

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

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

Кваліфікаційна робота бакалавра

на тему: Розробка інформаційної системи управління готелем

Виконала студентка
групи КН-19
спеціальності 122 Комп'ютерні науки
Задойко Анна Володимирівна

Керівник к.т.н. доцент
Гнатовська Ганна Арнольдівна

Консультант _____

Рецензент к.геогр.н., доцент
Кузніченко С.Д

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ	6
ВСТУП.....	7
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ.....	9
1.1 Дослідження предмета, цілей і особливостей інформаційної системи керуванням готелем.....	9
1.2 Аналіз існуючих систем	11
1.3 Постановка завдання	20
2 ВИБІР І ОБГРУНТУВАННЯ АРХІТЕКТУРИ СИСТЕМИ ТА ПРОГРАМНИХ ЗАСОБІВ РЕАЛІЗАЦІЇ	22
2.1 Загальна архітектура системи	22
2.2 Вибір Web-сервера	25
2.3 Вибір системи управління базами даних	27
2.4 Вибір мови програмування	29
2.4.1 Об'єктно-орієнтована мова Java.....	29
2.4.2 Мова розмітки гіпертексту HTML.....	30
2.4.3 Формальна мова розмітки CSS	32
3 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМ	35
3.1 Моделювання засобами графічної мови IDEF0	35
3.2 Проектування інформаційної системи за допомогою діаграми потоків даних DFD	40
3.3 Проектування та створення бази даних.....	43
4 ПРАКТИЧНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ	53
4.1 Інтерфейс користувач-менеджер	53
4.2 Інтерфейс користувач-бухгалтер	58

	5
4.3 Інтерфейс користувач-адміністратор	61
ВИСНОВКИ	64
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	65

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

ІС – інформаційна система

ІТ – інформаційні технології

ПЗ – програмне забезпечення

БД – база даних

СУБД – система управління базами даних

SQL – Structured Query Language – мова структурованих запитів

HTTP – Hypertext Transfer Protocol – протокол передачі гіпертексту

CGI – Common Gateway Interface – загальний інтерфейс шлюзу

HTML – HyperText Markup Language – мова розмітки гіпертексту

CSS – Cascading Style Sheets – каскадні таблиці стилів

SADT – Structured Analysis and Design Technique – структурований
аналіз і техніка проектування

IDEF0 – Integration DEFinition language 0 – метод функціонального
модельовання та графічного представлення процесів,
призначений для формалізації та опису бізнес-процесів

DFD – Data Flow Diagrams – модель проекту, графічне представлення
потоків даних в інформаційній системі

ВСТУП

Революція інформаційних технологій змінила характер і методи ведення бізнесу. Використання можливостей технічного обміну сьогодні дозволяє простіше і швидше створювати та продавати споживачам пакети послуг, вирішувати завдання фінансового і операційного менеджменту, маркетингового планування, підвищувати конкурентоспроможність та кількість продажів.

З кожним днем інформаційні технології впроваджують і розширюють нові функціональні можливості та активно використовуються у різних сферах людської діяльності, пов'язаних з обробкою інформації і представлення даних. Використання інформаційних систем та баз даних стає обов'язковою складовою будь-якої організації.

Цей вплив торкнувся і сфери готельного бізнесу. На сьогодні готель представляє собою комфортні номери, професіоналізм персоналу та всі складові, які пов'язані з відпочинком і роботою. Кожен готель має свої особливості. Це може бути SPA-салон, басейн, ресторан, бізнес-центр тощо. Тому для того, щоб керувати всім цим необхідно працювати зі спеціалізованим програмним забезпеченням.

Інформаційне забезпечення готельних підприємств дозволяє автоматизувати всі етапи обслуговування гостя, починаючи від бронювання, заселення та закінчуючи остаточним розрахунком, а також основні бізнес-процеси – від роботи покоївок до організації звітності на підприємстві. Вони мають таку можливість, як забезпечення керування усіма послугами і системами готелю, які доступні у номері.

Актуальність розробки інформаційної системи керування готелем зумовлена необхідністю впровадження сучасних інформаційних технологій у діяльність готелів, що є необхідною умовою успішного функціонування, оскільки такі показники, як надійність, точність, передача інформації,

висока швидкість обробки та оперативність визначають ефективність управлінських рішень у сфері готельного бізнесу.

Метою дипломної роботи є розробка інформаційної системи управління готелем, яка забезпечить можливість керування всіма бізнес-процесами готельного підприємства засобами створеного спеціалізованого програмного забезпечення, що значно підвищить якість надання послуг у готелі. Для того, щоб досягти поставленої мети треба визначити і вирішити наступні завдання:

- аналіз предметної області у керуванні готелем;
- аналіз існуючих систем;
- обґрунтування вибору програмних засобів розробки та технологій для вирішення поставленого завдання;
- проектування інформаційної системи;
- розробка бази даних;
- написання коду програми з використанням мови програмування Java;
- реалізація клієнтської та серверної частин веб-додатка.

Основним призначенням інформаційної системи управління готелем є забезпечення оперативного доступу до управління всіма бізнес-процесами, які забезпечують ефективне управління у готельному підприємстві, що підвищить ефективність та зручність надання послуг у готелі, використовуючи сучасні інформаційні технології.

Дипломна робота містить у собі 64 сторінки, 11 таблиць, 30 рисунків, 14 посилань.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ

1.1 Дослідження предмета, цілей і особливостей інформаційної системи керуванням готелем

У даному розділі розглядається робота готельного комплексу. Визначення функції персоналу та особливостей їх роботи. Існуючі у готелі інформаційні системи об'єднуються у різні служби. На першому місці знаходиться внутрішня організація, яка покликана керувати працівниками і направляти їх діяльність на досягнення цілей підприємства.

Готельне підприємство представляє собою велику структуру, яка об'єднана різними службами. Вони, у свою чергу, виконують необхідні функції для того, щоб готель міг працювати [2].

Основні служби готелю наведені нижче:

- прийом та розміщення;
- обслуговування;
- господарська служба;
- безпека;
- бухгалтерія;
- відділ персоналу;
- відділ ІТ.

Служба прийому та розміщення відповідає за реєстрацію гостей, розподіл номерів, реєстрація заселення та виселення гостей, а також надає інші послуги.

Процес обслуговування клієнтів можна розділити на декілька етапів:

- бронювання;
- прийом, реєстрація та розміщення гостей;
- надання стандартних та додаткових послуг проживання;
- остаточний розрахунок та оформлення виїзду.

Відділ обслуговування є найважливішим у готелі, оскільки його працівники перебувають у постійному контакті з клієнтами і виконують усі функції, пов'язані з їхнім безпосереднім обслуговуванням.

Господарську службу очолює менеджер, якому підпорядковуються портье, коридорні, вантажники, ліфтери, консьєржі, швейцари та водії (які обслуговують орендовані автомобілі та паркування машин гостей).

За чистоту всього готелю та номерів відповідає відділ господарської служби. Щодня відділ звіряє дані про номери з даними на рецепції. Повідомлення з рецепції про виселення гостя з номера, який фіксуються у відповідній колонці журналу прибирання номерів. В іншій колонці журналу зазначається ім'я прибиральника. Після завершення прибирання у журналі робиться відповідна відмітка. Інформація про завершення прибирання номера передається на рецепцію [2].

До технічних заходів безпеки у першу чергу відносяться телевізійні камери, за допомогою яких можна переглядати всі зони загального користування та багато службових приміщень. Записи повинні зберігатися не менше 24 годин. Впровадження електронних замків значно покращило безпеку номерів. Крім того, щоб запобігти крадіжкам, багато готелів на прохання клієнтів встановлюють у номерах індивідуальні сейфи [1].

Служба безпеки готелю займається такими завданнями:

- розробкою процедур реагування на надзвичайні події;
- повсякденною безпекою гостьових кімнат;
- контролем ключів;
- запобігання крадіжкам, контролю за замками;
- контролем доступу до будівлі готелю;
- системою охоронної сигналізації;
- контролем території;
- зовнішнім освітленням;
- системою спостереження за допомогою телекамер;
- сейфами та вогнетривкими шафами;

– збиранням та зберіганням інформації.

Бухгалтерією готелю займається фінансовим відділом, який відповідає за наступні операції:

- розрахунки за рахунками клієнтів;
- розрахунок щоденного доходу для кожного сегмента готелю, що генерує дохід;
- складання щоденного звіту генеральному директору, в якому подано доходи за підрозділами, завантаження номерного фонду та деякі інші показники порівняно з минулим періодом;
- нарахування та виплата заробітної плати;
- ведення фінансового та управлінського обліку.

Відділ кадрів є важливою передумовою для безперебійної роботи всього процесу обслуговування, для створення і розвитку команди, здатної ефективно виконувати свою роботу, а також для створення приємної атмосфери в готелі [1].

Відділ ІТ допомагає вирішувати наступні завдання:

- оптимізація бізнес-процесів, зниження паперового документообігу;
- посилення контролю за послугами та персоналом;
- підвищення якості обслуговування гостей;
- оптимізація операційних витрат;
- статистичні та аналітичні функції.

Для розробки інформаційної системи керування готелем була проаналізована робота готельного комплексу. Були визначені основні складові даної системи. Готельний бізнес представляє собою велику структуру, яка об'єднана різноманітними службами.

1.2 Аналіз існуючих систем

Наразі існує досить багато різних PMS-систем, призначених для менеджерів та адміністраторів. Більшість з них є платними програмними

продуктами зі схожим функціоналом. Також велика кількість з них мають можливість інтеграції з Channel Manager. У даному розділі будуть розглянути такі системи, як: Senium, JSolution та Parus.

З початку буде розглянута система Senium. Вона є постачальницею технологій для індустрії гостинності. Senium надає власникам готелів інтегрований доступ до всієї подорожі гостя за допомогою корпоративної системи, розробленої та підтримуваної експертами галузі гостинного бізнесу.

Використовуючи це універсальне та динамічне PMS-рішення можна легко налаштувати рішення для обробки складних групових резервувань кількох готелів, включаючи ефективні та швидкі процеси реєстрації заїзду та виїзду. Housekeeping UX, включаючи додатки для всього службового персоналу, Front Desk UX дозволяє працювати на стойці реєстрації у будь-якому веб-браузері, а Senium FastTrack забезпечує самообслуговування реєстрації заїзду та виїзду для гостей [3].

Дана система має такі складові, які має свої особливості, як:

- операції на стойці реєстрації;
- бронювання та керування фолію;
- управління групами;
- ведення господарства.

Управління групами:

- виконання дії з декількома бронюванням однією дією (заїзд/виїзд та призначення номерів);
- швидкий імпорт списків гостей;
- створення кілька резервувань номерів і налаштування кожного з них окремо, коли це необхідно;
- швидко, вручну або автоматично можна призначити кілька номерів і можливість фільтрувати за бажаними функціями та розташуванням;
- керування знижками на всі бронювання в групі однією дією.

Операції на сторінці реєстрації (рис. 1.1):

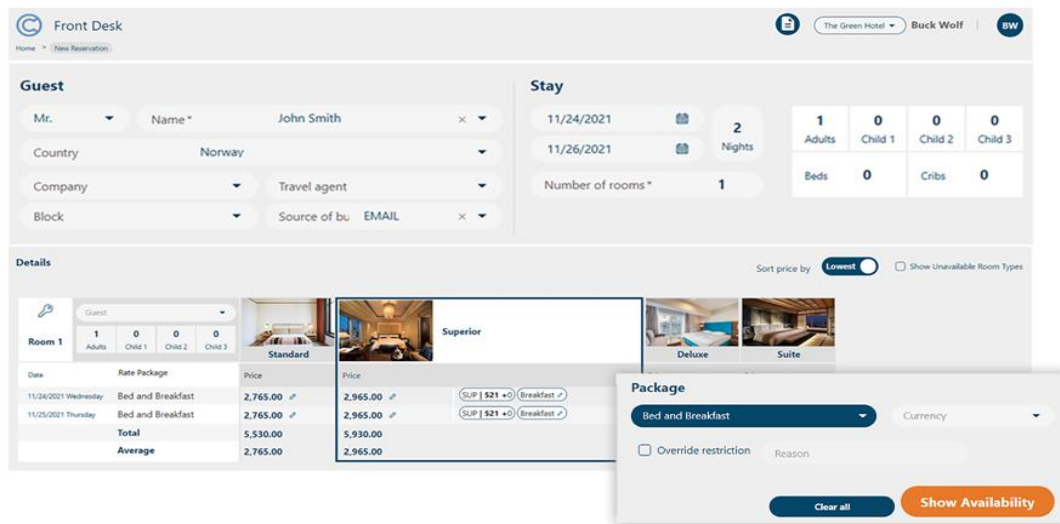


Рисунок 1.1 – Операції на стойці реєстрації

- швидкий процес реєстрації та виселення;
- керування всіма заброньованими компонентами бронювання в одному місці з усіх продуктів Senium, включаючи квитки, заходи, відмови тощо;
- бронювання за статусом або всі з швидким фільтруванням та пошуком;
- можливість перекваліфікувати, переміщувати, призначати, оновляти або здійснювати перехресні продажі за допомогою єдиного «простого у використанні» діалогового вікна;
- автоматично можна призначити номери на основі вподобань гостей або шукайте бажані функції чи місце розташування номера;
- легке створення акцій.

