

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних наук,
управління та адміністрування
Кафедра інформаційних технологій

Бакалаврська кваліфікаційна робота

на тему: Розробка інформаційної системи міні-готелю

Виконав студент 4 курсу групи К-41
Напрямок 6.050101 комп'ютерні науки
Алексеєвський Олександр Сергійович
Керівник старший викладач
Пономаренко Олена Леонідівна
Консультант к.геогр.н., доцент
Коваленко Людмила Борисівна
Рецензент к.ф.-м.н., доцент
Ткач Тетяна Борисівна

Одеса 2019

ЗМІСТ

Вступ.....	5
1 Характеристика предметної області. Постановка завдання	7
1.1 Використання інформаційних технологій в готельному бізнесі.....	7
1.2 Огляд інформаційної системи Fidelio	9
1.3 Система автоматизації готелів JSolution	10
1.4 Система управління та бронювання Lite PMS	11
1.5 Автоматизована система HOTEL 2000	15
1.6 Постановка завдання.....	18
2 Вибір та обґрунтування програмних засобів для реалізації інформаційної системи міні-готелю.....	20
2.1 Обґрунтування вибору архітектури	20
2.2 Вибір платформи .NET Framework	22
2.3 Вибір сервера MS SQL Server.....	24
2.4 Вибір мови програмування C#.....	25
3 Проектування інформаційної системи	28
3.1 Побудова функціональних діаграм	28
3.2 Побудова діаграм потоків даних	32
3.3 Проектування та створення бази даних.....	35
3.4 Проектування інтерфейсу сайту.....	41
4 Опис інтерфейсу користувачів інформаційної системи.....	44
4.1 Опис інтерфейсу менеджера готелю.....	44
4.2 Опис інтерфейсу клієнта готелю	55
Висновки	58
Перелік джерел посилання	60

ВСТУП

На сьогоднішній день неможливо уявити ефективне функціонування готелю без використання спеціалізованого програмного забезпечення, що дає можливість автоматизації процесів, які відбуваються в готельному бізнесі. Для інформаційних систем підприємств готельного бізнесу у сучасних умовах висувуються найвищі вимоги задля того, щоб оперативно реагувати на потреби ринку та задовольняти всі побажання клієнтів.

Для автоматизації бізнес-процесів готелю використовуються комп'ютерні мережі, глобальна мережа Інтернет, автоматизовані системи, спеціалізовані інформаційні системи.

Інформаційні технології, які використовуються в індустрії готельного бізнесу мають комплексний характер, складаються, як правило, з комп'ютерної системи бронювання, системи проведення телеконференцій, відеосистем, комп'ютерів, інформаційних систем управління, електронних інформаційних систем авіаліній, електронної пересилки грошей, телефонних мереж, глобальних комп'ютерних системи резервування, інтегрованих комунікаційних мереж, системи мультимедіа, смарт-карток.

В залежності від розміру готельного підприємства, структура підрозділу може налічувати лише чергового адміністратора або повний штат служби прийому та розміщення, який складається з портьє, касира, службовця із бронювання, адміністратора та ін. Також, потрібно пам'ятати, що основною задачею служби прийому та розміщення є робота з гостем, яка займає досить багато часу. Автоматизація інших функцій підрозділу, а саме бронюванню, надасть змогу службовцям раціонально використовувати робочий час та приділяти більше уваги гостю. Таким чином, оптимізація роботи служби прийому та розміщення визначає актуальність впровадження технології інтеграції он-лайн бронювань безпосередньо до системи управління готелем.

Актуальність теми обумовлена тим, що впровадження інформаційної системи в міні-готелі підвищить його конкурентноспроможність, дозволить

вдосконалити його діяльність, доповнить роботу адміністраторів та менеджерів елементами аналізу та контролю.

Об'єктом дослідження роботи є міні-готель.

Предметом дослідження є процеси бронювання, обліку готельних номерів, діяльність готелю в цілому.

Метою кваліфікаційної роботи є розробка інформаційної системи бронювання та обліку номерів у готелі.

Для вирішення поставленої задачі використано Ramus Educational, MS SQL Server 2014, MS Visual Studio 2017.

Пояснювальна записка містить 60 сторінок, 44 рисунки, 5 таблиць, 10 джерел посилання.

1 ХАРАКТЕРИСТИКА ПРЕДМЕТНОЇ ОБЛАСТІ. ПОСТАНОВКА ЗАВДАННЯ

1.1 Використання інформаційних технологій в готельному бізнесі

В українській готельній індустрії на даному етапі гостро стоїть питання якості надання готельних послуг. За даними державної служби статистики України кількість іноземних туристів, які відвідали країну зростає. Це спонукає готельні підприємства на пошук нових шляхів комунікації з туристами та підвищувати рівень якості наданих послуг. Конкурентна боротьба між готельними підприємствами спонукає їх до пошуку насамперед нових інформаційних технологій [1]¹⁾.

На даний момент кожен сучасний конкурентоспроможний готель використовує різноманітні комп'ютерні програми та спеціальне програмне забезпечення для автоматизації роботи. Системи управління готелями (HMS – Hotel Management System) – спеціальне програмне забезпечення для ефективного управління готелю та номерним фондом. Дане програмне забезпечення дозволяє автоматизувати такі процеси, як: бронювання (наявність вільних номерів, створення та відміна бронювання), реєстрація заїзду та виїзду (робота з «шахматкою»), розрахунок гостя, контроль роботи покоївок (формування задач по прибиранню на день/місяць), складання звітів та ін.

Різнманіття систем управління готелями на ринку, дозволяє готельному підприємству обрати найефективніше програмне забезпечення для управління готелем. Так, для великих та сітьових готельних підприємств пропонуються такі системи, як Opera та Fidelio. Такі програми, як ProHotel, Shelter, SmartLab, SuperHotel використовують переважно невеликі готельними підприємствами [2]²⁾.

¹⁾ [1] Державний комітет статистики України. Офіційний сайт. URL: <http://ukrstat.gov.ua> (дата звернення 18.04.2019).

²⁾ [2] Ткаченко В. Гостиничные системы: функции и возможности. Сети & Бизнес. 2011. №2(57). С.74-84

Жоден сучасний готель у наш час не уявляє своє існування без офіційного сайту. Це інструмент, який є обов'язковим для просування своїх послуг на ринку готельної індустрії. Відсутність офіційного сайту готельного підприємства сприймається в наш час, як показник низького рівня якості надаваних послуг готелю. Також, більшість сучасних готельних підприємств використовують платформи таких соціальних мереж, як Facebook та Instagram для просування своїх послуг. Користувачі, які звернули увагу на готель у соціальних мережах майже у більшості своїй відвідують офіційні сайти готелю.

Незважаючи на наявність офіційних сайтів з модулями бронювання та прямих контактів (телефон, електронна пошта, соціальні мережі), готельні підприємства приділяють велику увагу комп'ютерним системам бронювання. Такі лідери ринку глобальних систем бронювання (GDS), як Amadeus, Galileo, Sabre та Worldspan були створені насамперед для туристичних фірм, а не для безпосередніх користувачів та потенційних гостей саме готелю. Тому, зараз популярними платформами он-лайн бронювань для приватних осіб є альтернативні системи бронювання (ADS). Найбільш поширеними ADS у світі та Україні є Booking.com, Expedia.com, HRS, Agoda, Ostrobok.ru, Hotels24.ua, Ozon, Strip та ін.

Готельне підприємство отримує запити на бронювання послуг різноманітними шляхами. Це можуть бути прямі бронювання (офіційний сайт, телефон, електронна пошта, соціальні мережі), а також непрямі, за допомогою систем он-лайн бронювання. Можна спостерігати тенденцію високої підтримки соціальної активності за рахунок значного використання мережі Інтернет для бронювання готелів. Перевагою непрямих бронювань є гарантія. Так, коли приватна особа бронює номер, вона зобов'язана надати гарантію своєї платоспроможності у разі заїзду або оплати штраф у випадку відміни бронювання. Перевагою прямих бронювань є те, що готельне підприємство не зобов'язане сплачувати комісію ADS. Різноманіття каналів бронювання підвищує рівень продаж номерного фонду готельного підприємства, але й додає роботи службі

прийому та розміщення, які повинні постійно фіксувати кількість вільних номерів та її синхронізацію в усіх каналах.

1.2 Огляд інформаційної системи Fidelio

Система Fidelio [3]¹⁾ є складовою частиною глобальних комп'ютерних систем бронювання Amadeus, Sabre, Galileo, Worldspan. Всі готелі, представлені в Fidelio, автоматично завантажуються в ці глобальні комп'ютерні системи бронювання.

До основних функцій системи Fidelio відносяться: бронювання і заселення гостей, нарахування за проживання та інші послуги, що надаються готелем, акумулювання інформації про неоплачені рахунки клієнтів, що надходять з різних точок продажів, виставлення проміжних і остаточних рахунків для розрахунків з клієнтами, облік інформації про безготівкові розрахунки, отримання фінансових і статистичних звітів, ведення інформації про клієнтів і клієнтських рахунках: ведення історії по клієнтах, бронювання, заселення, ведення рахунків, виписка, безготівкові розрахунки з клієнтами.

Картка гостя містить інформацію про клієнта: прізвище, ім'я, мову, код VIP, адреса, країна проживання, телефон, факс, дата народження, бажаний готельний номер, терміни проживання (вимірюються ночами), число осіб, що проживають в номері, тип і категорія кімнати, ціновий код, тип бронювання (гарантоване або негарантований).

Система Fidelio FO складається з наступних основних модулів:

Модуль бронювання дозволяє завести бронь для клієнта, завести групову бронь, скасувати бронь, коригувати бронь, перенести бронь в список чекаючих, призначити конкретні номери по раніше зробленому бронюванню, роздрукувати підтвердження бронювання, прийняти депозит для бронювання.

¹⁾ [3] Система управління «Fidelio». URL: https://studopedia.su/5_45088_FIDELIO-V.html (дата звернення 20.04.2019).

Модуль порт'є. Він призначений для поселення клієнтів готелю, що мають бронь: пошук броні, призначення номерів і заселення, зміна броні, Відміна бронювання, друк підтверджень, підселення, прийом повідомлень для прибуваючих або прибулих гостей, прийом депозиту для бронювання, переміщення броні в список очікування, заклад службових рахунків готелю.

Модуль касира: виконувати поточну роботу з рахунками клієнтів, виставляти проміжні рахунки, переселяти клієнта в інший номер, здійснювати виписку, здійснювати ведення головної каси, здійснювати операції обмінного пункту, здійснювати прийом депозиту для бронювання, друкувати звіти для касирів, виправляти бронь клієнта, здійснювати прийом оплат, перенести нарахування з одного рахунку на інший, ділити та поєднувати нарахування, виконувати ручні нарахування на рахунки клієнтів.

Бухгалтерський модуль: виробляти деякі маніпуляції з нарахованими сумами (переносити нарахування з одного рахунку на інший, вводити оплату по кожній позиції, відстежувати історію по клієнтах, друкувати звіти і листи клієнтам).

Система Fidelio надає можливість складати різного роду звіти і прогнози, які необхідні для подальшого планування діяльності готелю.

1.3 Система автоматизації готелів JSolution

В сучасній Україні досить новою та популярною є інформаційна система для автоматизації роботи готелів JSolution [4]¹⁾. Група українських розробників пропонує цю систему з використанням хмарних технологій для автоматизації бізнеса з 2017 року. Головне вікно системи, яку можна продивитись безкоштовно в тестовому режимі надано на рис.1.

¹⁾ [4] Інформаційна система JSolution. URL: <https://jsolutions.ua/testoviy-dostup#hotel> (дата звернення 22.04.2019).

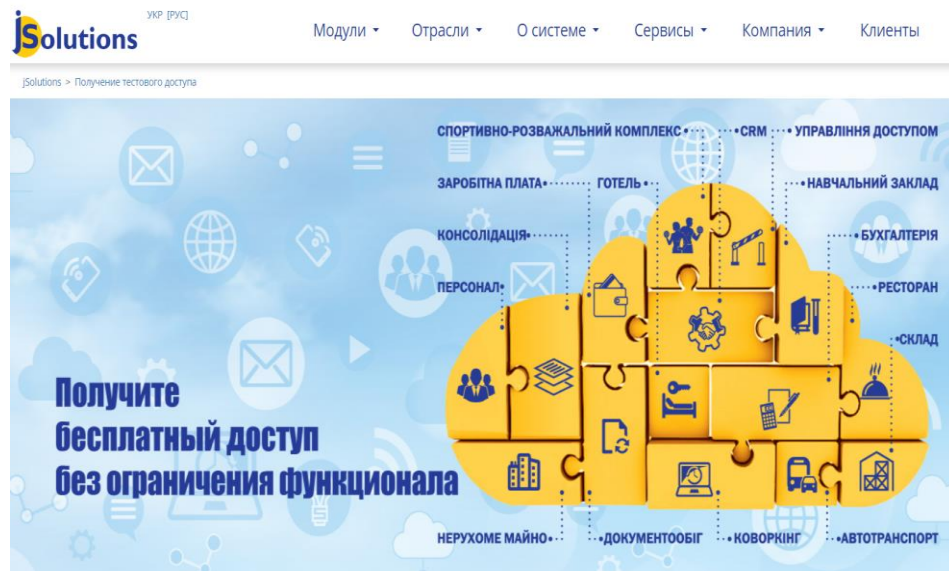


Рисунок 1 – Стартова сторінка системи JSolution

Особливості системи автоматизації готелів jSolutions:

- єдина база даних для будь-якої кількості готелів та корпусів;
- облік завантаження номерного фонду;
- розрахунки з контрагентами та гостями;
- система лояльності для клієнтів (дисконтні, подарункові карти);
- прийом різних видів оплати: готівка, карта, дисконтна карта;
- формування завдань на прибирання приміщень, планування і контроль виконання робіт;
- інтеграція системи з сайтом готелю;
- можливість роботи з використанням web-інтерфейсу;
- інтеграція з електронними замками, турнікетами, шлагбаумами;
- використання електронного ключа (брелок, карта, браслет);
- підтримка користувачів здійснюється по всій Україні.

1.4 Система управління та бронювання Lite PMS

Ця система дозволяє управляти будь-якими об'єктами для розміщення гостей на добу або на більш довгий строк. Вона може використовуватись у

готелях, хостелах, санаторіях, домах відпочинку, а також у випадках здачі нерухомості в оренду [5]¹⁾. На сайті можна проглянути демонстраційну версію роботи цієї системи.

