloseConnection();
    return target;
}

public void NonQueryDB(string s)
{
    OpenConnection();
    SqlCommand com1 = new SqlCommand(s, GetConnection());
    com1.ExecuteNonQuery();
    CloseConnection();
}
}
}

```

Main_Menu.cs

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Hotel_Complex
{
    public partial class Main_menu : Form
    {
        public Main_menu()
        {
            InitializeComponent();
            Painting();
        }
    }
}

```

```

private void btn_clients_Click(object sender, EventArgs e)
{
    Clients cli = new Clients();
    this.Hide();
    cli.Show();
}

private void Main_menu_FormClosing(object sender,
FormClosingEventArgs e)
{
    Application.Exit();
}

private void btn_nomera_Click(object sender, EventArgs e)
{
    Nomera num = new Nomera();
    this.Hide();
    num.Show();
}

private void CheckNum(TextBox text)
{
    DB db = new DB();
    db.OpenConnection();

    SqlCommand com = new SqlCommand("SELECT Room_N FROM
Reservation WHERE Date_IN <= '" + date1.Value.ToString("yyyy.MM.dd") +
"' AND DATE_OUT>" + date1.Value.ToString("yyyy.MM.dd")+""",
db.GetConnection());
    SqlDataReader reader = com.ExecuteReader();
    string[] data = new string[9] { "", "", "", "", "", "", "", "", "" };
    int c = 0;
    while (reader.Read())
    {
        data[c] = reader[0].ToString();
        c++;
    }
    int count = 0;
    for (int i=0; i < 9; i++)
    {
        if (text.Text == data[i]) count++;
    }
    if (count>0) text.BackColor = Color.Red;
    else text.BackColor = Color.Green;
}

private void date1_ValueChanged(object sender, EventArgs e)
{

```

```

        Painting();
    }

    private void Painting()
    {
        n101.BackColor = Color.Green; n102.BackColor = Color.Green;
n103.BackColor = Color.Green; n201.BackColor = Color.Green;
        n202.BackColor = Color.Green; n203.BackColor = Color.Green;
n301.BackColor = Color.Green; n302.BackColor = Color.Green;
        n303.BackColor = Color.Green;
        CheckNum(n101);
        CheckNum(n102);
        CheckNum(n103);
        CheckNum(n201);
        CheckNum(n202);
        CheckNum(n203);
        CheckNum(n301);
        CheckNum(n302);
        CheckNum(n303);
    }

    private void btn_bronirovano_Click(object sender, EventArgs e)
    {
        Zabronirovano zab = new Zabronirovano();
        this.Hide();
        zab.Show();
    }

    private void btn_bron_Click(object sender, EventArgs e)
    {
        Reservation res = new Reservation();
        this.Hide();
        res.Show();
    }
}
}
}

```