Бронювання та керування фоліо (рис. 1.2):

- автоматична маршрутизація фоліантів на основі шаблонів або ручної конфігурації;
- продаж додаткових продуктів та можливість підвищити вартість номерів під час бронювання;

- налаштування та продаж розширених пакетів з номерами, заходами та роздрібними товарами;
- динамічне встановлювання цін на номери та товари на основі управління тарифами або контрактів, а також промо-акцій та налаштувань знижок.

Бронювання та керування фоліо можна подивитися на рис. 1.2.

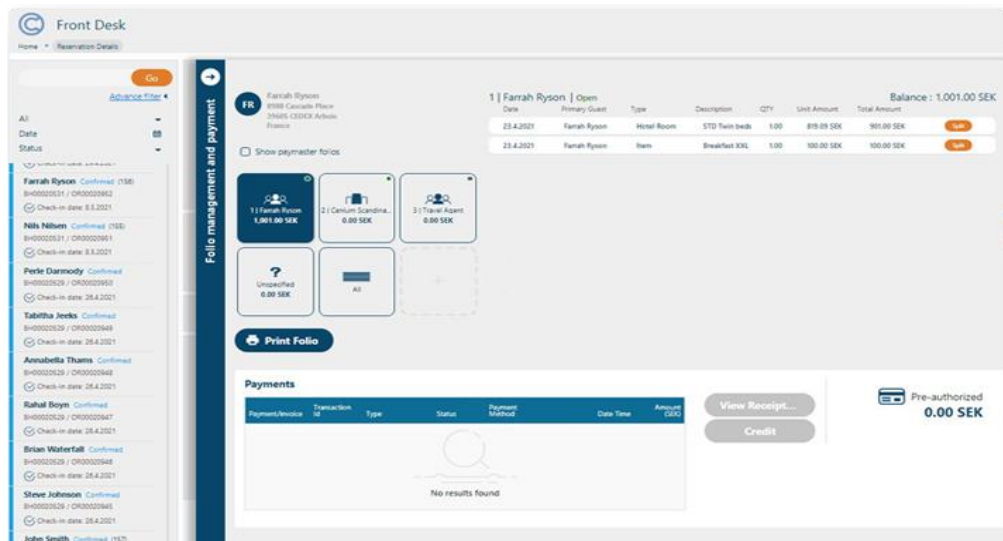


Рисунок 1.2 – Бронювання та керування фоліо

Наступним після Senium розглядається така програма, як JSolution, яка представляє з себе українську компанію з розробки різних програмних забезпечень у керуванні бізнесом. Це хмарна система, з якою можна автоматизувати управління деяких видів підприємств. Дана система використовує будь-яку операційну систему, інтегрується з різними обладнаннями та працює на різноманітному устаткуванні. Дане підприємство працює на ринку з 2017 року.

JSolution маю таку особливість, як створення номерного фонду готелю за типами: хостел (ліжко) або готельний (номер повністю). Формування структури за поверхами, приміщеннями, графічне розташування номерів готелю на схемі. Можливість додавання характеристик для кожного номера у

вигляді переліку меблів і обладнання із зазначенням вартості. Для кожного номера можливо додавання різних опцій щодо графіка прибирання або міні-бар.

Єдина база даних для всіх модулів, що дає змогу у режимі реального часу бачити дані, внесені іншими користувачами системи в інших модулях. Завдяки модульній структурі системи, можна поетапно автоматизувати різні бізнес-процеси вашої компанії. Система дозволяє організовувати графіки роботи і зміни: призначати начальників змін, видавати картки з певними правами доступу зареєстрованим особам (рис. 1.3).

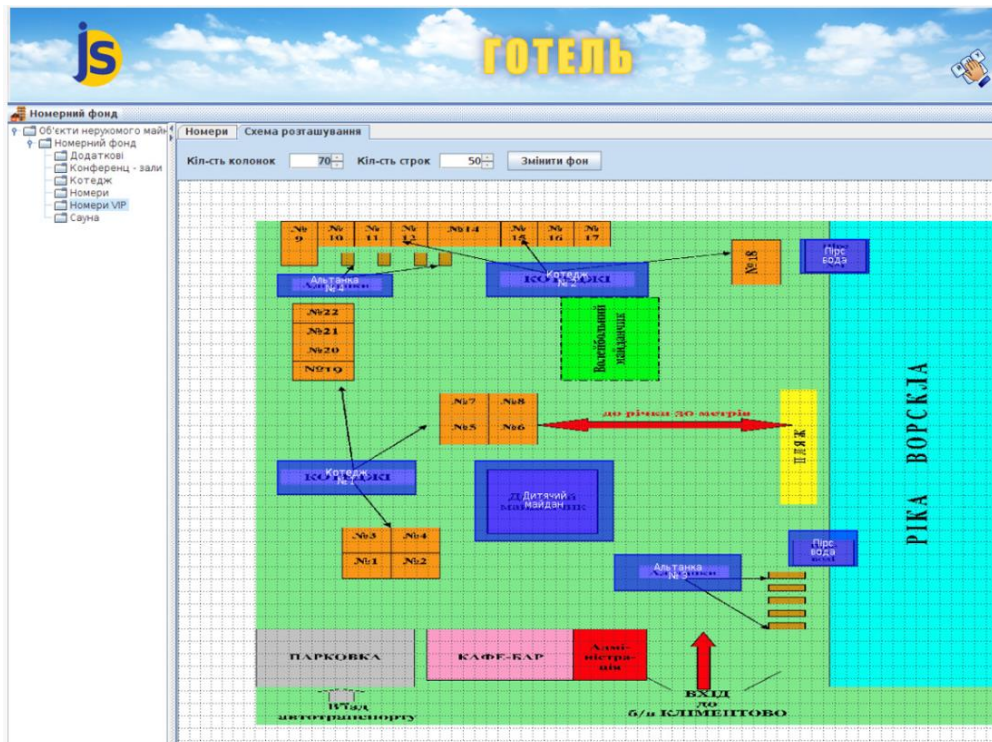


Рисунок 1.3 – Схема розташування

Система дозволяє вести облік всіх оброблених звітів з вибірковою статистикою по конкретному касиру, зберігати інформацію про поточний залишок готівки у касі, видавати і отримувати готівку, друкувати звіти про транзакції.

Усі потрібні функції доступні в єдиному інтерфейсі: аналіз всього переліку номерів, реєстрація гостей з внесенням персональних даних у базу даних клієнтів, надання стандартних і додаткових послуг з можливістю їх прив'язки до основного рахунку проживання, облік авансових платежів за послуги.

У вкладці «Ціни» можна коригувати ціни відповідно до сезону, кількості гостей та кількості додаткових місць. Система здатна обробляти запити, звернення та скарги клієнтів за допомогою механізму розгляду скарг.

Управління завантаженням номерів (рис. 1.4). Наявність місць на вибраний період і календар завантаження з графічним відображенням в розрізі місяця.

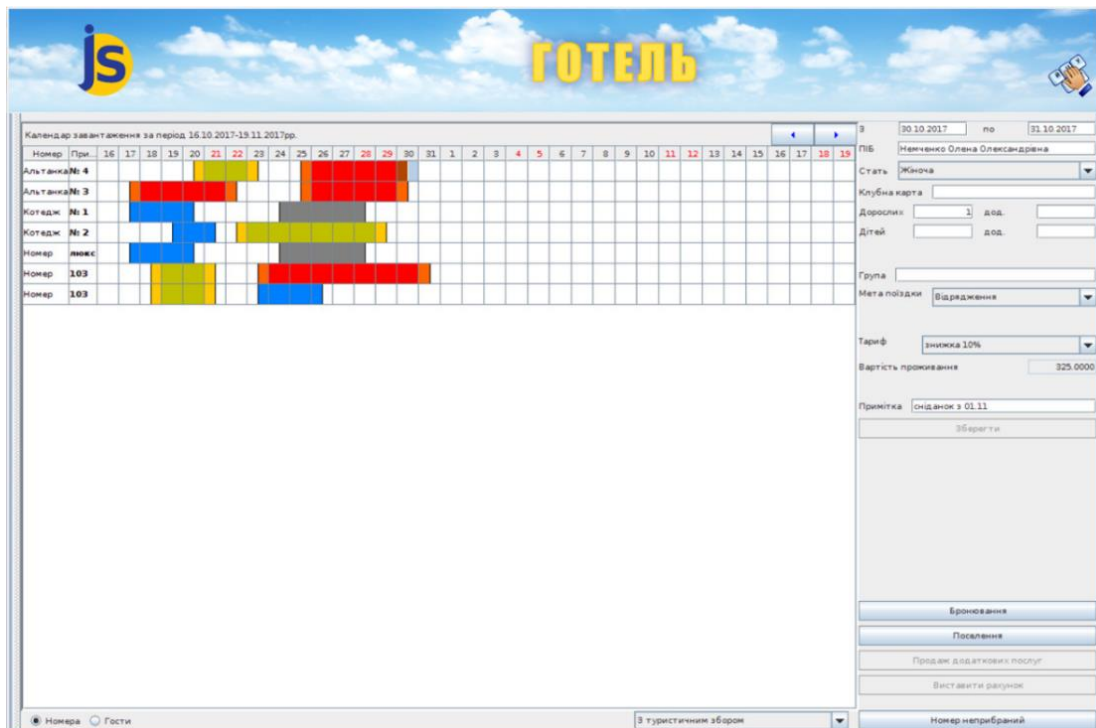


Рисунок 1.4 – Календар завантаження

Особливості системи автоматизації готелів jSolutions [4]:

- розрахунки з контрагентами та гостями;
- програма лояльності клієнтів (знижки та подарункові картки);

- підготовка замовлень на клінінгові роботи, планування та контроль виконання робіт;
- інтеграція системи з сайтом готелю;
- крос-платформність рішень дозволяє працювати на різних операційних системах Mac OS, Linux, Windows;
- робота з різними редакторами для формування друкованих форм документів та звітів із системи: OpenOffice, LibreOffice, MicrosoftOffice;
- інтеграція з Channel Manager.

Останньою у цьому розділі буде система Парус, яка є одною з перших IT-Підприємств України. Вона була заснована у 1989 році. Основною метою є впровадження технологій, що допомагають автоматизувати установи санаторно-курортного типу (готелів, лікувальних санаторіїв, будинків відпочинку тощо).

Програмний продукт дозволяє одночасно автоматизувати два блоки оперативного обліку на підприємстві (готельний та ресторанний), а також організувати гнучку взаємодію зазначених блоків в єдиній інформаційній базі. Програма може поставлятися комплексно чи модулями [5].

Управління санаторно-курортним комплексом забезпечить:

- підвищення прибутковості та підконтрольність бізнесу;
- аналіз, планування та управління фінансами і ресурсами компанії для підвищення ефективності та прибутковості підприємства;
- використання інструментів, що підвищують ефективність щоденної роботи за своїми напрямками;
- своєчасний оперативний облік, регламентований облік, калькуляція, розрахунок собівартості.

Основа рішення – максимальна зручність користувачів:

- комплексний програмний продукт;

- ведення єдиної бази даних готельних послуг, харчування і продуктів харчування, витратних матеріалів, медикаментів, обстеження та процедур;
- єдина система взаєморозрахунків за всіма блоками;
- управлінський облік: доходи і витрати, фонди, управлінський баланс;
- оперативне управління готельними номерами, кейтерингом та медичними послугами;
- понад 100 різних аналітичних звітів;
- розмежування прав доступу на рівні структури компанії;
- обмін даними з фронт-офісом та обліковими системами.

Програмне забезпечення складається з двох основних блоків:

Готельний блок (рис. 1.5), який має наступні програми:

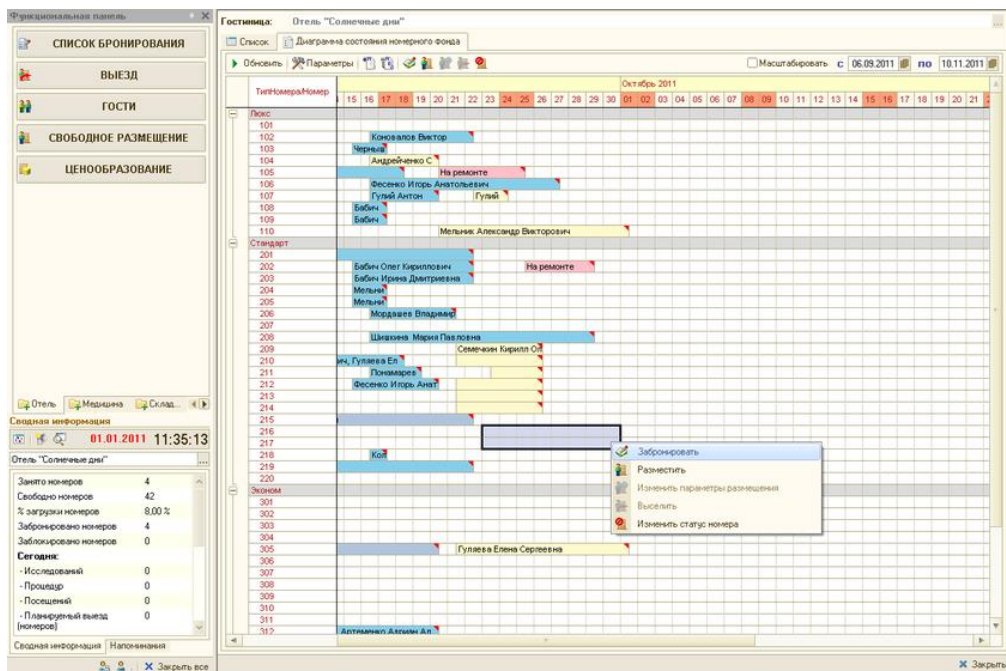


Рисунок 1.5 – Готельний блок

- бронювання – перегляньте інформацію про номер, забронуйте його та сплатіть заздалегідь на бажані дати.