Загальна інформація про номери, кількість постояльців наводиться у вигляді таблиці (рис.2). Тут можна обрати період перегляду, категорію номерів, а також здійснити пошук інформації про заброньовані номери. Зручним є кольорова індикація зайнятих та вільних номерів.

Lite PMS [Календарь](#) [Список броней](#) [Номера](#) [Клиенты](#) [Задачи](#) [Касса](#) [Отчеты](#) [Уведомления](#) [Настройки](#) test1, info@litepms.ru | Выход

Новое бронирование Новый клиент Новая задача Новое SMS Новое письмо

Поиск свободных номеров

Сегодня, 31 марта, 22:27:35

Завезды	15	Всего номеров	58 (65)
Выезды	19	Всего мест	186 (66)
Проживания	3	Свободно номеров	50 (49)
Дни поселения	1	Занято номеров	8 (16)
Задачи	0 (4)	Загрузка	14% (25%)

Внимание! Это тестовый аккаунт, не оставляйте здесь реальные данные, используйте только для изучения возможностей сервиса. Чтобы создать свой аккаунт, нажмите Выход, затем Регистрация.

29-03-2019 сегодня категория: отображать все поиск брони: ФИО клиента период: 30 дней

Номера	Март 2019							Апрель 2019																	
	Пт	Сб	Вс	Пн	Вт	Ср	Чт	Пт	Сб	Вс	Пн	Вт	Ср	Чт	Пт	Сб	Вс	Пн	Вт	Ср	Чт	Пт	Сб	Вс	
CDJ 6																									
1 этаж 1		чистый																							
2		чистый																							
caustom category																									
custom room 1			сертергерг	сертергерг										инкогнито г					серте						сертергерг
двухместный номер этаж 2, корпус Береговая За		грязный	серте	серте									ваши												
Сплав по реке																									
Сплав экскурсия		чистый																							
Стандарт																									
1		чистый	Козлов																						
2		чистый		Оспанов Аскар																					

Рисунок 2 – Загальна інформація про номери

Для нового бронювання створена форма (рис.3), в якій вводяться дані про дати заїзду, виїзду, кількість місць, обраний номер та побажання клієнтів. Адміністратор може залишити свій коментар.

¹⁾ [5] Lite PMS – система управління для власників апартаментів, хостелів, готельних номерів. Модуль бронювання онлайн. URL: <https://litepms.ru> (дата звернення 25.04.2019).

Новое бронирование

Информация о заезде Данные о клиенте Услуги Стоимость Платежи Счета Задачи Документы

Источник брони*:

Номер брони у источника:

Выберите номер*:

спальное место

Стоимость проживания:

Своя цена: за сутки

Заезд*: 31-03-2019 время 13:00

Выезд*: количество суток 01-04-2019 время 12:00

Основных мест*: 1 Доп. мест:

Статус: Не подтверждено Метка: без метки

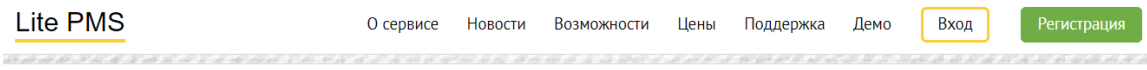
Комментарий администратора:

Пожелания клиента:

[Сохранить](#) [Сохранить и остаться](#) [Резерв](#) [Отмена](#)

Рисунок 3 – Форма бронирования номеров

Наглядная картина занятых номеров наведена у формі (рис.4).



Наглядная шахматка

Номера	Март 2016							Апрель 2016									
	Вт	Ср	Чт	Пт	Сб	Вс	Пн	Вт	Ср	Чт	Пт	Сб	Вс	Пн	Вт	Ср	Чт
Одноместный номер																	
102				Полева Алекс													
103							Горюнов										
104							Полева Александр										
Двухместный номер																	
201				Вячеслав			Серг			Петр							
202			Игорь								Иван						
1																	Олег
2																	
Трехместный номер																	
123							Александр		Юлия		Екатерина						Вас
302																	
Без категории																	
101																	
303																	

Рисунок 4 – Форма занятых номеров

Зручною функцією цієї системи є органайзер, за допомогою якого можна скласти список задач (рис.5).

← [скрыть календарь](#)

31 марта, воскресенье
22:30:33

Новая задача Быстрая задача

Список задач на 31 марта

сегодня < > период: 7 дней автор: все ответственный: все

Просроченные задачи

28 марта 12:00	важно	Встреча гостя в аэропорту (test1 - Иван Иванович)
28 марта 12:00	важно	Встреча гостя в аэропорту (test1 - Иван Иванович)
28 марта 12:00		Встреча гостя в аэропорту (test1 - Иван Иванович)
28 марта 12:00	важно	Встреча гостя в аэропорту (test1 - Иван Иванович)

7 апреля, воскресенье

12:00	важно	Убрать номер после выезда гостя
-------	--------------	---------------------------------

Рисунок 5 – Органайзер системи

Проаналізувавши тестову версію системи, можна виділити всі можливості системи:

- наочна шахматка і окремий журнал замовлень;
- модуль бронювання для вашого сайту, з можливістю вбудованої завантаження в ваш сайт, без спливаючих вікон і переходу на наш сервіс;
- онлайн оплата броні без посередників і додаткових комісій;
- необмежена кількість категорій і номерів;
- бронювання за койко-місце;
- погодинне бронювання;
- ведення клієнтської бази (CRM);
- створення сезонних тарифів і тарифів в залежності від кількості зайнятих місць в номері;
- створення додаткових послуг та знижок;
- звукове сповіщення про новий бронюванні;
- створення розширеної анкети клієнта;
- створення додаткових користувачів з гнучким налаштуванням прав доступу;

- журнал дій співробітників;
- побудова звітів;
- створення будь-яких друкованих форм;
- модуль Завдання (Task Manager);
- модуль Каса для фіксації доходів і витрат об'єкта розміщення;
- SMS і E-mail повідомлення клієнтам і адміністраторам;
- створення резервних копій та їх збереження, відновлення даних;
- управління замовленнями з будь-якого пристрою, з будь-якого місця.

1.5 Автоматизована система HOTEL 2000

Для готельних закладів України існує відносно недороге і практичне рішення, яке містить у собі систему автоматизації діяльності готельних служб (HOTEL 2000), систему автоматизації барів і ресторанів (Restaurant 2000), автоматизовану систему керування складом ресторану (Stock 2000), автоматизований інтерфейс з системами бухгалтерського обліку, автоматизований інтерфейс з внутрішньою АТС, автоматизований інтерфейс з системою обмеження прав доступу (електронні замки).

Система Hotel 2000 – це класична front office програма, яка здійснює автоматизацію основних готельних служб – бронювання, порт'є, касирів, покоївок тощо. Система забезпечує індивідуальне й групове бронювання номерного фонду, а також бронювання компаніями і туроператорами блоків номерів з відстеженням квот по кожній організації.

Система Hotel-2000 передбачає виконання таких функцій:

- управління номерним фондом, оптимізацію завантаження готелю та своєчасне надання актуальної інформації про стан номерів;
- управління роботою покоївок і контроль за своєчасним прибиранням номерів;

- індивідуальне та групове бронювання в реальному режимі часу з перевіркою наявності вільних номерів і можливістю оптимізації завантаження готелю;
- бронювання блоків номерів з гарантованою і негарантованою формами оплати;
- перегляд і зміна інформації про бронювання і моментальне відновлення скасованих броней і незаїзду;
- автоматизацію процедур реєстрації та оформлення індивідуальних гостей і груп як за попередньою броні, так і без неї з моментальним пошуком необхідного для гостя номера;
- виписку рахунків з автоматичним нарахуванням вартості проживання та наданих додаткових послуг;
- отримання в будь-який момент на вимогу клієнта всієї необхідної інформації про проживання з видачею проміжних рахунків;
- ведення архіву гостей;
- швидке і просте щодобове закриття дня, що дозволяє здійснити все нарахування за минулу добу, щоб переглянути роботу касирів без зупинки роботи готелю;
- управління ціновою політикою готелю з урахуванням сезонних коливань, умов взаємин з партнерами і т.д .;
- управління фінансами готелю;
- ведення бухгалтерського обліку.

Сайт Booking.com це система бронювання житла в мережі Інтернет. На вебсайті розміщено більше 28,616,893 об'єктів у 226 країнах і територіях. За добу бронюється 1,550,000 кімнат. Сервіс доступний на 43 мовах. Система надає можливість пошуку за багатьма критеріями, можна переглянути відгуки клієнтів, розташування готелю на мапі, фото номерів, опис номеру та додаткових послуг. Головна сторінка представлена на рис. 6.

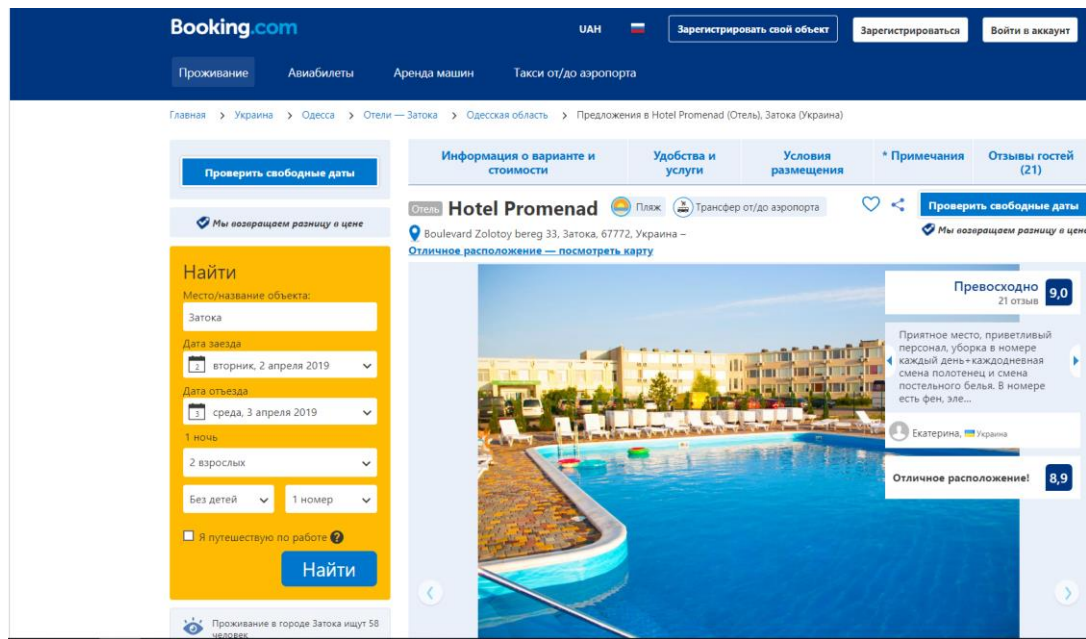


Рисунок 6 – Головна сторінка системи Booking.com

Проаналізувавши ці системи, можна зробити висновки: інформаційні системи в готельному бізнесі є економічно доцільними та ефективними за умови, якщо вони приносять готелю додаткові доходи, забезпечують конкурентні переваги на ринку, підвищують частку ринку, знижують витрати, вдосконалюють процес обслуговування, підвищують ефективність роботи окремих підрозділів та готелю в цілому. Також впровадження даних систем може кардинально змінювати методичну, інформаційну та технологічну складові управлінських процесів і здійснювати їх на якісно новому, більш ефективному рівні, що значно підвищує якість надання послуг у готелі.

Згідно даних, Україна відстає від провідних розвинених країн світу в сфері впровадження інформаційних технологій і автоматизації управління готелями. Проте в той же час є великі перспективи подальшого розвитку в цьому напрямку, оскільки на українському ринку є понад десяти високоякісних сучасних систем автоматизації управління готелями, причому деякі з них - це не адаптація зарубіжних систем, а оригінальні розробки українських компаній і тому найбільш пристосовані до умов України. Впровадження таких систем істотно підвищить конкурентоспроможність українських готелів.

1.6 Постановка завдання

Розробити інформаційну систему для автоматизації роботи міні-готелю. Необхідно забезпечити високу продуктивність і відмовостійкість системи, а також легкість адміністрування системи.

Програмний комплекс повинен складатися з серверної частини, модуля управління системою, з яким взаємодіятиме адміністратор і користувачі, а також базою даних, що представляє постійне сховище даних системи.

Так само повинен бути створений Інтернет-сайт, що надає послуги кінцевим користувачам. Сайт повинен дозволяти переглядати інформацію про номери готелю, послуги, надавати функції пошуку даних, відповідно до побажань клієнтів. Крім того, сайт повинен дозволяти переглядати фотографії номерів та території.

Важливою вимогою є забезпечення безпеки інформації. Система не повинна надавати прямий доступ до бази даних ні для однієї з підсистем, за виключенням сервера застосувань. Сервер застосувань, у свою чергу, не повинен містити інтерфейс доступу до бази даних. Потрібно реалізувати систему розмежування прав доступу на основі Access Control List.

Список функціональних можливостей системи:

Створення, видалення, редагування інформації про номери готелю. Для кожного об'єкту передбачити можливість завантаження фотографій. Кількість зображень не обмежена.

Створення, видалення, редагування інформації про додаткові послуги готелю.

Реєстрація і обробка замовлень. Передбачити можливість здійснення бронювання клієнтом через Web-інтерфейс застосування, а так само можливість замовлення номеру безпосередньо з робочого місця користувача клієнтського застосування (адміністратора готелю), у разі обслуговування клієнтів в готелі.

Реалізувати гнучку систему пошуку номерів готелю. Пошук повинен бути можливий по всіх атрибутах.

Забезпечити можливість редагування, створення і видалення застарілої інформації. Відразу ж після редагування всі зміни повинні бути доступні на Інтернет-сайті.

Забезпечити можливість створення, видалення і редагування груп і користувачів.

2 ВИБІР ТА ОБГРУНТУВАННЯ ПРОГРАМНИХ ЗАСОБІВ ДЛЯ РЕАЛІЗАЦІЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ МІНІ-ГОТЕЛЮ

2.1 Обґрунтування вибору архітектури

У даній роботі реалізована, так звана, "трирівнева архітектура". Це дозволило створити високопродуктивний додаток, який просто підтримувати і масштабувати. Завдяки даній архітектурі можливо використання одного і того ж програмного інтерфейсу як для роботи клієнтського додатку, так і для роботи Web-сайту [6]¹⁾.