- реєстрація – дозволяє гостям реєструватися з бронюванням або без нього.
- зберігання історії кожного гостя, контактну інформацію, вподобання, перелік наданих послуг та взаємні платежі.
- керування платіжками (готівковими та безготівковими), цінами та знижками з бізнес-партнерами;
- адміністрування – дозволяє менеджерам зберігати та керувати готельною та адміністративною документацією.

Ресторанний блок (рис. 1.6), який відповідає за виконання всіх операцій, які беруть участь у виробництві і реалізації продукції кухні і бара (облік продажів, виробничий облік, розрахунок собівартості продукції, облік рухів і залишків, потужна система аналітичної звітності).

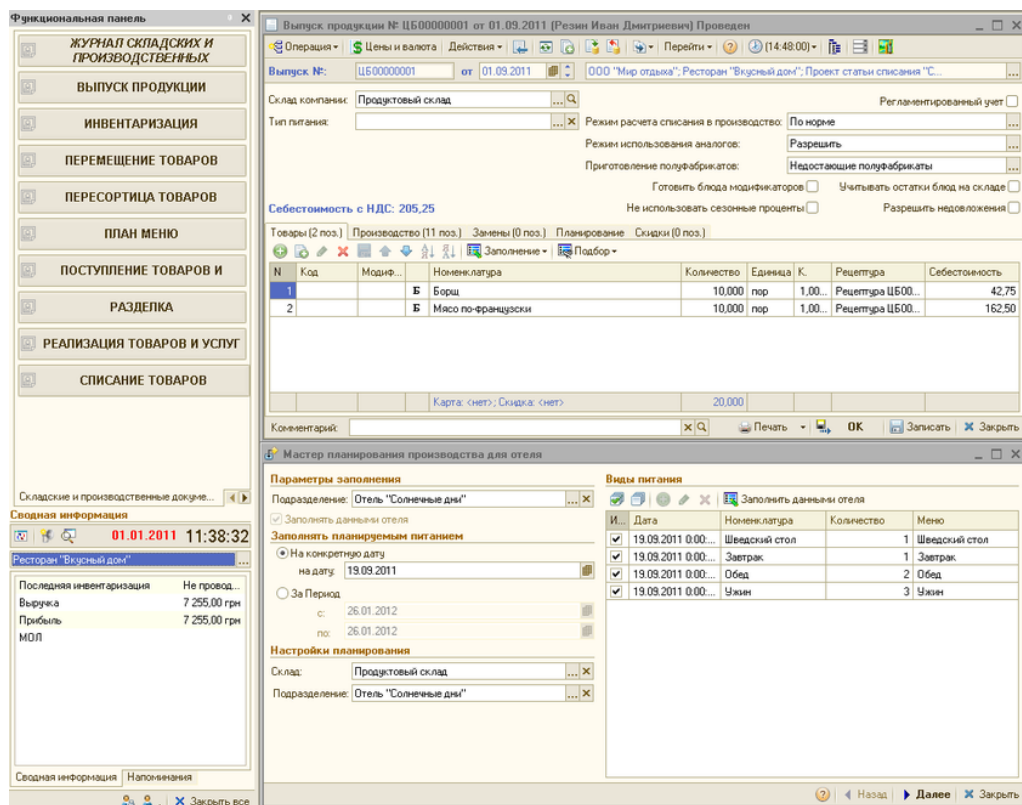


Рисунок 1.6 – Ресторанный блок

Також можливий обмін даними з продуктами «1С:Ресторан» і RestArt для автоматизації обслуговування гостей у закладах харчування і торгівлі, навіть у випадках, коли у готелі є «безготівкова зона» (гості пред'являють картку для оплати - електронні квитанції за номер).

Програма автоматизації готелю Парус може підключатися до системи «1С:Підприємство 8. Бухгалтерія Україна» для передачі рахунків, квитанцій, контактної інформації гостей, записів про надані послуги, а також для контролю фактичної оплати послуг клієнтом.

Після того, як подивились всі особливості роботи систем по керуванню за готелем. Можна зрозуміти, що вони мають приблизно один й той самий функціонал та зовнішність. Вони є лише маленькою частиною зі всієї кількості інформаційних систем, які допомагають управляти готельним підприємством. Використання подібних інформаційних систем мають наступні переваги: підвищення ефективності роботи всіх складових готелю; вдосконалює процес обслуговування; спрощення деяких процесів у підприємстві; зниження витрат та збільшує конкурентоспроможність між іншими готелями на ринку. Недоліками розглянутих систем є те, що вони платні. Тому, що початківці, які не мають достатньо бюджету, не зможуть користуватися ними у керуванні готелем.

1.3 Постановка завдання

У даній дипломній роботі треба розробити інформаційну систему керуванням готелем. Вона повинна бути реалізована у вигляді інформаційної системи з великою кількістю функцій для адміністратора, бухгалтера та менеджера.

Інформаційна система управління готелем розташована у локальному просторі, за допомогою якою працівники готельного бізнесу змогли якісно виконувати необхідні дії для керуванням готелем (наприклад, бронювання номеру або оновлення кімнати). Доступний перегляд гостей готелю, кімнат,

зроблених платежів, список усіх працівників та бронюванню номерів для користувача.

При розробці інформаційної системи управління готелем будуть розглядатися три види користувачів (адміністратор, бухгалтер та менеджер).

Користувач-бухгалтер має наступні дії:

- робити звіт та чек;
- дивитись список кімнат, які заброньовані;
- дивитися список клієнтів;
- працювати з платіжками.

Користувач-менеджер має доступ до таких дій, як:

- робити бронювання;
- реєструвати клієнта;
- дивитися та показувати клієнтам список вільних кімнат;
- працювати з календарем резервувань.

Користувач-адміністратор має доступ до всіх дій користувача-менеджера та користувача-бухгалтера:

- редагувати або дивитися список кімнат;
- робити бронювання;
- реєструвати клієнта;
- робити звіт та чек;
- дивитися список клієнтів та редагувати його;
- працювати з календарем резервувань;
- працювати з працівниками;
- складати список, які кімнати вільні на обрану дату.

Розробка даного проекту буде реалізована як інформаційна система з різними правами доступу до управління системою. Було розглянуто три види прав доступу: адміністратор, бухгалтер та менеджер. Кожен з них має певні можливості у керуванні даною інформаційною системою.

2 ВИБІР І ОБГРУНТУВАННЯ АРХІТЕКТУРИ СИСТЕМИ ТА ПРОГРАМНИХ ЗАСОБІВ РЕАЛІЗАЦІЇ

2.1 Загальна архітектура системи

Для того щоб розробити інформаційну систему керування готелем необхідно зробити вибір та обґрунтувати загальну архітектуру системи і програмних засобів реалізації з урахуванням усіх вимог до даної системи.

Треба здійснити вибір програмних засобів реалізації після визначення бізнес-логіки і функцій, які забезпечують систему дистанційного навчального центру. Даний крок є одним із найважливіших у проектуванні інформаційної системи. Для вправної роботи проекту необхідно використовувати потужні програмні продукти з відкритою архітектурою.

Сучасні технології сьогодні мають дуже високий рівень розвитку, ще це допомагає програмісту будувати інформаційну систему різної складності, масштабності або функціональності.

Якщо ж розглядати бізнес-вимоги на основі показників різних оцінок бізнесу, то виникають додаткові труднощі, вирішення яких полягає у забезпеченні раціонального підходу до процесу проектування, впровадження та подальшої експлуатації інформаційних систем. Одним з основних показників якості створення інформаційної системи є обрана архітектура.

Архітектура інформаційної системи будується на основі апаратних, телекомунікаційних і програмних засобів. Рівень розвитку кожного з компонентів визначається складністю інформаційної системи та технологією обробки даних, що зумовило появу таких схем обробки даних [6]:

- телеобробка;
- файл-сервер;
- клієнт-сервер;
- Internet-система;
- сховище даних;
- система оперативної аналітичної обробки даних тощо.

Ідея розподілених систем інформації полягає у тому, що де кілька компонентів мережі кооперуються для виконання однієї роботи. Технологія клієнт-сервер є найпопулярнішою середою для реалізації розподілених прикладних завдань.

Інформаційна система клієнт-серверна складається з наступних компонентів [6]:

- програмне забезпечення сервера;
- програмне забезпечення кінцевого користувача;
- проміжне програмне забезпечення.

Серверне програмне забезпечення забезпечує обслуговування клієнтів. Архітектура «клієнт-сервер» реалізується при використанні СУБД. Вона забезпечує механізми блокування та багатокористувацький контроль доступу, які захищають дані від небезпеки одночасного доступу. Крім того, сервер бази даних повинен захищати дані від несанкціонованого доступу, оптимізувати запити до бази даних, забезпечувати цілісність даних і контролювати завершення транзакцій.

Програмне забезпечення кінцевого користувача включає засоби розробки програм і генератори звітів, у тому числі електронні таблиці та текстові процесори. За допомогою цього програмного забезпечення користувачі встановлюють з'єднання з сервером, формують запити, які автоматично генеруються у SQL-запити і надсилаються на сервер. Сервер отримує та обробляє запити, а потім передає результати клієнтам. Проміжне програмне забезпечення – це частина клієнт-серверної системи, яка з'єднує програмне забезпечення кінцевого користувача з сервером [7].

Схема клієнт-сервер проста: клієнт надсилає серверу запит на отримання необхідних даних; сервер отримує їх, обробляє і надсилає клієнту тільки ті дані, які були замовлені. Дворівнева модель клієнт-сервер оптимальна для підприємств, якщо кількість користувачів не велика тому, що операційна система сервера перевантажується управлінням численними з'єднаннями з сервером при обслуговуванні великої кількості клієнтів.

Архітектура клієнт-сервер – це мережева інфраструктура, в якій сервери надають певні послуги, а клієнтські комп'ютери їх використовують. У таких системах сервер в основному виконує роль постачальника послуг доступу до бази даних та називається дворівневою архітектурою (рис. 2.1).

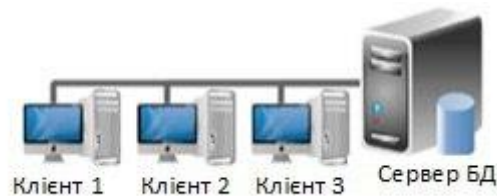


Рисунок 2.1 – Дворівнева модель

У міру зростання масштабу системи може виникнути необхідність заміни апаратного забезпечення серверних і клієнтських машин. У збільшені користувачів виникає потреба у синхронізації версій великої кількості додатків. Для вирішення цієї проблеми використовують багаторівневі архітектури (три і більше рівнів). Деякі поширені програми переносяться на виділений сервер, що знижує вимоги до продуктивності клієнтських комп'ютерів. У багаторівневій архітектурі (рис. 2.2) з виділеним сервером додатків можливе використання портативних пристроїв [5].



Рисунок 2.2 – Багатоланкова архітектура

Цей тип архітектури використовується, коли додаток має високі вимоги до ресурсів. У цьому випадку є сенс перенести його на окремий сервер і таким чином знизити вимоги до продуктивності робочих станцій.

При побудові інформаційної системи керуванням готелем була обрана дворівнева архітектура клієнт-сервер тому, що проект будуть використовувати невеликий круг користувачів. Вона надає можливість реалізувати зв'язок між сервером БД та клієнтами. Де один запитує якусь інформацію, а інший обробляє запит та відправляє тому відповідь. Також обрана архітектура легка у виконанні.

2.2 Вибір Web-сервера

Web-сервер – це програма, яка вміє приймати HTTP-запит, і віддавати Web-сторінку у відповідь. Відповідно, основною задачею веб-серверу є зрозуміти запит користувача і віддати необхідну сторінку згідно цього запиту (рис. 2.3).

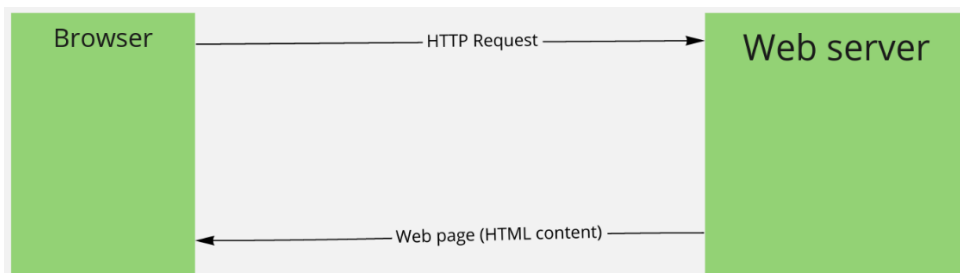


Рисунок 2.3 – Web-сервер

Коли Інтернет лише з'явивсь, Web-сторінки були статичними. Тобто, сторінки не реагували на дії користувача, і ніяк не були персоналізовані.