Архітектура клієнтського застосування побудована на основі бібліотеки Composite UI від компанії MicroSoft. Завдяки цьому, стало можливим забезпечити широкі можливості щодо інтеграції додаткових модулів безпосередньо в інтерфейс застосування без модифікації основного коду. Використання даної технології дозволило створити модульну структуру, в якій немає жорстких обмежень на функціональність. Кожен модуль ізольований від системи до певного ступеня і вільний сам вирішувати яку функціональність надавати системі.

Трирівневий додаток складається з наступних компонентів (рис.7).

Інтерфейсний (звичайно графічний) компонент представляє перший рівень. Це може бути, як віконний додаток, так і консольний клієнт. У багатьох випадках цей рівень може бути представлений додатком з Web-інтерфейсом. Перший рівень не повинен мати прямих зв'язків з базою даних і зберігати стан додатку.

На другому рівні розташовується сервер додатків. Це може бути програмна система, написана на будь-якій сучасній мові програмування, але, як правило, найчастіше реалізація створюється на мовах Java C# або C++. На другому рівні зосереджена велика частина прикладної логіки. Поза ним

¹⁾ [6] Фаулер Мартин. Архитектура корпоративных программных приложений. СПб.: Вильямс, 2006. 544 с.

залишаються фрагменти, що експортуються на клієнтські системи, а також, занурені в третій рівень, такі, що зберігаються процедури і тригери.

Третій рівень забезпечує зберігання даних. Звичайно це стандартна реляційна або об'єктно-орієнтована СУБД. Якщо третій рівень є базою даних разом з процедурами, що зберігаються, тригерами і схемою, що описує додаток в термінах реляційної моделі, то другий рівень будується як програмний інтерфейс, що пов'язує клієнтські компоненти з прикладною логікою бази даних.

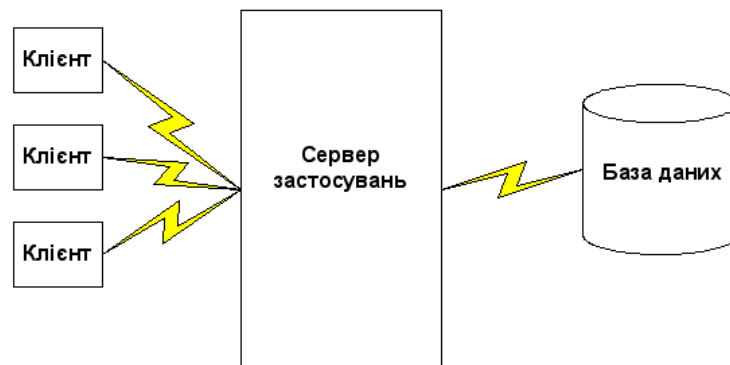


Рисунок 7 – Трирівнева архітектура «клієнт-сервер»

Завдання багаторівневої архітектури – ізолювати рівні один від одного. У трирівневій архітектурі відносно легко можна поміняти СУБД (наприклад, з Oracle на MSSQL або, навпаки) – для цього не потрібно переписувати додаток клієнта наново. Також, відносно легко можна створити новий тип клієнта (наприклад, є віконний клієнт з багатим призначенням для користувача інтерфейсом, а ми додаємо Web-клієнта).

Основними перевагами трирівневої архітектури в порівнянні з дворівневою є:

- масштабованість;
- конфігуруємість;
- можливість модифікації бізнес-логіки;
- багатократне використання сервісів.

2.2 Вибір платформи .NET Framework

Як платформа для розробки програмного комплексу був обраний .NET Framework версії 4.5 [7]¹⁾.

.NET Framework – це середовище виконання для додатків нового покоління. Це середовище може бути встановлене в будь-які версії Windows, і забезпечити віртуальну систему виконання (Virtual Execution System) для додатків, створених за допомогою Visual Studio.NET. Крім того, невід'ємною частиною .NET Framework є підсистема ASP.NET, що дозволяє створювати динамічні Інтернет-сторінки з мінімумом витрат.

ASP.NET – це технологія створення Web-застосувань і Web-сервісів від компанії Microsoft. Вона є складовою частиною платформи Microsoft .NET і розвитком старішої технології Microsoft ASP. ASP.NET зовні багато в чому зберігає схожість із старішою технологією ASP, що дозволяє розробникам відносно легко перейти на ASP.NET. В той же час внутрішній устрій ASP.NET істотно відрізняється від ASP, оскільки вона заснована на платформі .NET і, отже, використовує все нові можливості, що надаються цією платформою.

Переваги ASP.NET перед ASP:

- компільований код виконується швидше, більшість помилок відловлюються ще на стадії розробки;
- значно поліпшена обробка помилок часу виконання, з використанням блоків try..catch;
- призначені для користувача елементи управління дозволяють виділяти часто використовувані шаблони, такі як меню сайту;
- використання метафор, що вже застосовуються в Windows – застосуваннях, наприклад, таких як елементи управління і події;
- розширюваний набір елементів управління і бібліотек класів дозволяє

¹⁾ [7] Дон Бокс, Крис Селлз. Основы платформы .NET. Общеязыковая исполняющая среда. СПб.: Вильямс, 2003. 288 с.

- швидше розробляти застосування;
- ASP.NET спирається на багатомовні можливості .NET, що дозволяє писати код сторінок на VB.NET, C#, J# і т.д.;
- можливість кешування всієї сторінки або її частини для збільшення продуктивності;
- можливість розділення візуальної частини і бізнес логіки по різних файлах («code behind»).

Істотною перевагою конструктивного рішення .NET є компонентно-орієнтований підхід до проектування і реалізації програмного забезпечення. Суть підходу полягає в принциповій можливості створення незалежних складових програмного забезпечення з уніфікованою інтерфейсною частиною для багатократного повторного і розподіленого використання. При цьому продуктивність рішення обумовлена багатомовністю інтегрованих програмних проектів (концепція .NET потенційно підтримує довільну мову програмування, в числі найбільш відомих мов – C#, Visual Basic C++ і ін.).

У результаті програмний проект формується у вигляді збірки – самодостатнього компоненту для розгортання, тиражування і повторного використання. Збірка ідентифікується цифровим підписом автора і унікальним номером версії.

Середовища розробки прикладних систем доцільно використовувати як Microsoft Visual Studio .NET, що надає цілий комплекс розвинених засобів створення, редагування і відладки програмного коду на різних мовах програмування. У разі нескладних завдань можна обмежитися примітивними редакторами тексту програм, подібних Notepad.

Інтерфейсна частина прикладної програмної системи в Internet-архітектурі представлена так званими веб-формами, призначеними для введення і виведення даних в уніфікованому форматі.

Як мова реалізації може використовуватися мова гіпертекстової розмітки HTML. Взаємодія між клієнтом і додатком в простому випадку здійснюється з використанням традиційного Internet-протоколу передачі даних HTTP.

2.3 Вибір сервера MS SQL Server

Як сервер бази даних був використаний MS SQL Server. Це розробка компанії MicroSoft, що володіє великою кількістю додаткових можливостей[8]¹⁾. Інтеграція .NET Framework з SQL Server надає розробникам баз даних декілька істотних переваг:

- більш довершена модель програмування;
- більш довершений захист;
- призначені для користувача типи і агрегати;
- універсальне середовище розробки;
- продуктивність і масштабованість.

У SQL Server реалізовані три рівні захисту, що обмежують можливості зареєстрованих в базі даних. Ця модель захисту інтегрує модель SQL Server, засновану на аутентифікації і авторизації користувачів, і модель CLR, що використовує захист з прав доступу коду.

Керований код на рівні доступу до даних дозволяє проводити над об'єктами бази даних складні математичні розрахунки і застосовувати будь-яку логіку. .NET Framework забезпечує обширну підтримку роботи з рядками, регулярних виразів, перехоплення помилок і ін. Функціональність базової бібліотеки класів .NET Framework дозволяє дістати повний доступ до тисяч вже розроблених класів і підпрограм, до яких легко звернутися з будь-якої процедури, що зберігається, тригера або UDF. У ситуаціях, де потрібні обробка рядків, математичні функції, робота з датами, доступ до системних ресурсів, просунуті алгоритми шифрування, доступ до файлів, обробка зображень або маніпулювання XML-даними, виявляється, що керовані процедури, що зберігаються, функції, тригери і агрегати надають значно простішу програмну модель, чим T-SQL.

¹⁾ [8] Вийера Роберт. Программирование баз данных Microsoft SQL Server 2008. Базовый курс. СПб.: Вильямс, 2007. 832 с.

Ще одна перевага керованого коду – контроль типів. Перед виконанням керованого коду CLR здійснює на стороні сервера перевірки, щоб упевнитися, що запуск коду дійсно безпечний. Наприклад, перевіряється, як код звертається до пам'яті, щоб не було читання з тієї області, в яку нічого не записано.

Можливості по інтеграції з середовищем CLR, висока продуктивність і стабільність, широка і якнайповніша (серед нині існуючих серверів баз даних) підтримка XML – ось основні причини, по яких була вибрана саме ця СУБД.

2.4 Вибір мови програмування C#

Разом з випуском нової платформи .NET компанія Microsoft представила нову мову, ідеально відповідну для нової платформи – C#. При створенні мови в основу дизайну лягла легкість використання, домінуюча над потужністю мови і швидкістю виконання. Звідси і складальник сміття з керованими об'єктами посиленнями, який автоматично звільняє за Вас пам'ять, відбираючи при цьому процесорний час. Ви також одержуєте безпеку роботи з типами, а це, на думку багатьох, є другим найважливішим чинником уникнення помилок.

C# об'єктно-орієнтована мова, як і вся платформа .NET. Більш того, це мова, орієнтований на написання компонент. C# створений для програмування в керованому середовищі з присутнім складальником сміття, але дозволяє писати і некерований код. Цілісність концепції видно і в реалізації пакування/розпакування. Це дозволяє розглядати всі типи (навіть примітивні) як об'єкти, що вирішує багато проблем дизайну додатків.

У C# представлена концепція просторів імен, аналогічна пакетам в Java. Це дозволяє ієрархічно структурувати Вашу систему типів, роблячи код набагато зрозумілішим і дозволяючи уникнути проблем з іменуванням. Ви можете розглядати простори імен як директорії, а мовні типи як файли в цих директоріях.

У мові для роботи з подіями і зворотними викликами представлена нова

концепція об'єктно-орієнтовано і безпечною з погляду системи типів посилення на функцію-"делегат". Саме ця концепція дозволяє описувати події C# компоненту без поворотних інтерфейсів і додаткових зусиль, досить використовувати ключове слово `event`.

C# дозволяє використовувати при розробці парадигму атрибутивного програмування, коли в модулях, компіляцій, разом з типом може зберігатися мета – інформація. Дана інформація згодом може бути одержана і використана під час виконання. Більш того, цією інформацією може скористатися компілятор або, скажімо, дизайнер, що дозволяє, не запускаючи компонент, одержувати інформацію про нього.

Робота з атрибутами дозволяє використовувати COM COM+ і DLL в C# додатках, що працюють в середовищі .NET. Переліки дозволяють забезпечити роботу з перераховуваними типами. Реалізація структур як типів, робота з якими йде по значенню, укупі з можливістю використовувати не тільки вкладені масиви (як в Java), але і багатовимірні дозволяє оптимізувати продуктивність додатків.

C# увібрав в себе все краще з таких популярних мов як C++, Visual Basic, Java і Object Pascal. C# забезпечує швидку розробку, в той же час дозволяє писати ефективний код. Перерахуємо особливості мови [9]¹⁾:

- автоматична збірка сміття;
- можливість маніпулювати покажчиками і мати безпосередній доступ до пам'яті;
- підтримка властивостей і подій (аналогічно VB);
- підтримка атрибутів;
- вбудована підтримка основних типів (рядок, масив, ...);
- множинне спадкоємство можливо тільки від інтерфейсів (як в Java);
- підтримка 3 API, Windows API і COM+ на рівні мови;

¹⁾ [9] Дейтел Х. C# в подлиннике. Наиболее полное руководство. СПб.: БХВ-Петербург, 2006. 1056 с.

- підтримка асемблювання;
- контроль типів;
- автоматична ініціалізація змінних.

C# створювався як мова компонентного програмування, і в цьому одне з головних переваг цієї мови, яке направлене на можливість повторного використання створених компонентів.

3 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

3.1 Побудова функціональних діаграм

Проектування є важливим етапом, від якого залежать подальші етапи розробки системи. Час, витрачений розробником на проектування, зазвичай окупається високою швидкістю реалізації проекту.

Процес проектування починається зі збору інформації про досліджувану область, документування отриманої інформації і підготовці опису. На основі такого опису здійснюється визначення складу і структури даних, які повинні знаходитися в базі даних і забезпечувати виконання необхідних запитів і завдань користувача. Проведений аналіз даних та функцій системи відображається у вигляді моделей. На їх основі легко створюється реляційна база даних.

SADT (Structured Analysis and Design Technique) – технологія структурного аналізу і проектування (стандарт IDEF0). Методологія SADT являє собою сукупність методів, правил і процедур, призначених для побудови функціональної моделі об'єкта будь-якої предметної області. Широко використовується в проектуванні бізнес-процесів, технічних об'єктів, організаційних систем і т.д.[10]¹⁾. Претендує на роль обов'язкового інструментального засобу створення будь-яких проектів на ранніх стадіях.

Модель SADT відображає функціональну структуру об'єкта, тобто приведені ним дії й зв'язки між цими діями. Результатом застосування методології SADT є модель, яка складається з діаграм, які мають посилання один на одного. Діаграми – головні компоненти моделі, всі функції системи і інтерфейси на них представлені як блоки і дуги. Управляюча інформація (інструкції, нормативи, стандарти) входить в блок зверху, інформація, яка піддається обробці, показана з лівого боку, а результати виходу показані з правого боку. Механізм (людина або автоматизована система), який здійснює операцію,

¹⁾ [10] Девід А. Марка, Клемент Л. МакГоуэн. Методология структурного анализа и проектирования SADT. М.: Мета Технология, 1993. 242 с.

представляється дугою, що входить в блок знизу. Однією з найбільш важливих особливостей методології SADT є поступове введення все великих рівнів деталізації у міру створення діаграм, що відображають модель.