У такому випадку програміст не міг побудувати, наприклад, Web-додаток для інтернет банку. У кожного користувача є свої унікальні

персональні дані та свій рахунок. Цю всю інформацію потрібно підтягувати з якоїсь бази даних, відповідно веб-сторінка має бути динамічною.

Щоб вирішити цю проблему, придумали технологію у Java сервлети. Ключова ідея проста – програміст запускає спеціальну програму, яка називається контейнер сервлетів. Всередині цієї програми він запускає власний Java код, який постійно виконується та чекає надходження HTTP запиту – сервлет.

Коли приходить HTTP запит, то у програміста вже є готова запущена програма для його виконання. Сервлет має такі переваги, як швидкість та масштабуємість. Він представляє собою повноцінну Java програму, в якій програміст може робити все те саме, що і у звичайних Java програмах.

Контейнер сервлетів – це програма, яка запускає інші програми-сервлети. Фактично це Web-сервер. У контейнері сервлетів (рис. 2.5) може бути запущено одночасно декілька сервлетів. Програміст може додавати і видаляти окремі сервлети, не зупиняючи контейнер сервлетів (отже, не зупиняючи роботу всіх сервлетів) [8].



Рисунок 2.5 – Контейнер сервлетів

Важливо розуміти, що контейнер сервлетів – це лише специфікація, по якій створюються конкретні реалізації. Тобто немає програми контейнер сервлетів. Є програми, які реалізують специфікацію контейнер сервлетів.

Tomcat – open-source реалізація контейнера сервлетів. Одна з найпопулярніших реалізацій. За допомогою Apache Tomcat програміст

можете отримати необхідні ресурси та методи розгортання веб-додатків на Java. Розгортання програми є відносно простим процесом. Програмісту потрібно лише знайти файли або каталог, який містить веб-додаток. Крім того, програміст отримує деяку робочу програму для завантаження файлу та розгортання його на сервері. Ця програма надає доступ через браузер, оскільки Java-додатки мають багато спільного з веб-сайтами [9].

Інформаційна система повинна десь працювати та відображатися, для цього використовується Web-сервер. Тобто це програма, яка на вході отримує запит, потім обробляє його і на виході віддає необхідну сторінку згідно запиту. Для даного проекту було обрано сервлети, які представляють із себе повноцінну Java програму. Завдяки ним можна: підключати будь-які сторонні бібліотеки, працювати з базою даних, під час обробки запиту, звертатись до якогось іншого сервісу по HTTP API тощо. Для того щоб, це реалізувати буде використано Tomcat, який є одним з найкращих реалізацій контейнерів сервлетів.

2.3 Вибір системи управління базами даних

Багато програм того чи іншого типу вимагають зберігання даних, які будуть доступні навіть після перезапуску програми.

Інформація часто є найціннішим, що має компанія. Уявіть, що банк втрачає свою клієнтську базу або базу транзакцій. У такому випадку можна вважати, що його бізнесу прийшов кінець. Ось чому питання «як зберігати інформацію» є складним і не може бути вирішене за допомогою єдиного принципу.

База даних – це набір даних. Фізично на комп'ютері вона може бути представлена одним або декількома файлами. Крім того, база даних може бути розподіленою та займати багато файлів на різних машинах. Ключовим моментом тут є логічна організація – коли говориться про базу даних, тобто

мається на увазі, що ці дані логічно згруповані в єдине ціле (база даних банку тощо).

Система управління базами даних (СУБД) – це програма, яка може працювати з базами даних. На одному комп'ютері може бути багато баз даних, і одна СУБД може керувати ними всіма. Кожна СУБД працює з форматом, який розуміє тільки вона. Тому, коли програміст чує слово база даних, потрібно уточнювати, база даних якої СУБД. Основна проблема альтернативних СУБД (об'єктно-орієнтованих) полягає у тому, що вони не готові до виробничого використання [10].

SQL (Structured Query Language) уявляє собою мову структурованих запитів. Це текстова мова, яку програмісти використовують для спілкування з СУБД. Він пише текстовий запит, щоб отримати певну інформацію з бази даних або додати якусь інформацію. СУБД приймає цей запит, обробляє його і повертає результат. SQL є декларативною мовою. Це означає, що програміст просить у бази даних надати йому певну інформацію, але він не вказує, як цю інформацію отримати [10].

PostgreSQL – це система управління базами даних з відкритим вихідним кодом, яка використовує мову SQL у поєднанні з багатьма різними функціями, що дозволяють масштабувати складні дані та безпечно їх зберігати. Вона має багато різних функцій, які допомагають розробникам створювати додатки, адміністраторам захищати цілісність даних і створювати відмово стійке середовище, а також допомагають програмістам керувати даними незалежно від того, наскільки великим або малим є набір даних [11].

Для керування базою даних у програмній реалізації в інформаційній системі управлінням готелем була обрана СУБД PostgreSQL. Вона використовує мову структурованих запитів SQL і має відкритий вихідний код, що дає їй переваги перед іншими СУБД. Найчастіше PostgreSQL використовується з об'єктно-орієнтованими мовами програмування (у даному проекті застосовується мова програмування Java).

2.4 Вибір мови програмування

2.4.1 Об'єктно-орієнтована мова Java

Java – це мова програмування, яка є об'єктно-орієнтованою, мережево-орієнтованою та багатоплатформною. яка може бути використана як платформа сама по собі. Це швидка, безпечна та надійна мова програмування для кодування всього, від мобільних додатків та корпоративного програмного забезпечення до додатків для роботи з великими даними та серверних технологій.

У 90-х роках минулого століття, коли розроблялася мова Java, з'явився такий чинник, який відіграє вирішальну роль у майбутньому даної мови. Цим фактором є поява Всесвітньої павутини. Якби Web не з'явився приблизно у той самий час, коли реалізувалася Java, вона могла б залишитися корисною, але маловідомою мовою для програмування побутової електроніки. Однак із появою World Wide Web мова Java була висунута на передній план у розробці комп'ютерних мов, бо Web теж вимагала програм, які можна перенести. Java створювалася на основі мови C++, і у зв'язку з цим багато хто думає, що Java – це просто «версія мови C++ для Інтернету». Однак це було б великою помилкою. Java має суттєві практичні та філософські відмінності. Хоча вірно, що Java зазнала впливу C++, вона не є вдосконаленою версією C++. Java була розроблена для вирішення певного набору проблем, а C++ для іншого [12].

Комп'ютерні мови розвиваються з двох причин: для адаптації до змін у навколишньому середовищі та для реалізації досягнень у мистецтві програмування. Зміни у навколишньому середовищі, які послужили поштовхом до створення Java, були пов'язані з необхідністю створення незалежних від платформи програм, призначених для поширення в Інтернеті. Однак Java також втілює зміни у підході людей до написання програм. Наприклад, Java розширила й удосконалила об'єктно-орієнтовану парадигму, використовувану у C++, додала інтегровану підтримку багатопотоковості та

надала бібліотеку, що спрощує доступ до Інтернету. Однак, зрештою, не окремі особливості Java зробили її такою видатною. Швидше, це була мова загалом. Java була ідеальною відповіддю на вимоги щойно з'явившогося високорозподіленого обчислювального середовища.

Інтернет допоміг вивести Java на передові позиції у програмуванні, а він, своєю чергою, справив глибокий вплив на Інтернет. Крім спрощення веб-програмування загалом, Java створила новий тип мережеских програм під назвою аплет, який змінив уявлення мережевого світу про контент. Java також вирішила деякі з найскладніших питань, пов'язаних з Інтернетом: перенесення та безпека.

Для того, щоб інформаційна система керуванням готелем мала змогу вправно та ефективно працювати буде використовуватися мова програмування Java. На сьогодні Java досі залишається актуальною не дивлячись на те, що мова доволі складна. Після появи Java 8 з'явилося функціональне програмування (наприклад, лямбда) дещо скоротило написання коду. Одною з головних особливостей Java є те, що «написав одного раз – працює де завгодно».

2.4.2 Мова розмітки гіпертексту HTML

Web-сторінка – це інформаційний ресурс або документ, який розташований і доступний у мережі. Вона побудована на таких технологіях:

- HTML відповідає за наповнення веб-сторінки контентом, структуру та зміст;
- CSS використовується для зовнішнього вигляду, позиціонування та оформлення;
- JavaScript може додати реакцію, функціонал на дії користувача.

Щоб створити Web-сайт недостатньо лише цих базових технологій. Під час розробки використовуються редактори коду, системи контролю версій, збирачі, фреймворки, бібліотеки, хостинг та інші концепції.

HTML (HyperText Markup Language) представляє собою мову гіпертекстової розмітки тексту. Гіпертекст – це звичайний текст, просто у ньому є посилання до іншого тексту у ресурс контенту. В Інтернет цей термін прийшов з літератури: зміст книги і є гіпертекст, тому що містить посилання на інші сторінки. HTML – це набір правил для структурування (розмітки) текстової інформації, додавання зображень, створення таблиць, форм, списків тощо. HTML-документ – текстовий файл із розширенням .html. Документ, розмічений цим розширенням, інтерпретується браузером, де у результаті обробки користувач бачить саму Web-сторінку. Він вибудовує сам візуальний фундамент інформаційної системи, яка «запускається» самостійно. HTML вказує, де розташовуються елементи та який у них буде базовий дизайн [13].

HTML не є мовою програмування, але це мова розмітки. Він лише описує все, що знаходиться у документі: текст, картинки, посилання та відео. Описує за допомогою спеціальних слів-тегів. Теги – це ключові слова HTML, найменші будівельні блоки, з яких складається будь-яка Web-сторінка. Кожен тег означає якусь сутність. Їй може бути зображення, заголовок, список, абзац тексту тощо. Для виділення тегів серед тексту документа використовуються кутові дужки, у яких вказується ім'я тегу та його атрибути.

Теги бувають парні та непарні. Парні складаються з тегу, що відкриває і закриває, які обертають контент, дозволяючи змінювати його властивості або відображення, групувати за змістом. Всередині парних тегів можуть бути вкладені інші теги, як матрешка. Непарні теги складаються тільки з тегу, що відкриває, не містять текстового контенту і отримують контент або поведінку з атрибутів. Служать для змін властивостей документа, підключення файлів тощо.

Атрибути – це додаткові параметри тегів, за допомогою яких можна змінювати властивості та поведінку елемента. Кожен тег має обов'язкові і необов'язкові атрибути, їх може бути кілька або не бути зовсім.

У середині тегу записується атрибут, що відкриває значення, яких розташовуються всередині подвійних лапок. Декілька атрибутів поділяються пробілом. HTML-документ складається з «дерева» тегів, яке є основою будь-якого HTML-документа, як фундамент для дому.

Розмітка майбутньої Web-сторінки повинна мати сенс, як сторінка газети чи журналу. На сторінці розташовується заголовки, секції, зображення, списки, абзаци тексту тощо. Для опису цього багатства типів контенту є відповідні теги.

Семантична розмітка – підхід до створення HTML-розмітки заснований на використанні тегів відповідно до їх призначення згідно специфікації і відповідними типом контенту. Семантична розмітка означає, що теги не вибираються на основі того, як вони відображаються у браузері – вони вибираються на основі типу і структури контенту, що розмічається. Популярність HTML та використання його як основи Web-сторінок призвело до необхідності створення, підтримки та розвитку стандартів. Наразі цим займається Web Hypertext Application Technology Working Group (WHATWG), яка працює над специфікацією HTML Living Standard [13].

Для здійснення програмної реалізації інформаційної системи керуванням готелем була обрана мова розмітки HTML. Вона надає можливість наповнення Web-сторінки контентом, структурою і змістом. Мова легка та зрозуміла у використанні і має широкий попит у застосуванні Web-сторінок.

2.4.3 Формальна мова розмітки CSS

Мова для опису або для зміни зовнішнього вигляду елементів називається CSS (Cascading Style Sheets, каскадні таблиці стилів). Він використовується для позиціонування та оформлення тоді, коли HTML – для визначення семантики і структури документа [13].

Як і HTML, CSS насправді не є мовою програмування. Це мова таблиці стилів. Тобто він дозволяє застосовувати будь-які стилі вибірково до елементів у документах HTML. Наприклад, за допомогою CSS можна змінювати шрифт, колір, розмір, інтервал між рядками, розділяти вміст на колонки, а також додавати анімацію та інші декоративні елементи.

Метою CSS є відокремлення та огороження опису логічної структури Web-сторінки (яке розробляється за допомогою HTML або інших мов) від опису зовнішнього вигляду цієї Web-сторінки. Є три способи додати стилі до HTML-документу: вбудовані стилі (inline styles), вбудована таблиця стилів (embedded stylesheet) та зовнішня таблиця стилів (external stylesheet). Кожен спосіб має свої переваги, недоліки, обмеження і сферу застосування [13].

Блок CSS-коду – це правило, що складається з групи оголошень та селектора. Кожне оголошення складається з пари (властивість: значення), де властивість та його значення поділяються двокрапкою з пропуском і обов'язково закінчуються крапкою з комою. Вони описують, що треба зробити зі стилями елемента. Селектор – це ім'я HTML-елемента на початку набору правил. Він вибирає елемент(и) для застосування стилю. Для стилізації іншого елемента просто змінюється селектор.

Каскадування – це механізм, який керує кінцевими значеннями властивостей елемента, якщо до нього застосовується кілька CSS правил [13]:

- якщо до елемента застосовується кілька правил, їх властивості комбінуються;
- якщо у правилах є однакові властивості з різними значеннями, то вони конфліктують.

Для того, щоб зібрати фінальні стилі елемента та вирішити конфлікти значень властивостей, браузер використовує два механізми: специфічність та успадкування.

Для привабливого зовнішнього вигляду інформаційної системи керування готелем було обрано мову стилів CSS. Вона використовується для позиціонування та оформлення. CSS працює разом з мовою HTML, тобто

вони доповнюють один одного. Метою обраної мови є відокремлення та огороження опису логічної структури Web-сторінки від опису зовнішнього вигляду цієї Web-сторінки.

Для розробки інформаційної системи використовувалось багато технологій і програмних засобів. Перш за все, була обрана архітектура системи «клієнт-сервер». Мова програмування Java відповідає за те, щоб інформаційна система коректно працювала. До вибору Web-серверу вибір впав до Apache Tomcat, завдяки якому можна отримати необхідні ресурси та методи розгортання Web-додатків на Java. До проекту також входять база даних. Зазвичай с об'єктно-орієнтованими мовами як Java використовуються об'єктно-орієнтовані СУБД, тому було обрано PostgreSQL. Web-сторінка інформаційної системи побудована на HTML та CSS, вони відповідають за наповнення структури та змістом, а також за зовнішнім виглядом та оформленням.

3 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМ

3.1 Моделювання засобами графічної мови IDEF0

При проектуванні інформаційної системи керування готелем була обрана методологія функціонального моделювання SADT. Метод структурного аналізу та проектування (SADT) базується на використанні графічних моделей і символів для представлення різних компонентів та процесів системи. SADT є систематизованим і організованим підходом до системного аналізу та проектування, який базується на концепції декомпозиції системи на менші, більш керовані частини, які можуть бути проаналізовані і спроектовані окремо.

У SADT системи, що аналізується, представляється за допомогою ієрархії моделей, кожна з яких представляє різний рівень деталізації та абстракції. Верхній рівень ієрархії представляє систему у цілому, у той час як нижчі рівні представляють все більш детальні та специфічні аспекти системи.

IDEF0 (Integration DEFinition language) метод моделювання, що включає комбіновану графіку і текст для отримання розуміння, підтримки аналізу, надання логіки для налаштування системи, визначення вимог або підтримки проектування та інтеграцію на системному рівні. IDEF0 базується на SADT (Structured Analysis and Design Technique). Він включає у себе як визначення мови графічного моделювання (синтаксис), так і мови графічного моделювання (синтаксис і семантика), так і методологію розробки моделей [14].

IDEF0 можна використовувати для моделювання різних видів систем, включаючи підприємства, програмне забезпечення, апаратне забезпечення, продукти і так далі. Модель IDEF0 включає у себе ієрархічну серію діаграм, текст і глосарій з перехресними посиланнями один на одного. Двома основними компонентами цього методу моделювання є функції та дані/об'єкти.

Принципи IDEF0, призначені для посилення зв'язності, включають:

- діаграми, засновані на оригінальних блоках і графіці зі стрілками;
- англomовні позначення, що ідентифікують рамки і стрілки, а також глосарій і мова для визначення точних значень елементів на діаграмі;
- поступове розкриття деталей використовує ієрархічну структуру, з основними функціями на вершині та наступними етапами підфункцій, що демонструють чітке розмежування;
- «Карта вузлів» забезпечує простий індекс для пошуку інформації всередині ієрархічної структури діаграми.

Функціональний блок перетворює вхідну інформацію на вихідну. Керування визначає, коли і як це перетворення може або має відбутися. Блок A0 представляється у вигляді функціональної моделі готелю. У вхід поступають дані клієнтів, які використовують у механізмі персонал, клієнт, адміністратор, програмне забезпечення, клієнтська база і технічні засоби. Все це керується законом про права споживачів, правилами організації та списком програм. На виході отримуємо звіт і чек (рис.3.1).

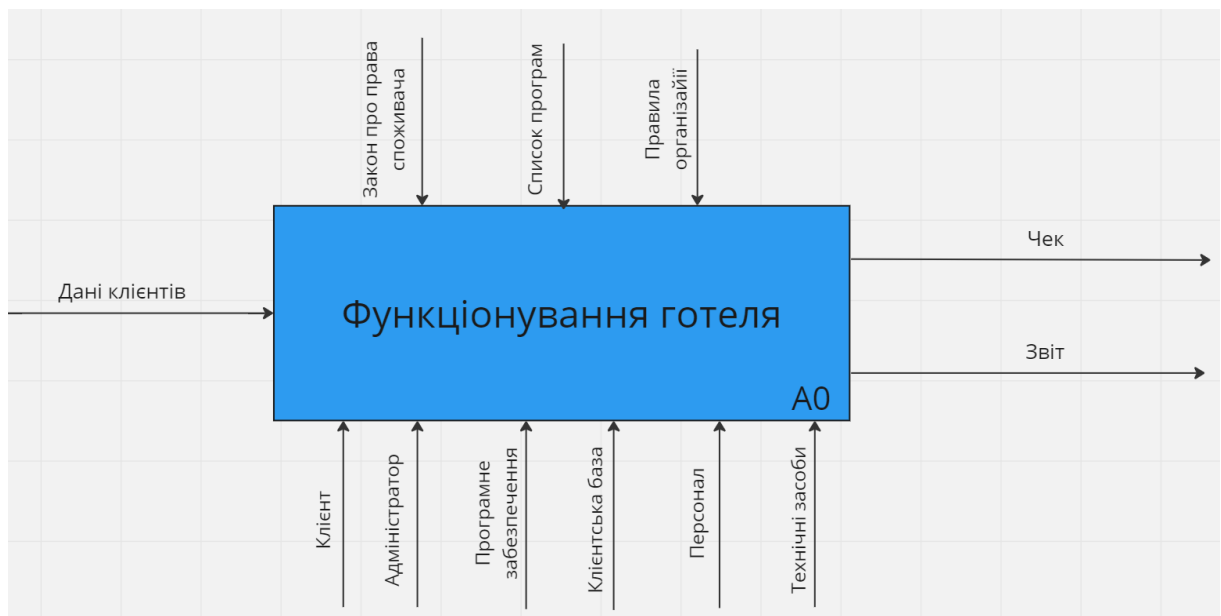


Рисунок 3.1 – Діаграма «функціонування готелю» в ІС керування готелем

У блоці A0 входять такі блоки: A1, A2, A3, які зображені на рис. 3.2.

Блок А1 є реєстрація клієнта у готелі. Де вхід представляється із себе дані клієнта, а на виході отримуємо номер кімнати. За управління цього блоку відповідає закон про права споживача та правила організації, а механізм представляє у вигляді клієнта, адміністратора та клієнтської бази. Наступним йде Блок А2, який відповідає за надання послуг. Вихід з Блоку А1 поступає на вхід Блоку А2, і це номери. Управляють ними персонал, який керується законом про права споживачів, списком програм та правилами організації. Де на виході отримуємо перелік послуг, який також є входом у Блоку А3. Він відповідає за оформлення звіту. Механізм представлений у вигляді адміністратора, програмного забезпечення та технічні засоби. За управління керуються список програм, закон про права споживача та правила організації.

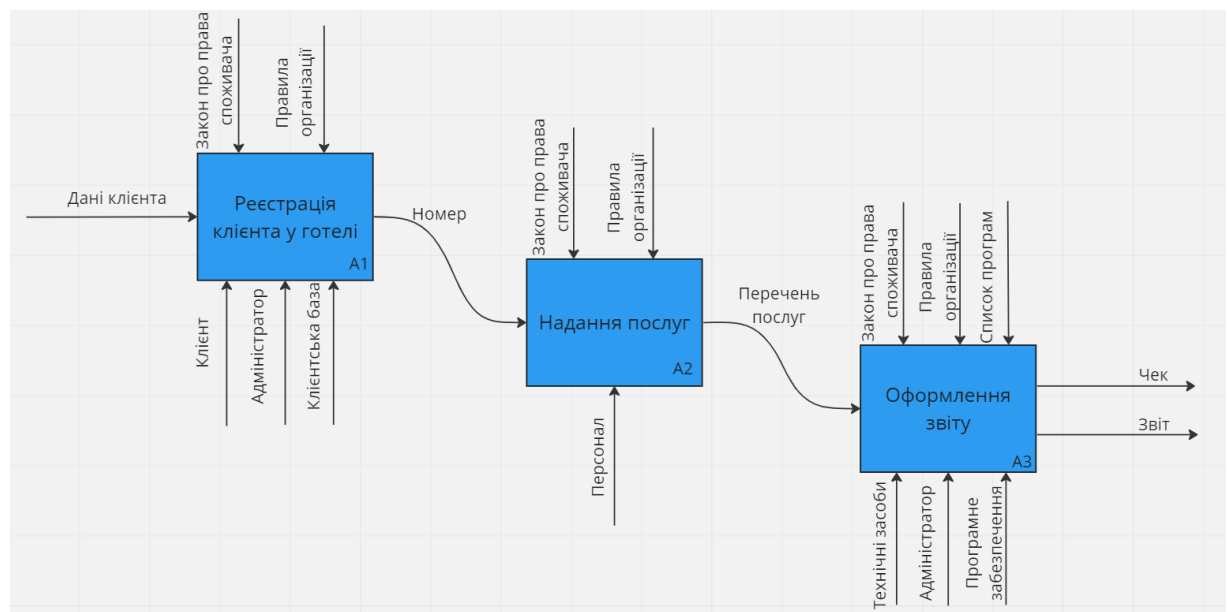


Рисунок 3.2 – Діаграми про управління в ІС керування готелем

Блок А1 включає блоки А11, А12, А13. Блок А11 відповідає за резервування номеру. Вхідною інформацією є дані клієнта. Закон про права споживача відноситься до управління, а клієнт – до механізму. На виході отримуємо список вільних номерів. Блок А12 відповідає за оформлення

поселення. На вхід поступає інформація у вигляді списку вільних номерів. Адміністратор працює з цими даними, де керується правилами організації. І на вихід отримуємо узгодження. Блок A13 відповідає за приймання передоплати. Узгодження є вхідною інформацією. Для того щоб зробити передоплату адміністратор керується правилами організації, де вихід представляється у вигляді номеру (рис. 3.3).

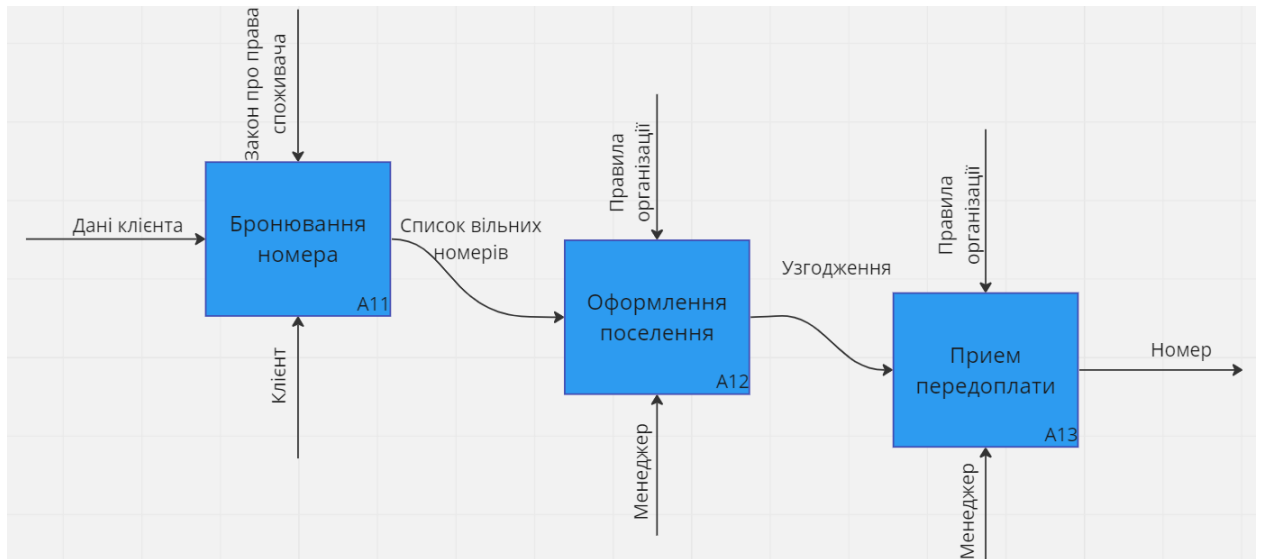


Рисунок 3.3 – Діаграми про надання номерів в ІС керування готелем

Блок A2 також має свої підрозділи. Це блок A21 та A22. Блок A21 відповідає за надання стандартних послуг. Вхід представлений як номер кімнати. Механізм – це персонал. При роботі з номерами треба керувати правилами організації та законом про права споживача. Список стандартних послуг є вихідною у Блоку A21 та вхідною інформацією у Блоку A22. Він відповідає за надання додаткових послуг. Механізм представляє з себе персонал, а управління представлено у вигляді правил організації та законом про права споживача. На виході отримуємо перелік наданих послуг (рис. 3.4).