Побудова SADT-моделі починається з представлення всієї системи у вигляді простої компоненти – одного блоку і дуг, що зображають інтерфейси з функціями поза системою. Оскільки єдиний блок представляє всю систему як єдине ціле, ім'я, вказане в блоці, є загальним. Це вірно для інтерфейсних дуг – вони також представляють повний набір зовнішніх інтерфейсів системи в цілому.

Потім блок, який представляє систему як єдиний модуль, деталізує на іншій діаграмі за допомогою декількох блоків, сполучених інтерфейсними дугами. Ці блоки представляють основні підфункції початкової функції. Дана декомпозиція виявляє повний набір підфункцій, кожна з яких представлена як блок, межі якого визначені інтерфейсними дугами. Кожна з цих підфункцій може бути декомпозована так само для детальнішого уявлення.

У всіх випадках кожна підфункція може містити тільки ті елементи, які входять в початкову функцію. Крім того, модель не може опустити які-небудь елементи, тобто, як вже наголошувалося, батьківський блок і його інтерфейси забезпечують контекст. До нього не можна нічого додати, і з нього не може бути нічого видалено.

Контекстна діаграма інформаційної системи готелю представлена на рис.8. Модель SADT є серією діаграм з супровідною документацією, що розбивають складний об'єкт на складові частини, які представлені у вигляді блоків. Деталі кожного з основних блоків показані у вигляді блоків на інших діаграмах. Кожна детальна діаграма є декомпозицією блоку з більш загальної діаграми. На кожному кроці декомпозиції більш загальна діаграма називається батьківською для детальнішої діаграми.

Дуги, що входять в блок і виходять з нього на діаграмі верхнього рівня, є точно тими ж самими, що і дуги, що входять в діаграму нижнього рівня і

виходять з неї, тому що блок і діаграма представляють одну і ту ж частину системи.

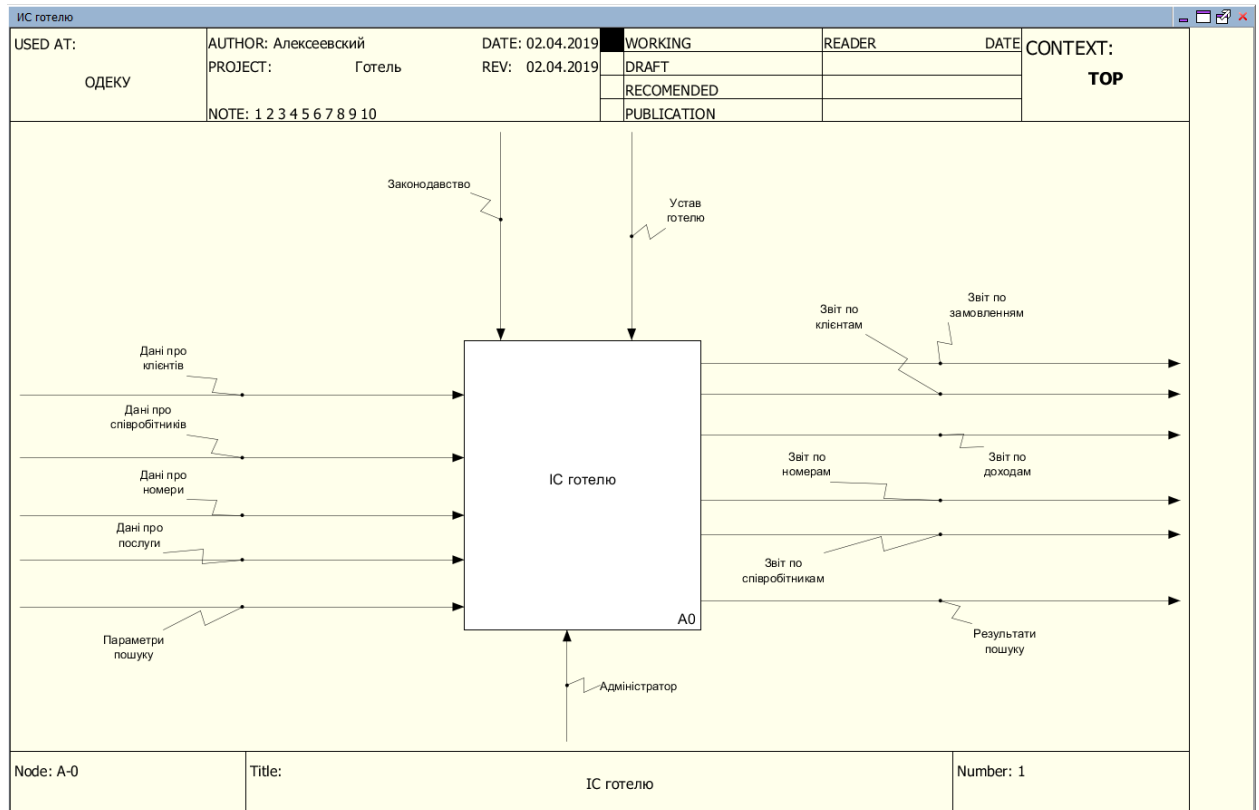


Рисунок 8 – Контекстна функціональна діаграма

Однією з найважливіших особливостей методології SADT є поступове введення рівнів деталізації створених діаграм, коли кожен процес може бути розділений на більш детальні. Така декомпозиція продовжується доки не буде досягнутий такий рівень декомпозиції, на якому процеси стають елементарними і деталізувати їх далі неможливо. Процес розбиття може тривати аж до конкретних процедур. Кожен блок на діаграмі має свій номер. Блок будь-якої діаграми може бути далі описаний діаграмою нижнього рівня, яка, у свою чергу, може бути далі деталізована за допомогою необхідного числа діаграм. Таким чином, формується ієрархія діаграм.

Деталізована функціональна діаграма першого рівня наведена на рис. 9.

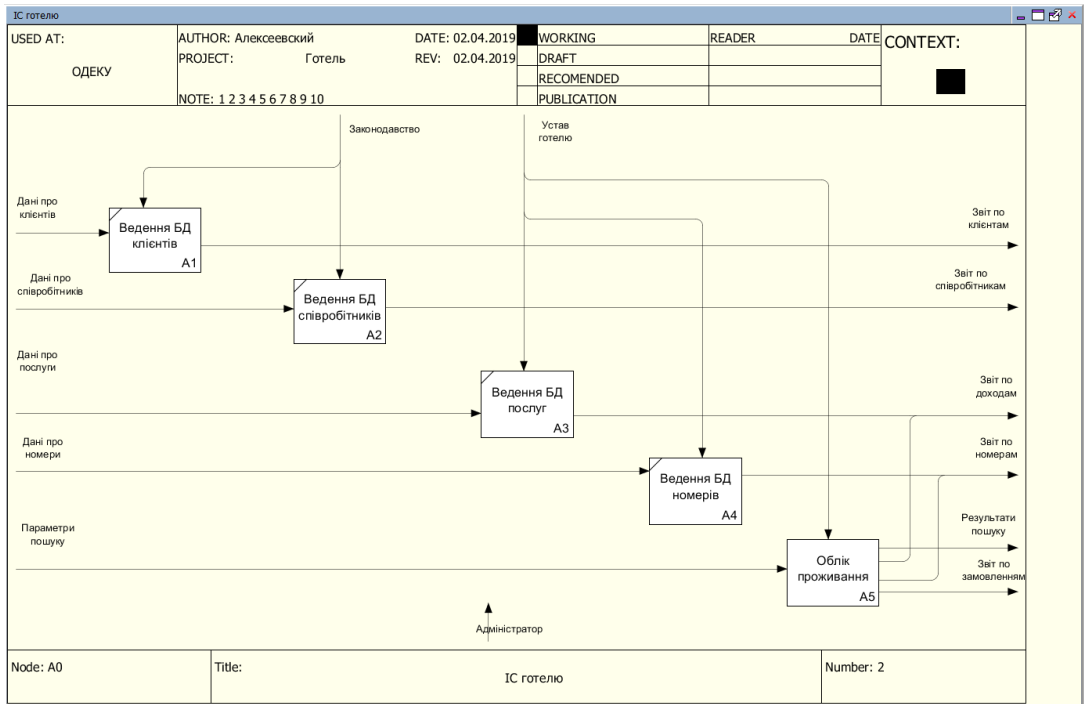


Рисунок 9 – Деталізована функціональна діаграма першого рівня

Деталізована функціональна діаграма процесу Облік номерів наведена на рис. 10.

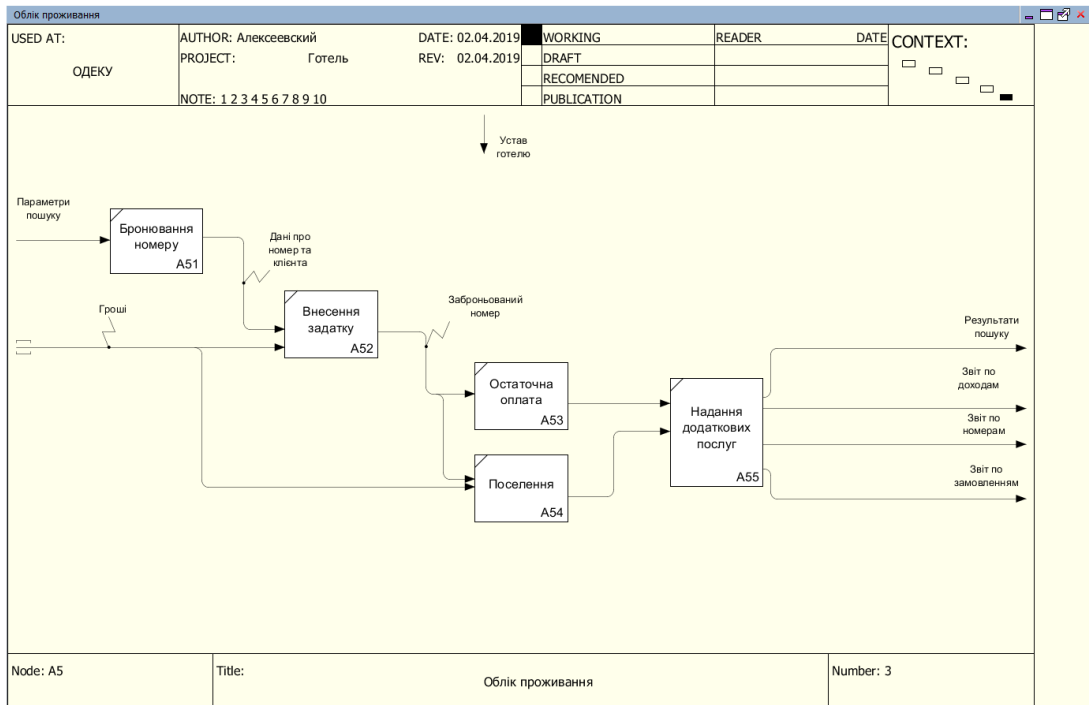


Рисунок 10 – Деталізована функціональна діаграма другого рівня

3.2 Побудова діаграм потоків даних

Діаграми потоків даних (Data Flow Diagramming) є ще одним засобом моделювання функціональних вимог до майбутньої системи. Вимоги представляються у вигляді ієрархії процесів, пов'язаних потоками даних. Діаграми потоків даних показують, як кожен процес перетворює свої вхідні дані у вихідні, і виявляють відношення між цими процесами. Ці діаграми успішно використовуються як доповнення до моделі IDEF0. Подібно IDEF0, DFD представляє систему як мережу пов'язаних робіт.

Основні компоненти DFD роботи (процеси), зовнішні сутності, потоки даних, накопичувачі даних (сховища). Джерела інформації (зовнішня сутність) породжують інформаційні потоки (потоки даних), що переносять інформацію до підсистем або процесів. Ті в свою чергу перетворюють інформацію і породжують нові потоки, які переносять інформацію до інших процесів або підсистем, накопичувачів даних або зовнішньої сутності – споживачам інформації.

У DFD роботи (процеси) представляють собою функції системи, що перетворюють входи у виходи. Процеси у програмі Ramus Educational зображають прямокутником із закругленими кутами.

Зовнішні сутності зображують входи в систему і/або виходи з системи. Зовнішні сутності зображуються у вигляді прямокутника з тінню і зазвичай розташовуються по краях. Одна зовнішня сутність може бути використана багаторазово на одній або декількох діаграмах. Зазвичай такий прийом використовують, щоб не малювати занадто довгих і заплутаних стрілок.

Потоки зображуються стрілками і описують рух об'єктів з однієї частини системи в іншу. Оскільки в DFD кожна сторона процесу не має чіткого призначення, як в IDEF0, стрілки можуть підходити і виходити з будь-якої грані процесу. У DFD також застосовуються двонаправлені стрілки для опису діалогів типу "команда-відповідь" між процесами, між процесом і зовнішньої сутністю і між зовнішніми сутностями.

У системах обробки інформації сховища даних є механізмом, який дозволяє зберегти дані для подальших процесів.

На рис. 11 представлена контекстна діаграма потоків даних, що описує головний процес майбутньої системи – робота готелю. Зовнішньою сутністю є клієнт. Це фізична особа, яка є джерелом або приймачем інформації. Визначення деякого об'єкту або системи як зовнішня сутність указує на те, що вона знаходиться за межами меж аналізованої інформаційної системи. В процесі аналізу деяка зовнішня суть може бути перенесена всередину діаграми аналізованої системи, якщо це необхідно, або, навпаки, частина процесів може бути винесена за межі діаграми і представлена як зовнішня сутність.

Накопичувач даних є абстрактним пристроєм для зберігання інформації, яку можна у будь-який момент помістити в накопичувач і через деякий час витягнути, причому способи приміщення і витягання можуть бути будь-якими. Накопичувачами даних є Клієнти, Співробітники, Номери, Заселення, Послуги.

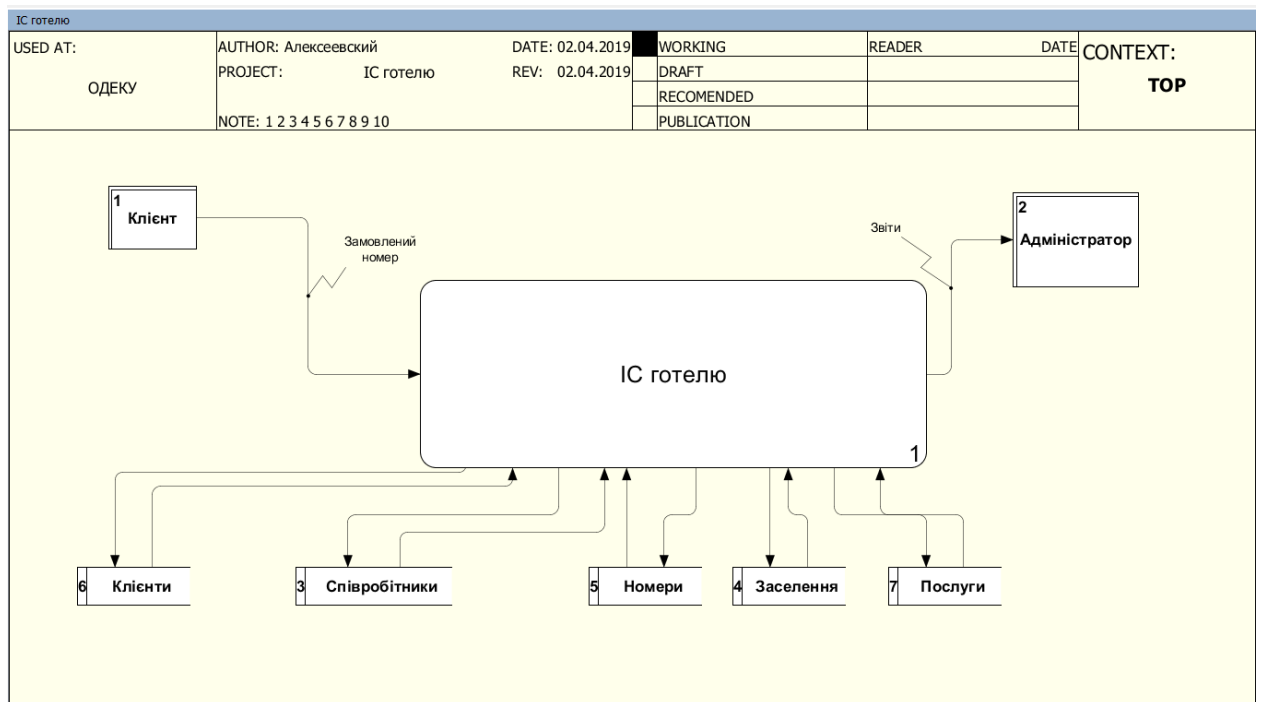


Рисунок 11 – Контекстна діаграма потоків даних

Діаграми потоків даних також піддаються декомпозиції. У ході деталізації були виявлені наступні процеси системи: реєстрація клієнтів, пошук необхідного номеру, бронювання номеру, оформлення заселення, надання додаткових послуг. На рис. 12 показана декомпована діаграма потоків даних.

Після побудови закінченої моделі системи її необхідно верифікувати, тобто перевірити на повноту і узгодженість. У повній моделі всі її об'єкти (підсистеми, процеси, потоки даних) повинні бути детально описані і деталізовані. Виявлені недеталізовані об'єкти слід деталізувати, повернувшись на попередні кроки розробки. У узгодженій моделі для всіх потоків даних і накопичувачів даних повинне виконуватися правило збереження інформації: всі дані, що поступають куди-небудь, повинні бути прочитані, а всі зчитувані дані повинні бути записані.

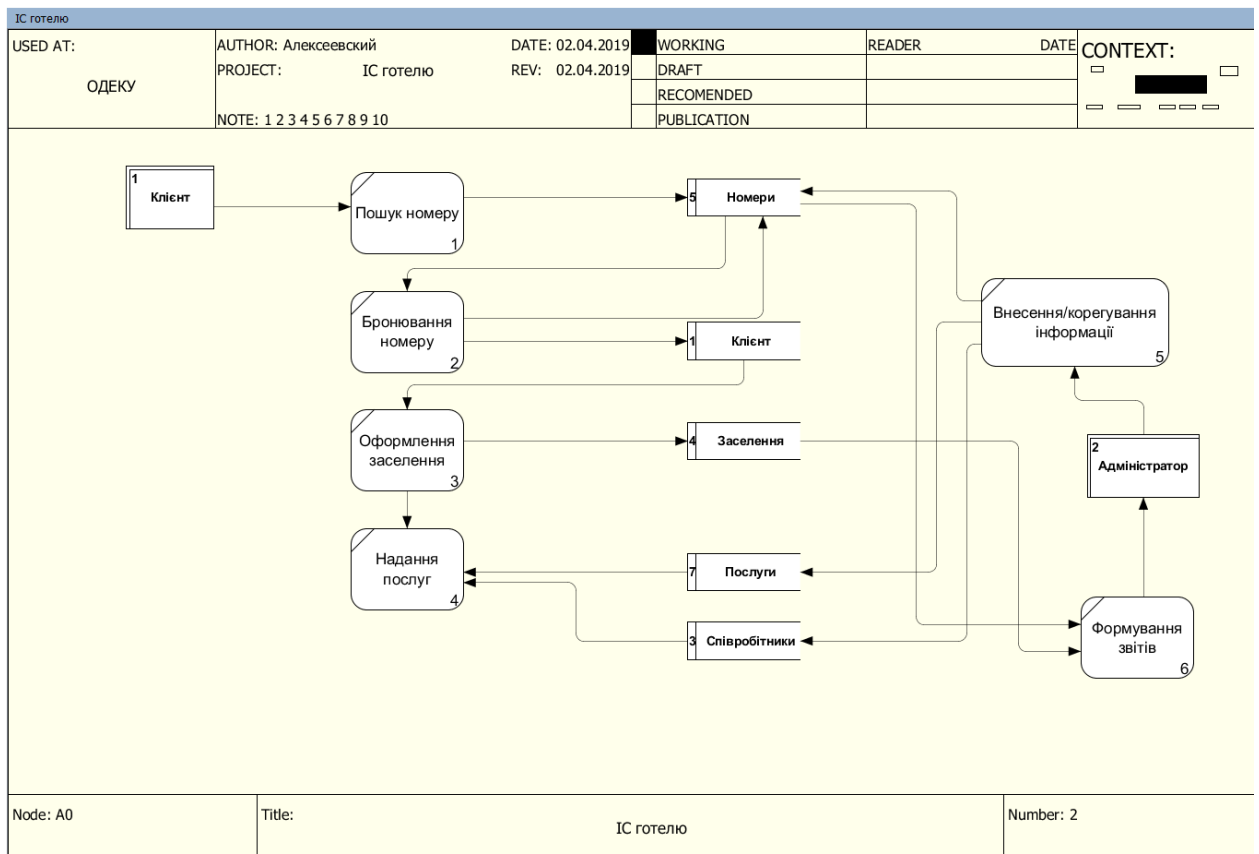


Рисунок 12 – Деталізована діаграма потоків даних

3.3 Проектування та створення бази даних

Архітектура інформаційної системи підтримки діяльності готелю реалізована архітектурою «клієнт-сервер» передбачає, що база даних розміщена на сервері комп'ютерної мережі. На робочих станціях розміщують клієнтську програму, що формує і відсилає запити віддаленого сервера з БД, який забезпечує виконання запиту і видачу клієнтові результату. Вся обробка інформації відбувається на віддаленому сервері.

Основним компонентом реляційних БД є таблиця. Таблиця використовується для структуризації та зберігання інформації. Сутність – це суб'єкт, місце, річ, подія або поняття, що містять інформацію. Точніше, сутність – це набір (об'єднання) об'єктів, який називається екземплярами. Кожен екземпляр індивідуальний і повинен відрізнятися від всіх інших екземплярів. Кожен екземпляр сутності має набір характеристик. У логічній моделі БД всі ці характеристики називаються атрибутами сутності.

Логічна (концептуальна) модель – це загальний погляд на дані, модель даних в певній предметній області. Інколи її називають концептуальною моделлю даних. Такою моделлю і є модель типу сутність-зв'язок.

Фізична модель – це модель даних, реалізована у вигляді опису бази даних. Вона може бути реалізована на двох рівнях, а саме у вигляді трансформаційної моделі і моделі в системі керування базами даних. Для створення фізичної моделі необхідно вибрати тип сервера, для якого будуть генеруватись коди (створюватись програма).

Кожна сутність містить первинний ключ призначений для унікальної ідентифікації примірника сутності. При створенні сутності необхідно виділити групу атрибутів, які потенційно можуть стати первинним ключем, потім провести відбір атрибутів для включення до складу первинного ключа.

Для бази даних інформаційної системи міні-готелю визначимо такі сутності: Постояльці, Номери, Категорії номерів, Послуги, Замовлення, Види харчування, Користувачі системи, Оплата. Також визначимо довідники: Фірми,

Посади, Місця роботи, Статуси замовлень, Країни, Регіони, Міста, Види оплат, Ролі, Користувачі.

Таблиці відповідають умовам третьої нормальної форми. Це необхідно для усунення надлишковості даних, аномалій додавання, редагування та видалення даних з таблиць, а також транзитивних залежностей між неключовими атрибутами. Імена таблиць і полів задані латиницею, що є зручним при подальшому зверненні до таблиць і їх полів при програмуванні і створенні запитів на мові SQL. У кожній таблиці визначений первинний ключ, для зв'язку таблиць і забезпечення цілісності даних створені зовнішні ключі, також визначені обов'язкові поля та властивості полів. Кожна таблиця включає до свого складу поле коду, що використовується зазвичай як лічильник головного її параметра і, як правило, що є ключовим полем.

Структура таблиць-довідників однакова, наведена у табл.1.

Таблиця 1 – Опис полів таблиць-довідників

Ім'я поля	Короткий опис поля
Id	Код (первинний ключ)
Name	Найменування

Структура таблиці Номери наведена у табл.2. Таблиця містить інформацію про номери міні-готелю.

Таблиця 2 – Опис полів таблиці Номери

Ім'я поля	Короткий опис поля
Id	Код номеру (первинний ключ)
Name	Найменування
Kategory	Категорія номеру
Cina	Ціна за добу

Структура таблиці Постояльці наведена у табл.3. Таблиця містить інформацію про клієнтів готелю. Клієнтами є особи, які проживають в готелі або зареєструвались на сайті і мають можливість забронювати номер.

Таблиця 3 – Опис полів таблиці Постояльці

Ім'я поля	Короткий опис поля
Id	Код (первинний ключ)
F	Прізвище
I	Ім'я
O	По батькові
pol	Стать
Id_mesto_raboti	Де працює (зовнішній ключ)
passport	Документ
Seriya	Серія
nomer	Номер
Kem_vidan	Ким виданий
Koli_vidan	Коли виданий
Data_rozd	Дата народження
Telefon	Номер телефону
E_mail	Електронна адреса
Id_region	Регіон (зовнішній ключ)
Id_kraina	Країна (зовнішній ключ)
Id_misto	Місто (зовнішній ключ)

Таблиця Замовлення містить інформацію про замовлення на бронювання, мешкання в готелю зареєстрованого клієнта. Зареєструватись та зробити замовлення заздалегідь можна як на сайті готелю, так безпосередньо особисто відвідавши готель.

Структура таблиці наведена у табл.4.

Таблиця 4 – Опис полів таблиці Замовлення

Ім'я поля	Короткий опис поля
Id	Код (первинний ключ)
Id_klient	Клієнт (зовнішній ключ)
Id_nomer	Номер (зовнішній ключ)
Data_zaezda	Дата заїзду
Data_viezda	Дата виїзду
Id_pitania	Харчування (зовнішній ключ)
Id_uslugi	Послуги (зовнішній ключ)
Id_naznachenia	Призначення (зовнішній ключ)
Id_statusa	Статус (зовнішній ключ)
Id_statti	Стаття (зовнішній ключ)
info	Додаткова інформація

Таблиця Оплата містить інформацію про оплати, зроблені клієнтами за мешкання та послуги готелю. Опис полів наведено у табл.5.

Таблиця 5 – Опис полів таблиці Оплата

Ім'я поля	Короткий опис поля
Id	Код (первинний ключ)
Id_zamovlenna	Замовлення (зовнішній ключ)
Id_tip_oplat	Тип оплати (зовнішній ключ)
Id_vid_oplat	Вид оплати (зовнішній ключ)
Suma	Сума

Створення бази даних та її таблиць у СУБД Microsoft SQL Server може проходити двома шляхам – за допомогою конструктора або згенерованого SQL-файлу. Створити нову базу даних можна шляхом вибору в меню пункту New Database. Таблиці бази даних та довідники створені на сервері в режимі

конструктора таблиць. Перелік створених таблиць бази даних готелю наведений на рис. 13.

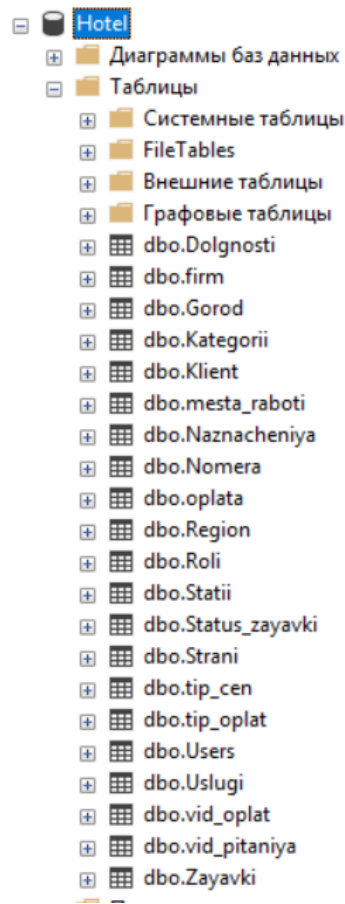


Рисунок 13 – Таблиці бази даних на сервері

Визначивши сутності бази даних, можна переходити до визначення залежностей між ними. Зв'язок – це деяка асоціація між двома сутностями. Одна сутність може бути пов'язана з іншою сутністю чи сама з собою. Зв'язки дозволяють по одній сутності знаходити інші сутності, пов'язані з нею. Кожен зв'язок має два кінці і одне або два найменування. Кожен зв'язок може мати один з таких типів зв'язку:

- один-до-одного – 1:1;
- один-до-багатьох – 1: N;
- багато-до-багатьох – M:N.

Зв'язок типу один-до-одного означає, що один примірник першої сутності (лівої) пов'язаний з одним примірником другий сутності (правої).

Зв'язок типу один-до-багатьох означає, що один примірник першої сутності (лівої) пов'язаний з декількома екземплярами другий сутності (правої). Це найбільш часто використовуваний тип зв'язку.