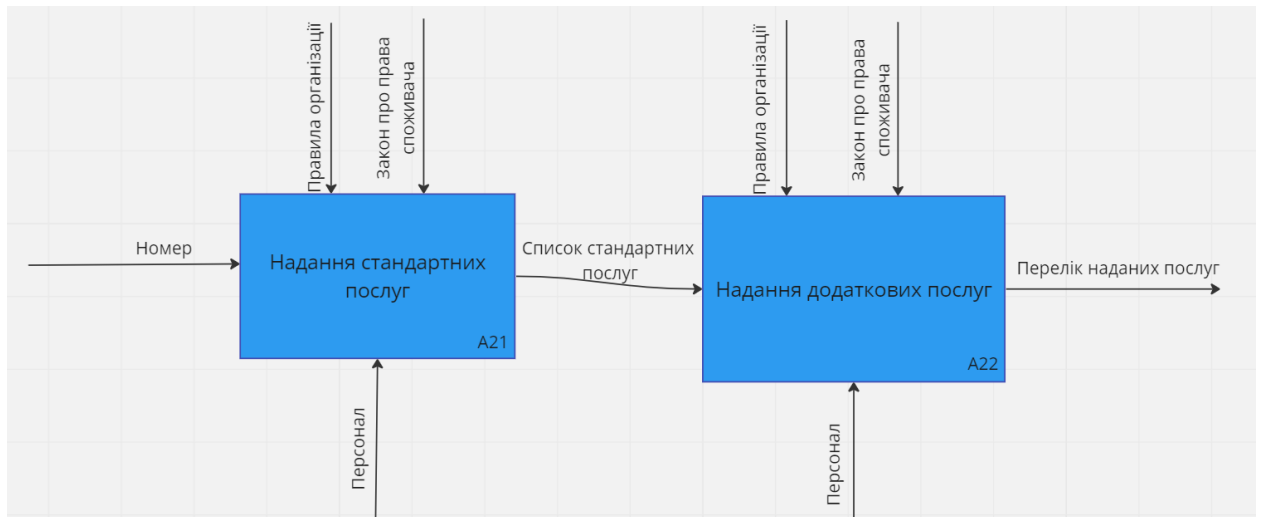


Рисунок 3.4 – Діаграми про обслуговування номерів ІС керування готелем

У блоку А3 входять блоки А31, А32. Блок А31 відповідає за обробка даних. Вхід – це перелінь послуг. Управління представляє із себе список програм з правилами організації. Механізмом: адміністратор та технічні засоби. Дані є виходом у Блоку А31 та входом у Блоці А32, який відповідає за узагальнення даних. Управління: список програм і правила організації. За механізм у цьому блоці відповідає адміністратор, технічні засоби та програмне забезпечення. На виходу отримуємо звіт та чек (рис. 3.5).

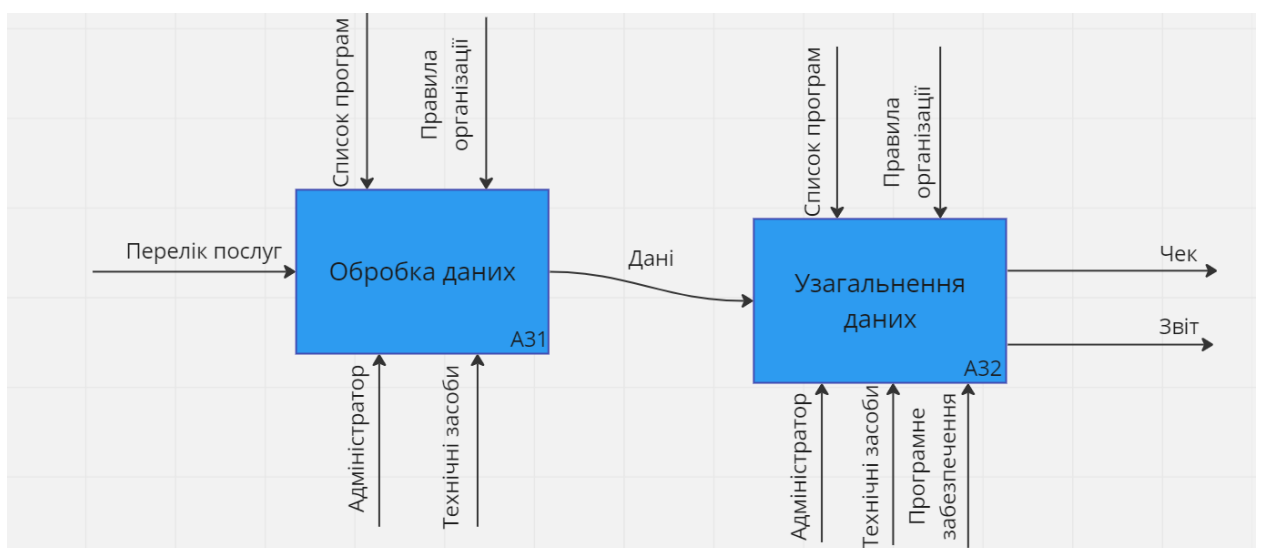


Рисунок 3.5 – Діаграми про виписку з рахунком ІС керування готелем

Моделювання інформаційної системи керуванням готелем було виконано з використанням методу структурного аналізу та проектування SADT, а саме методом моделювання IDEF0. При використанні обраних методів були спроектовані функціональні блоки керуванням готелем. Вони складаються з таких блоків: функціонування готелю, обслуговування номерів, надання номерів, керування готелем та виписки з рахунку.

3.2 Проектування інформаційної системи за допомогою діаграми потоків даних DFD

Діаграма потоків даних (DFD) відображає потік інформації для будь-якого процесу або системи. Вона використовує певні символи, такі як прямокутники, кола і стрілки, а також короткі текстові мітки, щоб показати вхідні та вихідні дані, точки зберігання і маршрути між кожним пунктом призначення. Схеми потоків даних можуть варіюватися від простих, навіть намальованих від руки оглядів процесів, до глибоких, багаторівневих DFD, які поступово заглиблюються у те, як обробляються дані. Їх можна використовувати для аналізу існуючої системи або моделювання нової. Як і всі найкращі діаграми та графіки, DFD часто можуть візуально «сказати» речі, які важко пояснити словами, і вони працюють як для технічної, так і для нетехнічної аудиторії, від розробника до генерального директора [13].

Процес створення діаграми потоків даних складається з чотирьох основних компонентів:

- зовнішній об'єкт;
- процес;
- сховище даних;
- потік даних.

Контекстна діаграма потоків даних описує такий процес, як управління готелем (рис. 3.6).

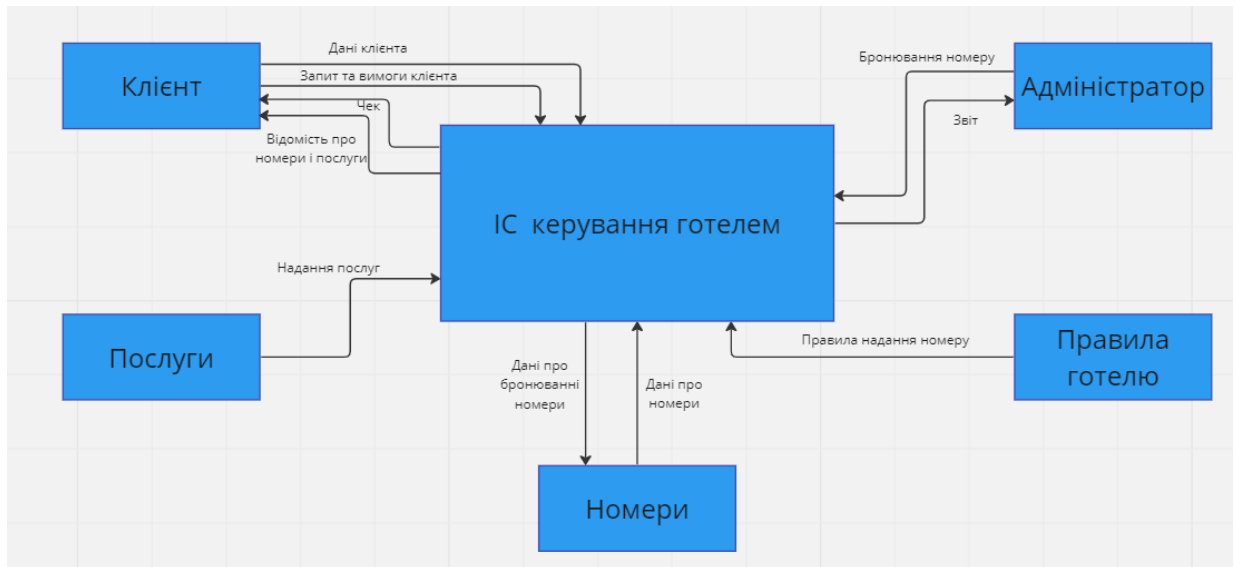


Рисунок 3.6 – Контекстна діаграма потоків даних ІС керуванням готелем

Зовнішній об'єкт представлений у вигляді клієнта. Зовнішня система, яка надсилає або отримує дані, взаємодіючі з системою, що діагностується. Вони є пунктами призначення інформації, що надходить до системи або виходить з неї. У даному випадку зовнішній об'єкт представлений у вигляді фізичної особи.

Сховище даних для даної діаграми є послуги, номери та правила готелю. Це файли чи сховища, де зберігається інформація для подальшого використання. Стрілки діаграми показують як саме об'єкти з даними працюють від однієї до іншої роботи. Тобто запит клієнта додає процедуру до обробки запиту, де дану процедуру дивиться та виконує адміністратор. Він просить дані про номері готелю (які беруться зі сховища даних), потім, де адміністратор узгоджується з правилами готелю про надання номеру. Наприкінці він повідомляє клієнтові про номер, якій є зайнятим або вільним. У випадку бронювання номеру дані оновлюється про обраний номер і заноситься дані про зарезервованій номер до сховища даних про номера готелю.

Дана модель складається з наступних процесів:

- реєстрація клієнтів;

- пошук необхідного номеру;
- бронювання номеру;
- оформлення заселення;
- надання стандартних та додаткових послуг;

Також у декомпованій діаграмі є такі потоки даних:

- дані про клієнтів;
- дані про номери;
- правила заселення клієнтів;
- правила надання номеру;
- дані про номери;
- звіти та чеки;
- дані про послуги.

Більш детально процес керуванням готелем показано на декомпованій діаграмі потоків даних. Описується реєстрація клієнта, оформлення запиту, оформлення бронювання, надання стандартних та додаткових послуг, оформляється звіт для готелю та чек для клієнта (рис. 3.7).

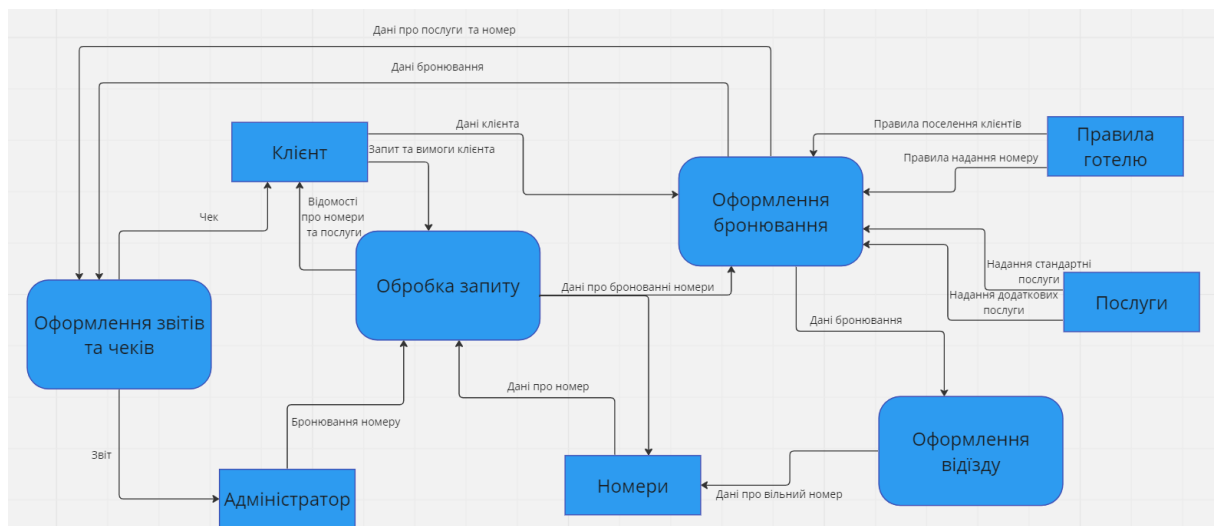


Рисунок 3.7 – Деталізована діаграма потоків даних ІС керуванням готелем

Правила готелю, нормативні документи, дані про клієнта та дані про номери представлено у вигляді сховища даних.