Зв'язок типу багато-до-багатьох повинен бути замінений двома зв'язками типу один-до-багатьох шляхом створення проміжної сутності.

Схема створеної бази даних наведена на рис. 14. Між таблицями встановлені зв'язки типу один-до-багатьох.

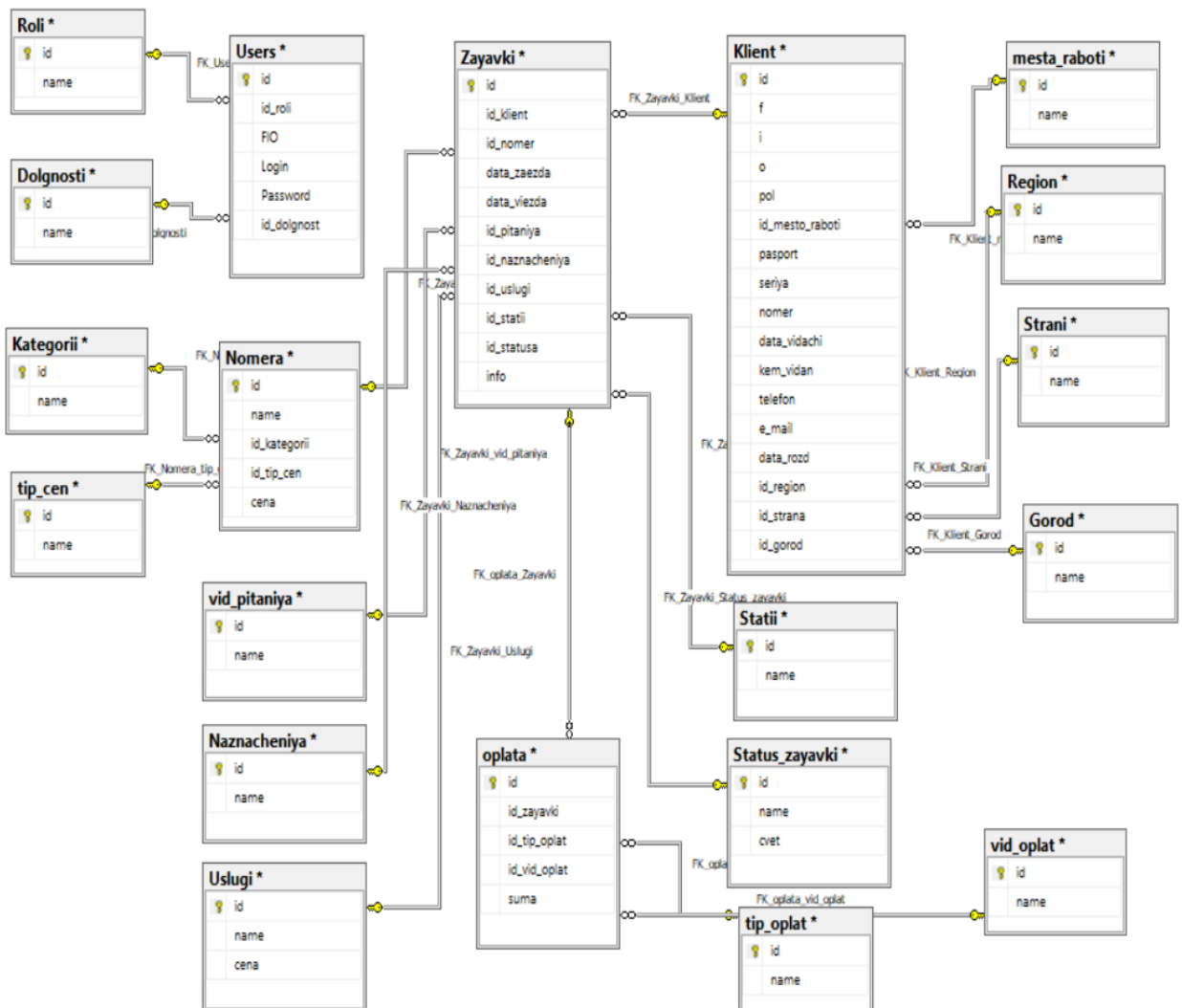


Рисунок 14 – Схема бази даних

3.4 Проектування інтерфейсу сайту

Навігація інтернет-системи – один з найважливіших етапів при проектуванні системи. Кожен відвідувач, переглядаючи інтернет-ресурс повинен, так чи інакше, до нього пристосовуватися. Це пов'язано з безліччю технічних рішень і дизайнерських можливостей при проектуванні системи. Тим більше, що в розпорядженні будь-якого користувача по статистиці всього 7-12 секунд, після чого якщо йому веб-сайт не сподобається, він просто піде з нього, закривши сторінку браузера. Тому навігація повинна бути простою і зрозумілою для користувача. Все те, що створено на сайті, а саме шапка сайту, нижня його частина, бічні панелі, навігаційна панель, повинні бути присутніми в незмінному вигляді. Щоб не лякати відвідувачів і змушувати їх шукати тий чи інший блок при переході від сторінки до сторінки. Навігація повинна бути легко знайдена поглядом і завжди бути присутнім в одному місці, мало того вона повинна бути зручна для користувачів. Розмір посилань повинен бути таким, щоб користувач міг легко натиснути на них, підвівши курсор маніпулятора.

Пристаюючи до розробки інформаційної системи міні-готелю, частиною якої є Інтернет-сайт, необхідно чітко уявляти структуру навігації та інформаційного наповнення. Це дозволить визначити структуру інтернет-додатку і використовувати технології.

З головної сторінки сайту користувач переходить на сторінки, в яких надається конкретна інформація по цікавлячим користувача аспектам роботи інформаційної системи. Доступ до засобів навігації відбувається з будь-якої сторінки інтернет-додатка.

На головній сторінці користувач має можливість ознайомитися зі структурою веб-ресурсу, використовувати пошук та прочитати корисні статті.

Схема користувача інформаційної системи міні-готелю надасть відвідувачу можливість зрозуміти структуру веб-сайту. Вона полегшить подальшу навігацію по ресурсу. Схема представлена на рис.15.

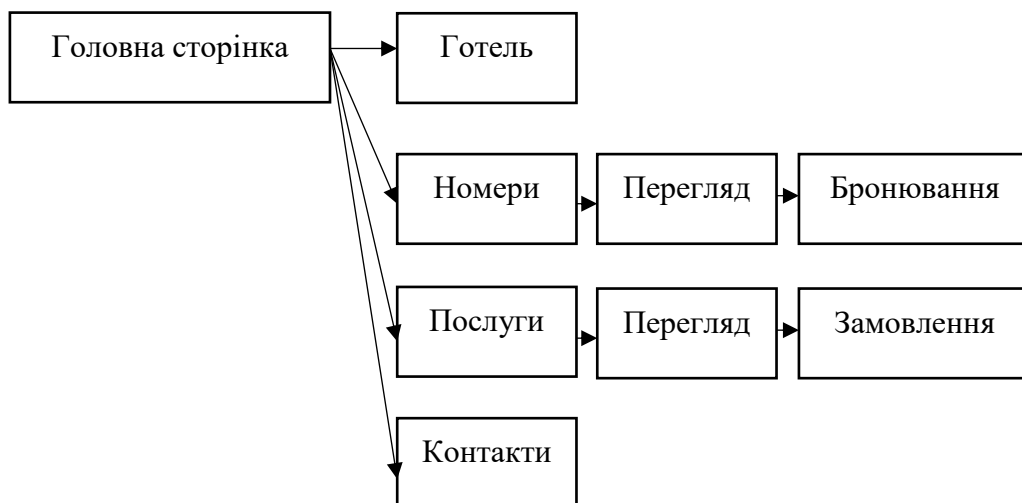


Рисунок 15 – Схема користувача сайту інформаційної системи

Адміністрування включає дві функції: редагування номерного фонду та редагування списку замовлень.

Редагувати довідники зручно в табличному вигляді. Для цього створюється спеціальна сторінка. Редагувати зручно в тому вигляді, в якому його побачить користувач. Навігаційна карта для адміністратора сайту представлена на рис. 16.



Рисунок 16 – Схема адміністраторської частини системи

Вимоги до графічного дизайну сайту:

- дизайн повинен бути лаконічним і в той же час виглядати стильно, сучасно;
- використання невеликих, але стильних графічних елементів в оформленні контенту (вмісту) додатка;
- графічна оболонка внутрішніх сторінок зазвичай є спільною для всіх підрозділів. Вона повинна бути стилістично витримана з початковою сторінкою сайту, вона не повинна відволікати від інформаційного наповнення сторінки (контенту);
- дизайн повинен допомагати користувачеві сприймати інформацію, а не заважати і відволікати його.

Щоб витримати всі сторінки інформаційної системи у єдиному стилі, необхідно розробити шаблон сторінки. Завдяки єдиному шаблону всі сторінки верстаються за подобою однієї сторінки, що забезпечує їх однорідність. Структура шаблону (рис.17), складається з елементів, які повинні бути присутніми на всіх сторінках інтернет-ресурсу.

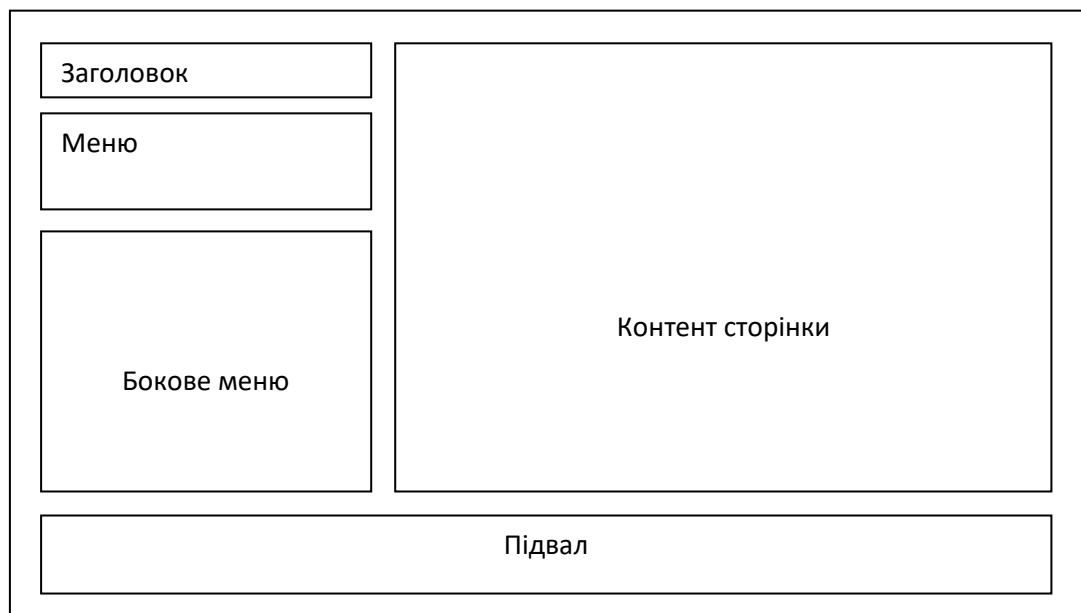


Рисунок 17 – Структура сторінок інтерфейсу сайту

4 ОПИС ІНТЕРФЕЙСУ КОРИСТУВАЧІВ ІНФОРМАЦІЙНОЇ СИСТЕМИ

4.1 Опис інтерфейсу менеджера готелю

Після запуску застосування з'являється форма для введення логіну і паролю користувача (рис.18).

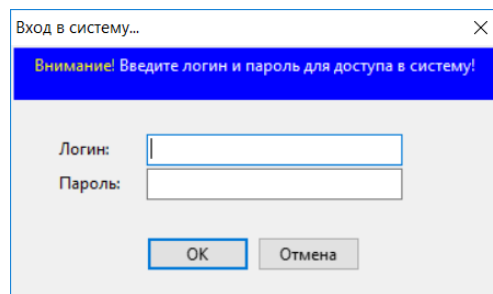


Рисунок 18 – Форма входу в систему

Після правильного введення необхідних даних, менеджеру доступні основні функції, наведені на панелі (рис.19)

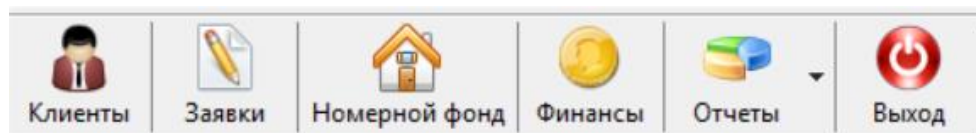


Рисунок 19 – Панель функцій менеджера міні-готелю

Розглянемо послідовно порядок роботи з системою. Після натискання кнопки з піктограмою Клієнти відкриється форма, наведена на рис. 20. Клієнтами готелю можуть бути як звичайні люди (фізичні особи), так і фірми (юридичні особи), які замовляють послуги для свого персоналу. Для роботи з клієнтами доступні функції додавання, видалення, редагування, фільтрації, експорту, оновлення. Також, є можливість створити нову групу клієнтів, або внести зміни в існуючу.

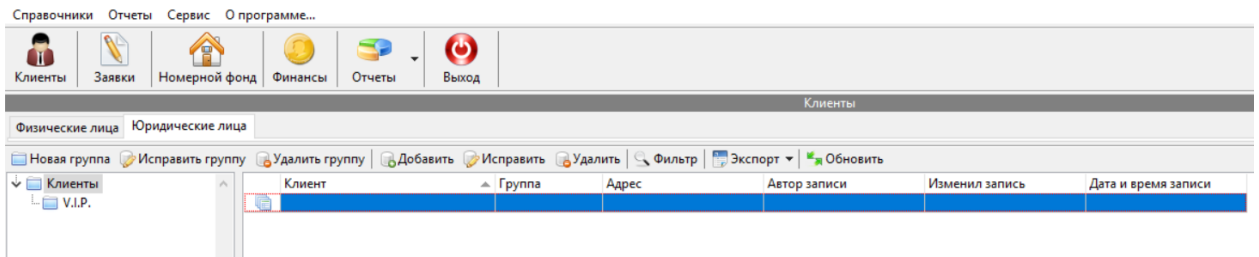


Рисунок 20 – Форма для роботи з клієнтами

Для додавання інформації про нового клієнта(постояльца, гостя) готелю необхідно натиснути кнопку Додати. Після чого відкриється форма, наведена на рис. 21.

Рисунок 21 – Форма для введення інформації про клієнта

Після підтвердження вводу інформація стає видимою на формі (рис.22).

Клиент	Группа	Место работы	Должность	Документ	Серия	Номер	Адрес	Автор записи	Изменил запись
Андреев Артур	Клиенты			паспорт				Администратор	Администратор
Вохменцева Татьяна Борисовна	Клиенты	ООО "ФФФ"	Руководитель	паспорт	1111	111111	Адрес Иванова	Администратор	Администратор
Иванов Иван Иванович	V.I.P.	ООО Фирма	Директор	паспорт	2222	222222	Адрес Петрова	Администратор	Администратор
Петров Петр Петрович	V.I.P.	ООО Фирма	Директор	паспорт	2222	222222	Адрес Петрова	Администратор	Администратор
Петрова Мария Ивановна	Клиенты	ООО "РРРРР"	Бухгалтер	паспорт	333	555555	Ярославль, ул. Тургенева	Администратор	Администратор

Рисунок 22 – Список клієнтів

Для зареєстрованого клієнта можна створити нове замовлення на номери та послуги готелю. Для цього потрібно натиснути кнопку з піктограмою Замовлення. У формі (рис.23) обираємо клієнта, необхідний номер та інші дані, вводимо дати, необхідну додаткову інформацію (за бажанням). В цій же формі є можливість ввести дані про гостей, роздрукувати замовлення або експортувати його.

Заявка №: 5 от: 08.04.2019 15 Фирма: ООО "Моя фирма"

Клиент*: Вохменцева Татьяна Борисов Заезд: 08.04.2019 15 часы: 14 минуты: 00
 Пол: Ж Паспорт: № Выезд: 09.04.2019 15 часы: 12 минуты: 30
 Телефон: 749-83-03 Кол-во: 1 дней Инфо:
 Номер*: Номер 5 Цена: 2221.00
 Тип цен*: Стандарт Сумма: 2221.00
 Вид оплаты*: По договору Услуги: 0.00
 Питание*: Комплекс Скидка: 0 %
 Назначение*: Назначение 1 Итого: 2221.00
 Статус: Резерв НДС: Без НДС

Гости Услуги Оплаты

Добавить Исправить Удалить Печать

№	Ф.И.О.	Пол	Паспорт

Печать Экспорт ОК Отмена

Рисунок 23 – Форма введення нового замовлення

Для перегляду номерного фонду після натискання відповідної кнопки відкриється форма (рис.24), на якій у вигляді таблиці-шахматки буде відображена інформація про стан номерів. Можна задати початкову дату, кількість днів для перегляду. Відображається інформація про номери, постояльців та

статус номерів. Для зручності різні статуси (резерв, бронь, зайнятий) номерів позначаються різним кольором. Колір можна обрати при створенні довідників.

Номер	Цена, грн	08.04	09.04	10.04	11.04	12.04	13.04	14.04	15.04	16.04	17.04	18.04	19.04	20.04	21.04	22.04	23.04	24.04	25.04	26.04	27.04	28.04	29.04	30.04
Номер 1 VIP	444,00	Иванов Иван Иванович																						
Номер 3 VIP	550,00																							
Номер 4 VIP	3 000,00																							
Номер 5 VIP	2 221,00	Вохм																						
Номер 6 VIP	300,00																							
Люкс VIP	1 700,00	Андреев Артур																						
Номер 7 ДвМ	500,00																							
Номер 8 ДвМ	1,00																							
Номер 9 ДвМ	77,00																							
Номер 10 ДвМ	333,00																							
Номер 11 ДвМ	400,00																							
Номер 12 ДвМ	500,00																							
Номер 2 ДвМ	230,00																							
Номер 13 Сем	200,00																							
Номер 14 Сем	11,00																							
Супер люкс Сем	500,00																							

Рисунок 24 – Форма для роботи з номерним фондом

У цій формі є можливість коригування інформації, пошуку, оновлення, а також, експорту даних у інші програми для подальшої обробки та проведення розрахунків. На рис. 25 показано вибір експорту у програму MS Excel.

Номер	Цена, грн	08.04	09.04	10.04
Номер 1 VIP	444,00	Иванов Иван Иванович		
Номер 3 VIP	550,00			
Номер 4 VIP	3 000,00			
Номер 5 VIP	2 221,00	Вохм		
Номер 6 VIP	300,00			
Люкс VIP	1 700,00	Андреев Артур		
Номер 7 ДвМ	500,00			

Рисунок 25 – Експорт даних

На рис. 26 показана таблиця в середовищі обраного MS Excel.

Номер	Цена, Руб	08.04.2019	09.04.2019	10.04.2019	11.04.2019	12.04.2019	13.04.2019	14.04.2019	15.04.2019
Номер 1 VIP	444.00		Иванов Иван Иванович	Иванов Иван Иванович	Иванов Иван Иванович	Иванов Иван Иванович	Иванов Иван Иванович	Иванов Иван Иванович	Иванов Иван Иванович
Номер 3 VIP	550.00								
Номер 4 VIP	3 000.00								
Номер 5 VIP	2 221.00								
Номер 6 VIP	300.00								
Люкс VIP	1 700.00		Андреев Артур	Андреев Артур	Андреев Артур	Андреев Артур			
Номер 7 ДеМ	500.00								

Рисунок 26 – Таблица номерного фонду в MS Excel

Інформацію про грошові надходження можна переглянути, натиснувши на відповідну кнопку Фінанси на панелі програми. У формі (рис.27) є можливість перегляду надходжень за обрану дату, від конкретного клієнта, за обраною статтею та видом оплати.

Приход денежных средств

Дата: 08.04.2019

Клиент: Вохменцева Татьяна Борисовна

Статья: ! оплата клиента

Вид оплаты: Нал

Сумма: 2000.00

Описание:

Доп. информация:

OK Отмена

Рисунок 27 – Форма перегляду грошових надходжень

Можно зробити вибірку, задавши інтервал дат (рис.28).

Выборка

Интервал с: 08.04.2019 15 по: 30.04.2019 15

Контрагент: Андреев Артур

Статья: ! оплата клиента

Вид оплаты: Безнал

OK Отмена

Рисунок 28 – Форма задания параметров выборки

Результат наведено на рис. 29.

Итого по выборке

Вид оплаты	Сумма, грн
Безнал	37 666.19
Нал	2 000.00
ИТОГО:	39 666.19

OK

Рисунок 29 – Результаты грошових надходжень за обраний інтервал дат

В інформаційній системі є можливість створення різних звітів (рис.30).

Справочники Отчеты Сервис О программе...

Клиенты Заявки Номерной фонд Финансы Отчеты Выход

Приход Расход Исправить Удалить

Вид оплаты	Дата	Приход грн	Рас
Безнал	11.07.2017	4 351.19	
Безнал	11.07.2017	33 315.00	
Нал	08.04.2019	2 000.00	

- График заезда
- График выезда
- Отчет по поступившим гостям
- Отчет по заехавшим гостям
- Отчет по питанию
- Отчет по дате рождения
- Отчет о проживании гостя
- Отчет по назначениям
- Отчет по назначениям по оплатам
- Отчет взаиморасчеты
- Отчет по количеству гостей
- Движение денежных средств

Обновить

Клиент

Иванов Иван Иванович

Петров Петр Петрович

Вохменцева Татьяна Борисовна

Рисунок 30 – Вікно обрання звіту

Приклад звіту-графіку заїзду гостей на обраний період наведено на рис. 31.

График заезда гостей с 01.04.2019 по 08.04.2019

№ п/п	Комната	Ф.И.О. гостя	Кол-во дней	Заезд	Выезд	Вид оплаты
Заезд 01.04.2019						
1	Номер 13	Иванов Иван Иванович	1	01.04.2019	02.04.2019	По договору
2	Номер 13	Андреев Артур	7	01.04.2019	07.04.2019	По договору
Заезд 08.04.2019						
3	Номер 1	Иванов Иван Иванович	7	08.04.2019	14.04.2019	По договору
4	Люкс	Андреев Артур	5	08.04.2019	12.04.2019	По договору
5	Номер 5	Вохменцева Татьяна Борисовна	1	08.04.2019	09.04.2019	По договору

Директор _____ Администратор

Рисунок 31 – Приклад звіту про графік заїзду

Приклад звіту по гостям за обраний період наведено на рис. 32.

Отчет по заехавшим гостям с 01.04.2019 по 08.04.2019

№ п/п	Пос.	Вып.	Ф.И.О. гостя	Д.р.	Адрес проживания	Место работы	Пасп.	Ком.	Дн.	Сумма	Питание	Заметки
1	01.04	07.04	Андреев Артур	0			№	Номер 13	7	1 400,00	Заказное	
2	01.04	02.04	Иванов Иван Иванович	07.02.1973	Адрес Иванова	ООО "ФФФ"	1111 № 111111	Номер 13	1	200,00	Комплекс	
3	08.04	12.04	Андреев Артур	0			№	Люкс	5	8 500,00	Заказное	
4	08.04	09.04	Вохменцева Татьяна Борисовна	0			№	Номер 5	1	2 221,00	Комплекс	
5	08.04	14.04	Иванов Иван Иванович	07.02.1973	Адрес Иванова	ООО "ФФФ"	1111 № 111111	Номер 1	7	3 108,00	Комплекс	

Директор _____ Администратор

Рисунок 32 – Приклад звіту по гостям за обраний період

В інформаційній системі передбачена можливість створення звіту про перебування обраного гостя в готелі. Для створення звіту спочатку потрібно обрати прізвище (або назву) клієнта в формі (рис.33). Приклад звіту наведено на рис. 34.

Рисунок 33 – Вікно вибору клієнта для створення звіту

Информация о проживании гостя

Ф.И.О. гостя: **Вохменцева Татьяна Борисовна**
 Паспорт: **Серия №**
 Дата рождения: **749-83-03** Возраст:
 Телефон:
 Адрес:

№ п/п	Дата заезда	Дата выезда	Ком-ната	Примечание
1	08.04.2019	09.04.2019	Номер 5	

Директор _____ Администратор

Рисунок 34 – Приклад звіту про мешкання гостя

Для створення звіту про кількість гостей, які перебували в готелі за обраний період треба спочатку задати місяць, рік у формі, яка наведена на рис. 35.

Рисунок 35 – Вікно вибору періоду для звіту про кількість гостей

Ежедневный отчет по количеству гостей

	<i>проживает:</i>	<i>забронировано:</i>	<i>выезжает:</i>	<i>итого:</i>
<i>01.04.2019</i>	<i>0</i>	<i>2</i>	<i>0</i>	<i>2</i>
<i>02.04.2019</i>	<i>0</i>	<i>2</i>	<i>1</i>	<i>3</i>
<i>03.04.2019</i>	<i>0</i>	<i>1</i>	<i>0</i>	<i>1</i>
<i>04.04.2019</i>	<i>0</i>	<i>1</i>	<i>0</i>	<i>1</i>
<i>05.04.2019</i>	<i>0</i>	<i>1</i>	<i>0</i>	<i>1</i>
<i>06.04.2019</i>	<i>0</i>	<i>1</i>	<i>0</i>	<i>1</i>
<i>07.04.2019</i>	<i>0</i>	<i>1</i>	<i>1</i>	<i>2</i>
<i>08.04.2019</i>	<i>0</i>	<i>3</i>	<i>0</i>	<i>3</i>
<i>09.04.2019</i>	<i>0</i>	<i>3</i>	<i>1</i>	<i>4</i>
<i>10.04.2019</i>	<i>0</i>	<i>2</i>	<i>0</i>	<i>2</i>
<i>11.04.2019</i>	<i>0</i>	<i>2</i>	<i>0</i>	<i>2</i>
<i>12.04.2019</i>	<i>0</i>	<i>2</i>	<i>1</i>	<i>3</i>
<i>13.04.2019</i>	<i>0</i>	<i>1</i>	<i>0</i>	<i>1</i>
<i>14.04.2019</i>	<i>0</i>	<i>1</i>	<i>1</i>	<i>2</i>

Рисунок 36 – Приклад звіту про щоденне перебування гостей

Головне меню застосування інформаційної системи міні-готелю для користувачів-адміністраторів системи містить наступні пункти: Довідники, Звіти, Сервіс, Про програму. Розглянемо їх детальніше.

На рис. 37 наведено повний перелік довідників системи. Довідники заповнені та збережені в базі даних, але, за необхідності адміністратор зможе додати нові записи в таблиці, відредагувати вже існуючі, видалити неактуальні дані. Для маніпулювання цими даними необхідно мати відповідні права доступу.

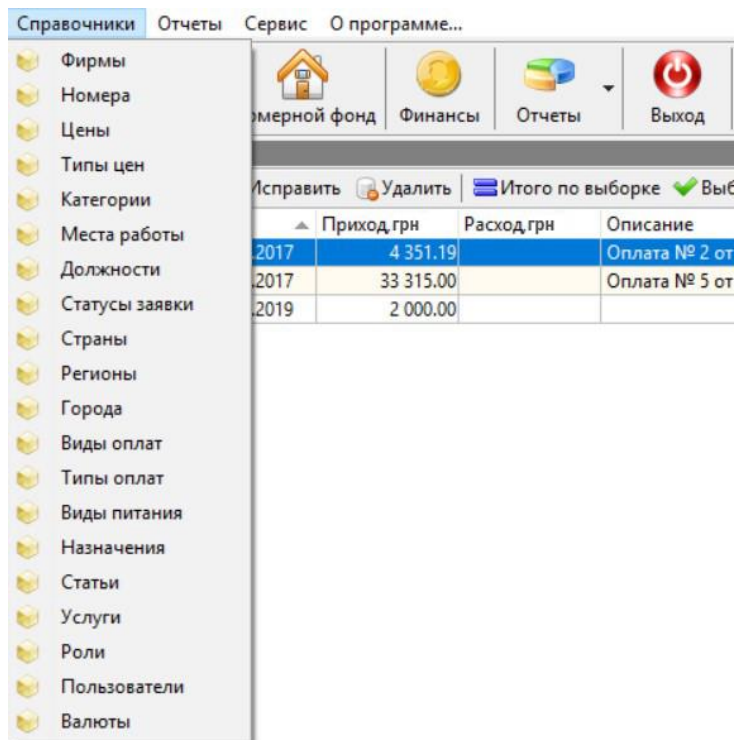


Рисунок 37 – Перелік довідників системи

Для кожного довідника адміністратор системи має можливість додавання, видалення, редагування інформації. Наприклад, на рис. 38 наведено довідник номерів готелю.