Процес управління готелем виконує наступні кроки:

- все починається з того, що клієнт робить запит та ставить свої вимоги до перебування у готелі;
- адміністратор у свою чергу обробляє запит і після чого повідомляє клієнтові необхідну інформацію про номери та послуги;
- якщо клієнт знайшов відповідний за його бажанням номер та послуги він оформлює резервування номеру, в іншому випадку відвідувач відмовляється від бронювання. Адміністратор реєструє клієнта, який надав всі потрібні дані для оформлення;
- адміністратор оформлює бронювання, керуючись з правилами заселення клієнта та наданням номеру для даного готелю;
- після бронювання адміністратор робить чек, який надається клієнтові. Також дані про бронювання він оновлює у сховище даних;
- після того, як клієнт був деякий час у готелі виїжджає, і адміністратор робить оновлення даних про номери, що даний номер вільний.

У даному підрозділі проектування інформаційної системи керуванням готелем було здійснено завдяки діаграмі потоків даних DFD, яка показує потік інформації для системи або процесу. При розробці проекту були зроблені контекстна та деталізована діаграми потоків даних інформаційної системи керуванням готелем.

3.3 Проектування та створення бази даних

Для того, щоб уявити логічну структуру інформаційної системи керуванням готелем була використана модель представлення даних «сутність-зв'язок». Модель «сутність-зв'язок» базується на важливій семантичній інформації про реальний світ і призначена для логічного

представлення даних. Вона визначає значення даних у контексті їх взаємозв'язку з іншими даними.

Сутність (entity) – це об'єкт, який можна ідентифікувати певним чином, що відрізняє його від інших об'єктів. Також сутність представляє собою набір атрибутів, які описують властивості всіх членів певної множини сутностей. Домен – це множина значень (тобто область видимості) атрибута. Асоціація, яка встановлена між кількома сутностями називається зв'язком.

Для розробки бази даних було створено одинадцять таблиць, які називаються: `room_types` (типи кімнат), `rooms` (кімнати), `customers` (відвідувачі готелю), `booking` (бронювання), `payment` (оплата), `services` (послуги), `roles` (ролі), `users` (користувачі), `users_roles` (користувачі_ролі), `employees` (працівники) та `status` (статус). Де кожна таблиця представляє із себе сутність, яка повинна містити атрибут або групу атрибутів, які будуть однозначно ідентифікувати кожен екземпляр сутності. Такий атрибут називають первинним ключем.

При проектуванні інформаційної системи управління готелем враховувалися сутності та їх атрибути.

Сутність «`room_types`» (типи кімнат) містить такі атрибути як: `id` та `name_type` (назва типу), які можна побачити у таблиці 3.1:

Таблиця 3.1 – Сутність «`room_types`» (типи кімнат)

№	Атрибут	Тип
1	<code>id</code>	<code>int</code>
2	<code>name_type</code>	<code>varchar(30)</code>

Сутність «`rooms`» (кімнати) містить такі атрибути як: `id`, `number_room` (номер кімнати), `count_of_days` (кількість д), `room_type_id` (`id` типа кімнати), `id_status` (`id` статусу), `date_in` (день заїзду), `date_out` (день виїзду), `id_customer`

(id відвідувача), adults (дорослі) та kids (діти), які можна побачити у наведеній таблиці 3.2:

Таблиця 3.2 – Сутність «rooms» (кімнати)

№	Атрибут	Тип
1	id	int
2	number_room	int
3	count_of_days	int
4	room_type_id	int
5	id_status	int
6	date_in	date
7	date_out	date
8	id_customer	int
9	adults	int
10	kids	int

Сутність «services» (послуги) містить наступні атрибути: id, name_service(назва послуги) та price (ціна), які можна подивитися у наведеній таблиці 3.3:

Таблиця 3.3 – Сутність «services» (послуги)

№	Атрибут	Тип
1	id	int
2	name_service	varchar(15)
3	price	int

Сутність «payment» (оплата) містить такі атрибути як: id, number_room (номер кімнати), id_customer (id відвідувача), sum (сума), id_status (id статусу), date_of_payment (день оплати) та the_last_day_of_payment (останній день оплати), які можна побачити у таблиці 3.4:

Таблиця 3.4 – Сутність «payment» (оплата)

№	Атрибут	Тип
1	id	int
2	number_room	int
3	id_customer	int
4	sum	int
5	id_status	int
6	date_of_payment	date
7	the_last_day_of_payment	date

Сутність «users» (користувачі) містить такі атрибути як: id, user_name (ім'я користувача), login (логін) та password (пароль), які можна побачити у таблиці 3.5:

Таблиця 3.5 – Сутність «users» (користувачі)

№	Атрибут	Тип
1	id	int
2	user_name	varchar(30)
3	login	varchar(20)
4	password	varchar(20)

Сутність «employees» (робітники) містить такі атрибути як: id, first_name (ім'я), last_name (прізвище), gender (під), date_of_birth (дата народження), address (місце проживання), job_department (відділ роботи), email (електрона пошта), phone_number (номер телефону) та salary (заробітна плата), які можна побачити у таблиці 3.6.

Таблиця 3.6 – Сутність «employees» (робітники)

№	Атрибут	Тип
1	id	int
2	first_name	varchar(20)
3	last_name	varchar(20)
4	gender	varchar(15)
5	date_of_birth	date
6	address	varchar(50)
7	job_department	varchar(15)
8	email	varchar(50)
9	phone_number	varchar(15)
10	salary	int

Сутність «booking» (бронювання) містить такі атрибути як: id, number_order (номер заказу), organization (організація), id_customer (id відвідувача), phone_number (номер телефону), email (електрона пошта), extra_customers_1 (додатковий відвідувач 1), extra_customers_2 (додатковий відвідувач 2), extra_customers_3 (додатковий відвідувач 3), adults (дорослі), kids (діти), type_room_id (id типа кімнати), number_room (номер кімнати), additional_service (додаткові послуги), tariff (трафік), price_of_services (ціна

послуг), discount (знижка) та amount (всього), які можна побачити у наведеній таблиці 3.7:

Таблиця 3.7 – Сутність «booking» (бронювання)

№	Атрибут	Тип
1	id	int
2	number_order	int
3	organization	varchar(30)
4	id_customer	int
5	phone_number	varchar(15)
6	email	varchar(50)
7	extra_customers_1	varchar(100)
8	extra_customers_2	varchar(100)
9	extra_customers_3	varchar(100)
10	adults	int
11	kids	int
12	type_room_id	int
13	number_room	int
14	additional_service	varchar(20)
15	tariff	int
16	price_of_services	int
17	discount	int
18	amount	int

Сутність «roles» (ролі) містить такі атрибути як: id та role_type (тип ролі) , які можна побачити у таблиці 3.8:

Таблиця 3.8 – Сутність «roles» (ролі)

№	Атрибут	Тип
1	id	int
2	role_name	varchar(20)

Сутність «customers» (відвідувачі готелю) містить такі атрибути як: id, first_name (им'я), last_name (прізвище), gender (рід), phone_number (номер телефону), email (електрона пошта), nationality (національність), type_passport (тип паспорту), number_passport (номер паспорта), address (адрес) та patronymic (по-батьківськи), які можна побачити у таблиці 3.9:

Таблиця 3.9 – Сутність «customers» (відвідувачі готелю)

№	Атрибут	Тип
1	id	int
2	first_name	varchar(20)
3	last_name	varchar(20)
4	gender	varchar(15)
5	phone_number	varchar(15)
6	email	varchar(50)
7	nationality	varchar(15)
8	type_passport	varchar(15)
9	number_passport	varchar(15)
10	address	varchar(50)
11	patronymic	varchar(30)

Сутність «users_roles» (користувачі-ролі) містить такі атрибути як: user_id (id користувача) та role_id (id роль) , які можна побачити у наведеній таблиці 3.10:

Таблиця 3.10 – Сутність «users_roles» (користувачі-ролі)

№	Атрибут	Тип
1	user_id	int
2	role_id	int

Сутність «status» (статус) містить такі атрибути як: id та status_name (назва статусу) , які можна побачити у таблиці 3.11:

Таблиця 3.11 – Сутність «status» (статус)

№	Атрибут	Тип
1	id	int
2	status_name	varchar(20)

Після того, як були описані всі атрибути та сутності для розробки бази даних, для інформаційної системи керуванням готелем, можна визначити зв'язки між усіма сутностями. При створенні моделі «сутність-зв'язок» використовується декілька зв'язків, а саме: «один-до-багатьох» та «один-до-одного». Діаграма складається з одинадцяти таблиць, які називаються: room_types (типи кімнат), rooms (кімнати), customers (відвідувачі готелю), booking (бронювання), payment (оплата), services (послуги), roles (ролі), users (користувачі), users_roles (користувачі_ролі), employees (працівники) та status (статус). Дану діаграму можна подивитися на рис. 3.8.

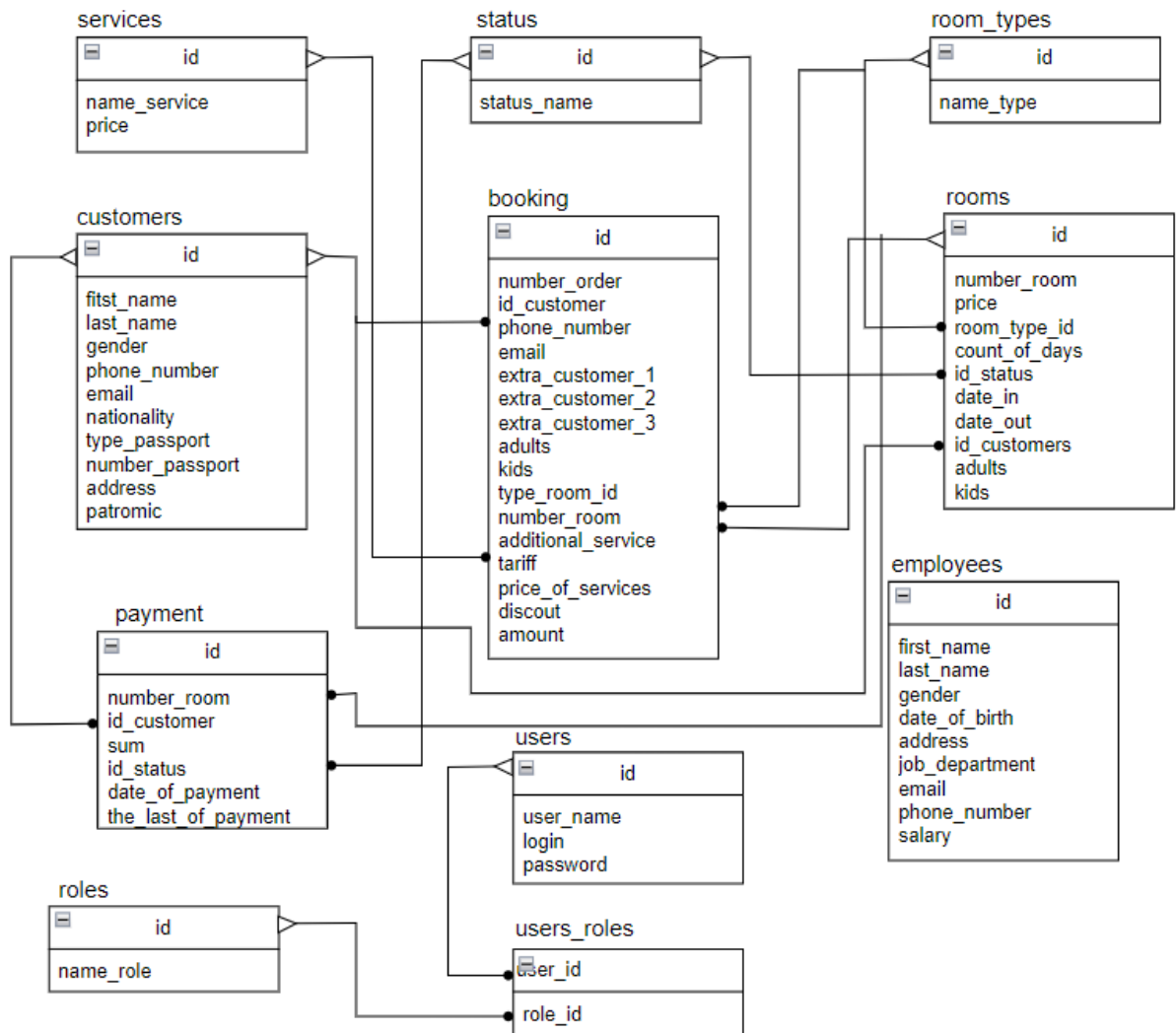


Рисунок 3.8 – Модель «сутність-зв'язок» БД ІС управління готелем

База даних розробляється для того, щоб зберігати важливу інформацію та які будуть доступні після перезавантаження програми. Інформація найчастіше і є найціннішим, що є у компанії тому база даних є невід'ємною частиною будь-якого проекту. У даному підрозділі проектувалася база даних, яка складається з одинадцяти таблиць. Кожна таблиця має певну роль у розробці інформаційної системи управлінням готелем. База даних створюється для того, щоб керувати інформацією (додавати, змінювати, виділяти або дивитися).

При розробці використовувались такі засоби моделювання, як IDEF0, DFD та модель «сутність-зв'язок». Кожна з них показує певні системи та

процеси для створені даної інформаційної системи. При використанні IDEF0 моделі були побудовані діаграми функціонування ІС керування готелем. Використання діаграма потоків даних відображає процес управління системою.