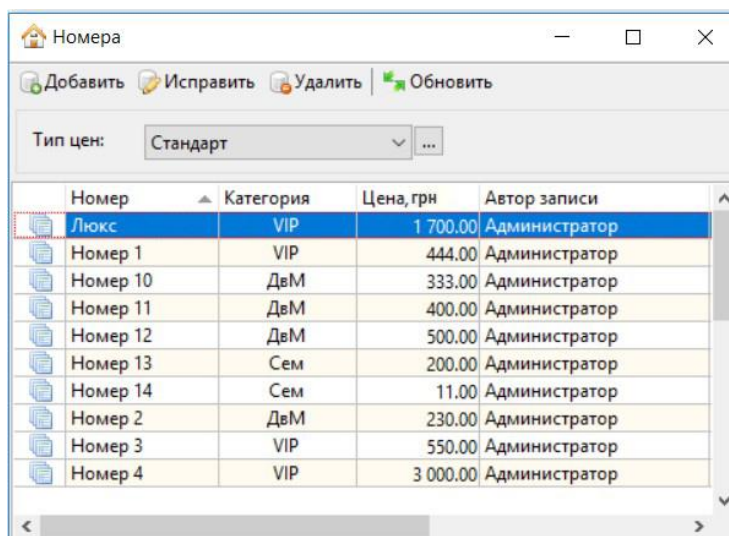


Рисунок 38 – Довідник номерів готелю

Пункт меню Звіти наведений на рис. 39. Перелік звітів дублює звіти, доступні при натисканні на кнопку Звіти.

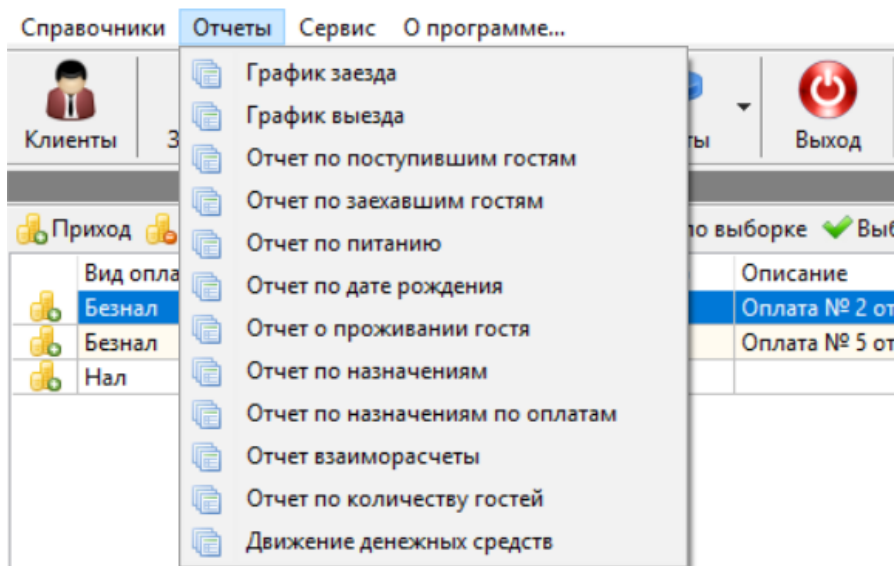


Рисунок 39 – Пункт меню Звіти

На рис. 40 наведений список функцій пункту меню Сервіс. У цьому меню доступні функції адміністратора системи для реєстрації нових користувачів системи, надання їм прав доступу, для створення резервних копій бази даних, відновлення даних.

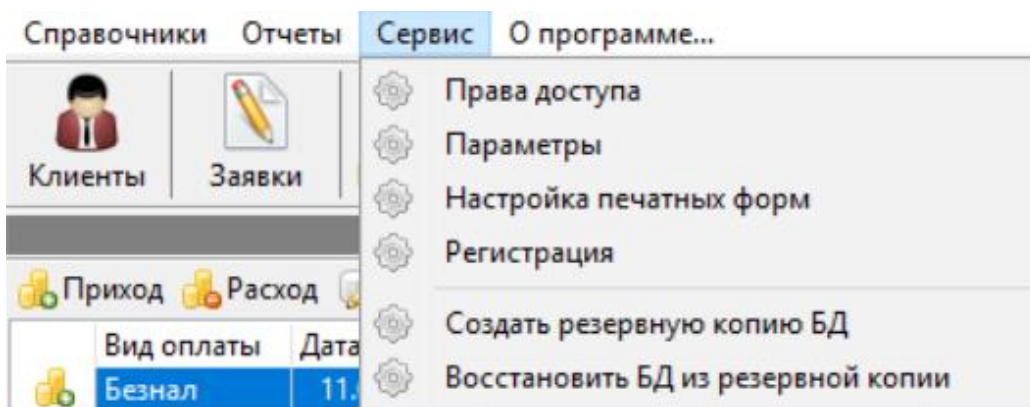


Рисунок 40 – Меню Сервіс

4.2 Опис інтерфейсу клієнта готелю

Для звичайного користувача системи – потенційного постояльця міні-готелю необхідно створити сайт та розмістити його в мережі Інтернет. За допомогою цього ресурсу відвідувач буде мати можливість ознайомитись з послугами готелю, продивитись його цінову політику, забронювати номер. Головна сторінка сайту представлена на рис. 41.



Рисунок 41 – Головна сторінка сайту

Основною метою користувача є ознайомлення з номерами, цінами на них та можливість бронювання вільних номерів, перегляд цікавої інформації та фотогалереї. Наведена на рис.42 форма дозволяє користувачеві обрати бажані дати заїзду та виїзду, вказати скільки дорослих та дітей буде розміщатися в номері.

Готель "Лагуна" + 067 123-45-67
info@otel-diplom.com
Одесса, ул. Такая-то, 23

Головна Готель Номера Послуги Контакти

Бронирование номеров

С помощью приведенной ниже формы вы можете забронировать наши номера в режиме онлайн и получить гарантированную бронь.

Выберите даты заезда, выезда и количество гостей

Дата заезда: 17 апреля 2019
Дата выезда: 25 апреля 2019

Размещение в номере: 1 взрослый

НАЙТИ НОМЕР

Рисунок 42 – Форма для бронювання номеру

Після натискання кнопки «Знайти номер» відкриється форма, наведена на рис.43.

Готель "Лагуна" + 067 123-45-67
info@otel-diplom.com
Одесса, ул. Такая-то, 23

Головна Готель Номера Послуги Контакти

Бронирование номеров

С помощью приведенной ниже формы вы можете забронировать наши номера в режиме онлайн и получить гарантированную бронь.

17 апреля — 25 апреля 1 номер, 1 гость **Изменить даты**

Одноместный номер "Эконом"
Комфортный 1-местный 1-комнатный номер

Основной тариф без завтрака Питание не включено	528,00 ГРН	Забронировать
Основной тариф с завтраком Завтрак включен	688,00 ГРН	Забронировать

Одноместный номер
Комфортный 1-местный 1-комнатный номер

Основной тариф без завтрака Питание не включено	496,00 ГРН	Забронировать
-----------------------------------------------------------	------------	----------------------

Рисунок 43 – Пропоновані номери

На цій сторінці користувачу буде виведено список вільних на ці дати номерів. Якщо користувача задовольняють умови проживання, ціни, то він натискає кнопку «Забронювати». Інформація про бронювання відображається в базі даних та на формі-шахматці менеджера готелю.

Якщо користувач бажає змінити дати, то після натискання кнопки «Змінити дати» відкриється сторінка, наведена на рис. 44.

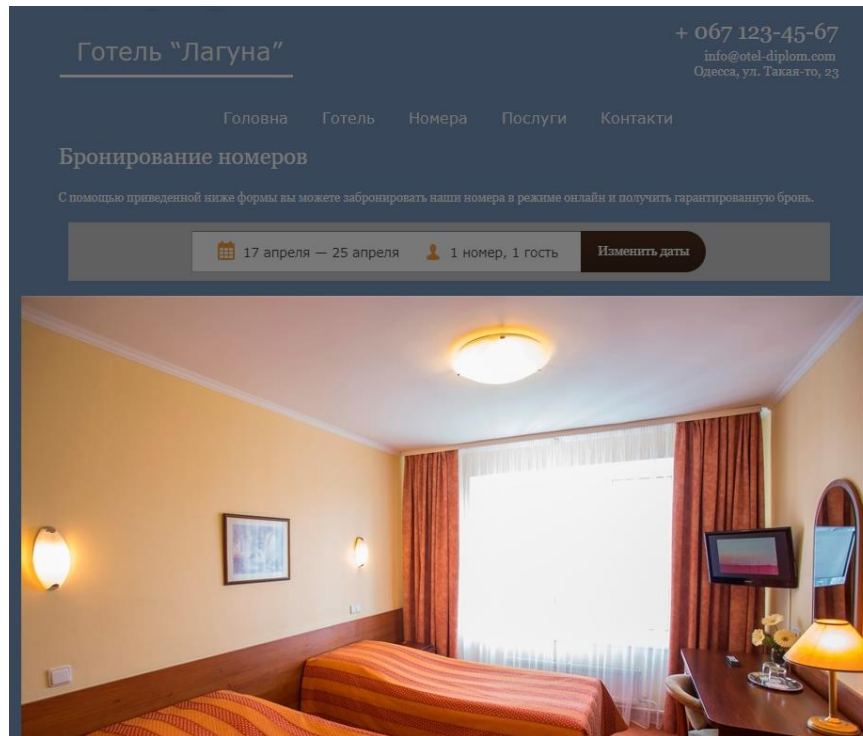


Рисунок 44 – Сторінка бронювання для зміни дат

На інших сторінках сайту можна продивитись інформацію про послуги та ціни на них, інформацію про місце розташування готелю та можливість доїхати до нього різними видами транспорту, про пам'ятки архітектури міста, цікаві екскурсії та спеціальні пропозиції.

На сторінці Контакти надані контактні телефони та електронна адреса для зв'язку з персоналом готелю.

ВИСНОВКИ

У бакалаврській кваліфікаційній роботі розглянуті питання, щодо проектування та реалізації інформаційної системи міні-готелю. Система, представлена в роботі, є ключовою ланкою в автоматизації роботи готельного підприємства. Рішення є комплексним, дякуючи чому, значно знижуються витрати на інтеграцію і навчання користувачів системи, в порівнянні з розрізненим середовищем. За допомогою подібної системи, користувачі можуть витрачати значно менше часу на роботу з даними в інформаційній системі, будь-то заповнення нових позицій або обробка замовлень клієнтів.

Інформаційні системи в готельному бізнесі є економічно доцільними та ефективними за умови, якщо вони приносять готелю додаткові доходи, забезпечують конкурентні переваги на ринку, підвищують частку ринку, знижують витрати, вдосконалюють процес обслуговування, підвищують ефективність роботи окремих підрозділів та готелю в цілому. Також впровадження даних систем може кардинально змінювати методичну, інформаційну та технологічну складові управлінських процесів і здійснювати їх на якісно новому, більш ефективному рівні, що значно підвищує якість надання послуг у готелі. Згідно даних, Україна відстає від провідних розвинених країн світу в сфері впровадження інформаційних технологій і автоматизації управління готелями. Проте, в той же час є великі перспективи подальшого розвитку в цьому напрямку, оскільки на українському ринку є понад десяти високоякісних сучасних систем автоматизації управління готелями, причому деякі з них – це не адаптація зарубіжних систем, а оригінальні розробки українських компаній і тому найбільш пристосовані до умов України. Впровадження таких систем істотно підвищить конкурентоспроможність українських готелів.

В ході виконання кваліфікаційної роботи був спроектований програмний продукт для автоматизації пошуково-інформаційного сервісу міні-готелю, який дозволяє переглядати номери готелю на своєму персональному комп'ютері, а також відправляти замовлення на бронювання номерів, що

економить час клієнта і збільшує продуктивність роботи робітників готелю. В ході виконання кваліфікаційної роботи були вирішені наступні завдання:

- проведений аналіз сайтів та програм готелів;
- зроблена постановка завдання;
- обґрунтована архітектура програмного комплексу;
- обрані програмні засоби для розробки застосунків;
- спроектована та створена база даних на сервері;
- спроектовані застосунки;
- проведено тестування системи.

Інтерфейс програми оптимізований для роботи з великими потоками інформації. Функції пошуку доступні кожній сторінці програми. Це дуже важливо, оскільки потік інформації, що проходить через компанії подібного роду, вельми значний, і обробляти такий об'єм уручну просто неможливо. Для вдалого впровадження програмного продукту і тривалої подальшої експлуатації важливо враховувати всі фактори, які забезпечуватимуть не лише безпомилкову роботу системи, а й безпечні умови її використання. Правильно організована робота по забезпеченню збільшення програми інноваційними технологіями приведе до збільшення користувачів та людей, зацікавлених у вкладанні коштів, для розвитку даної системи.

Мета поставленої роботи виконана.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Державний комітет статистики України. Офіційний сайт. URL: <http://ukrstat.gov.ua> (дата звернення 18.04.2019).
2. Ткаченко В. Гостиничные системы: функции и возможности. Сети & Бизнес, 2011. №2(57). С.74-84.
3. Система управління «Fidelio». URL: https://studopedia.su/5_45088_FIDELIO-V.html (дата звернення 20.04.2019).
4. Інформаційна система JSolution. URL: <https://jsolutions.ua/testoviy-dostup#hotel> (дата звернення 22.04.2019).
5. Lite PMS – система управління для власників апартаментів, хостелів, готельних. Модуль бронювання онлайн. URL:<https://litepms.ru> (дата звернення 25.04.2019).
6. Фаулер Мартин. Архитектура корпоративных программных приложений. СПб.: Вильямс, 2006. 544 с.
7. Дон Бокс, Крис Селлз. Основы платформы .NET. Общезыковая исполняющая среда. СПб.: Вильямс, 2003. 288 с.
8. Вийера Роберт. Программирование баз данных Microsoft SQL Server 2008. Базовый курс. СПб.: Вильямс, 2007. 832 с.
9. Дейтел Х. С# в подлиннике. Наиболее полное руководство. СПб.: БХВ-Петербург, 2006. 1056 с.
10. Дэвид А. Марка, Клемент Л. МакГоуэн. Методология структурного анализа и проектирования SADT. М.: Мета Технология, 1993. 231 с.