4 ПРАКТИЧНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

Дана інформаційна система керуванням готелем називається «Annetta-Travel», яка використовується для більш ефективної роботи працівників даного готелю. Співробітники можуть виконувати такі дії, як: реєстрування клієнта, розроблення звіту, перегляд списку кімнат тощо. У цій інформаційній системі розглядаються три види користувачів: менеджер, бухгалтер та адміністратор. Кожен з них має свої права та обов'язки.

4.1 Інтерфейс користувача-менеджера

У цьому підрозділі буде розглядатись користувач-менеджер. Він відповідає за бронювання та реєстрацію клієнтів, тобто він має доступ до таких вкладок, як:

- реєстрація;
- бронювання;
- календар кімнат;
- список вільних кімнат.

При першому використанні даної системи користувачу треба авторизуватися. Для цього першою сторінкою яку він бачить є «Авторизація». Вона у свою чергу складається з наступних елементів: логотипу, назви «Авторизація», поле «Логін», поле «Пароль» та кнопки «Вхід». Сторінка авторизація з'являється одразу після запуску проекту.

Процедура авторизації складається з заповнення таких полів, як: логін та пароль. Без знання цих пунктів будь-який користувач не зможе потрапити до головної сторінки, де він буде працювати. Після того, як були введені вірні логін та пароль користувач натискає на кнопку «Вхід» для входу у систему. Сторінку авторизації можна подивитися на рис. 4.1.

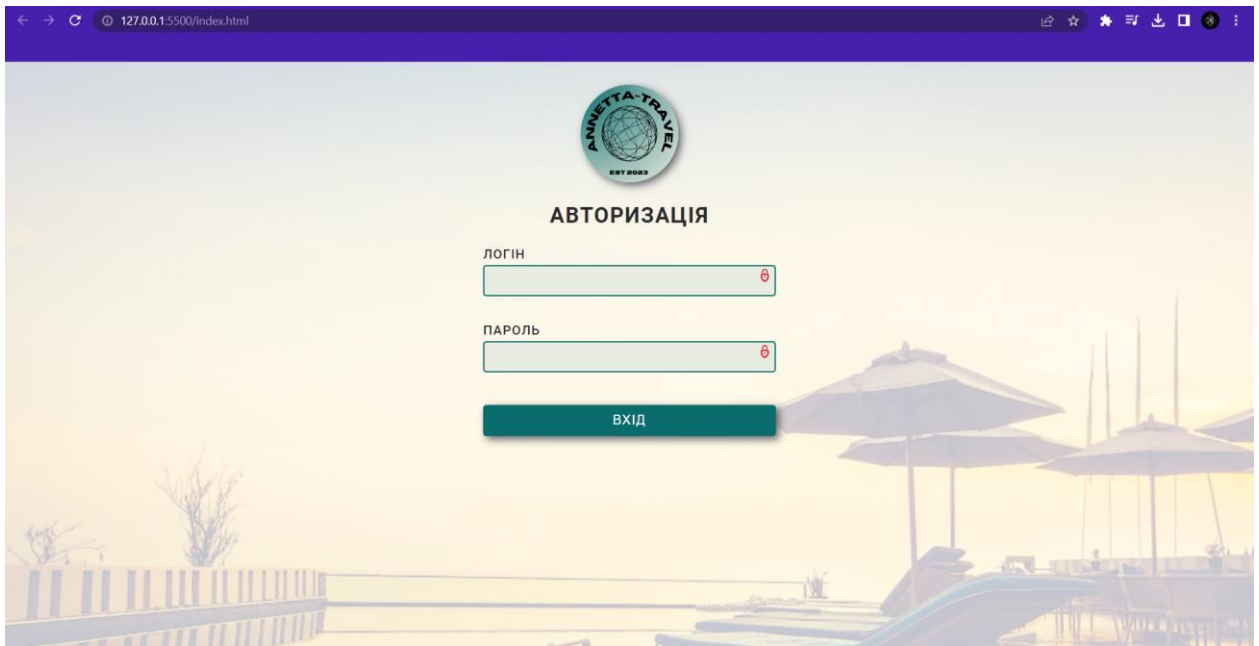


Рисунок 4.1 – Сторінка авторизації

Після того, як користувач-менеджер авторизувався у даній інформаційній системі він потрапляє до головної сторінки. Вона складається з таких елементів:

- логотипу «Annetta-Travel»;
- кнопки «Вихід»;
- надпису «Менеджер»;
- кнопки «реєстрація клієнтів»;
- кнопки «бронювання»;
- кнопки «календар»;
- кнопки «список вільних кімнат».

Інтерфейс менеджера можна подивитися на рис. 4.2. Натиснувши на одну з вище перерахованих кнопок користувач може вийти з системи або перейти до інших сторінок, а саме до:

- реєстрації відвідувачів готелю;
- бронюванню номеру;
- перегляду календарю;

– списку вільних кімнат.

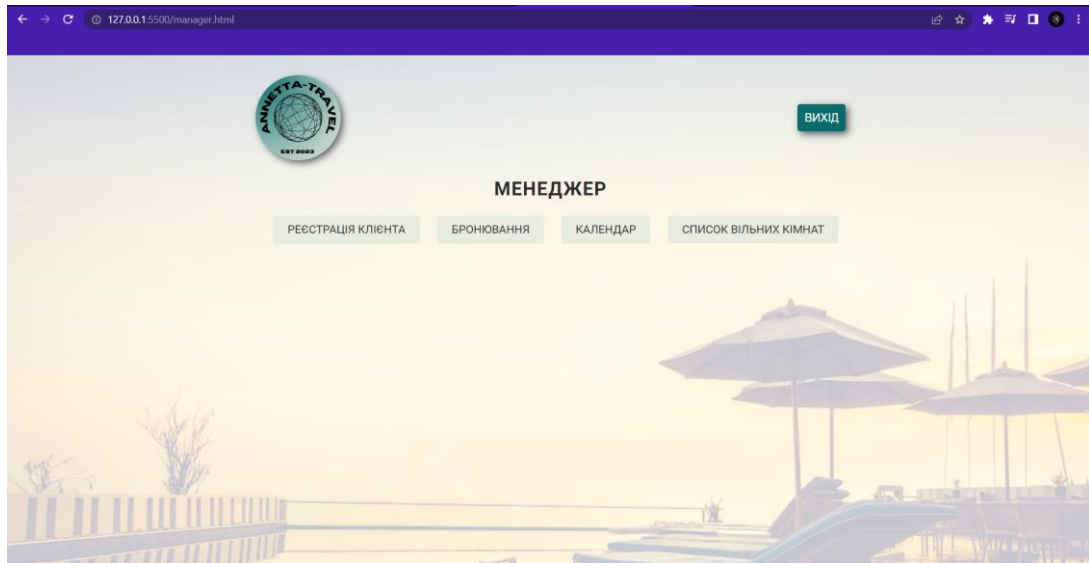


Рисунок 4.2 –Інтерфейс менеджера

Далі буду розглядатися сторінки, які доступні менеджеру. Він може керувати наступними вкладками:

- реєстрація клієнтів;
- бронювання;
- календар;
- список вільних кімнат.

Першою сторінкою буде розглянута «реєстрація клієнтів». У даній вкладці користувач може працювати даними клієнтів. Тобто додавати його до інформаційної системи, чи редагувати дані, чи виділити зовсім. Основними складовими є логотип та кнопки для переходу на інші сторінки. Реєстрація клієнта складається з фотографії та наступних полів: організація, прізвище, ім'я, по-батькові, дата народження, стать, громадянство, тип паспорту, номер паспорту, адреса проживання, телефон та електронна пошта. Заповнивши всі поля користувач натискає кнопку «додати» і клієнт з'явиться у базі даних інформаційної системи. Після того, як зареєстрували відвідувача можна буде оновлювати його дані або виділити. Для того, щоб

виділити треба лише заповнити такі поля: прізвище, ім'я, по-батькові та натиснути кнопку «виділити» (рис. 4.3).

Рисунок 4.3 – Сторінка реєстрації клієнта

Наступною вкладкою у менеджера йде «бронювання». Бронювання має наступні поля: номер замовлення, організація, основний гість, телефон, електронна пошта, три додаткових гостя, дата заїзду, дата виїзду, кількість ночей, дорослих, дітей, категорія кімнати, номер кімнати, додаткові послуги, тариф за номер, вартість послуг, знижка та все разом (рис. 4.4).

Рисунок 4.4 – Сторінка бронювання

Наступною у черзі буде розглядатися вкладка «календар». Вона представляє з себе календар на кожен день місяця (у даному випадку був травень) з номерами. Сторінка показує який номер зайнятий або заборонований, оплачений чи неоплачений. Червоним кольором помічені номери, які вже зайняті клієнтами, а зеленими – заброньовані. Заштриховані ділянки є оплачені, а незаштриховані відповідно неоплачені. У календарі по горизонталі розташовані дні місяця, по вертикалі – номери кімнат (рис. 4.5).

номери	травень																														
	01 пн	02 вт	03 ср	04 чт	05 пт	06 сб	07 нд	08 пн	09 вт	10 ср	11 чт	12 пт	13 сб	14 нд	15 пн	16 вт	17 ср	18 чт	19 пт	20 сб	21 нд	22 пн	23 вт	24 ср	25 чт	26 пт	27 сб	28 нд	29 пн	30 вт	31 ср
Одномісні																															
101		Захарченко										Шадрина																			
102			Бочаров													Колесникова															
103										ремонт																					
Двомісні																															
201				Чернобров															Миклушевский				Максимова								
202		Тичина										Захаров																			
203																															
Тримісні																															
301		Деркач																													
302																ремонт															
303						ремонт																									

Рисунок 4.5 – Сторінка календарю

Останньою вкладкою у користувача-менеджера є список вільних кімнат. Менеджер, працюючи на стойці реєстрації, спілкується з клієнтами. Вони розповідають про свої запити, а менеджер у свою чергу обробляє запити відвідувачів та відповідає їм. Тому ця сторінка призначення для клієнтів. Тобто адміністратор настраює запит клієнтові (наприклад, дату прибуття, кількість людей у кімнаті) у налаштуваннях та натискає кнопку пошуку. У виході з'являється відповідні номери у вигляді списку. До кожної кімнати йде фотографія, опис, вартість за добу та кнопка «забронювати». Адміністратор показує дані кімнати клієнтам, де вони дивляться результат (рис. 4.6).

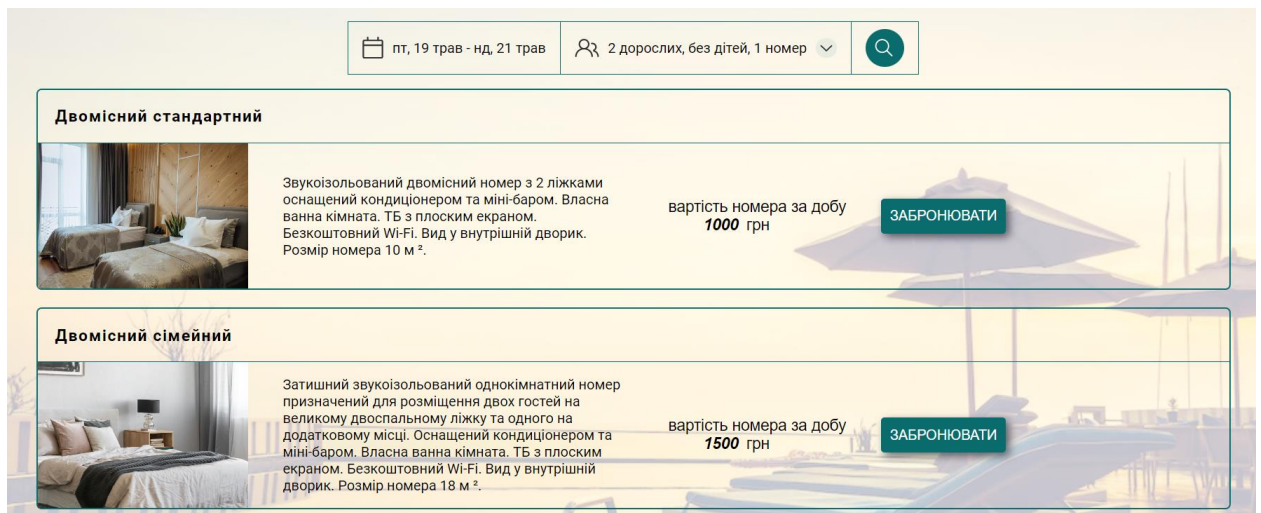


Рисунок 4.6 – Сторінка вільних кімнат

Було розглянуто всі дії та сторінки інформаційної системи керуванням готелем з правами доступу користувач-менеджер. У даному проекті він знаходиться на стойці реєстрації та виконує бронювання та реєстрацію відвідувачів готелю.

4.2 Інтерфейс користувача-бухгалтера

У цьому підрозділі буде розглядатись користувач-бухгалтер. Він відповідає за надходження платежів, розроблення звітів та чеків. Інтерфейс бухгалтера складається з наступних елементів:

- логотип;
- кнопка «вихід»;
- надпис «Бухгалтер»;
- вкладка клієнти;
- вкладка кімнати;
- вкладка оплата.

Інтерфейс даного користувача можна подивитися на рис. 4.7.

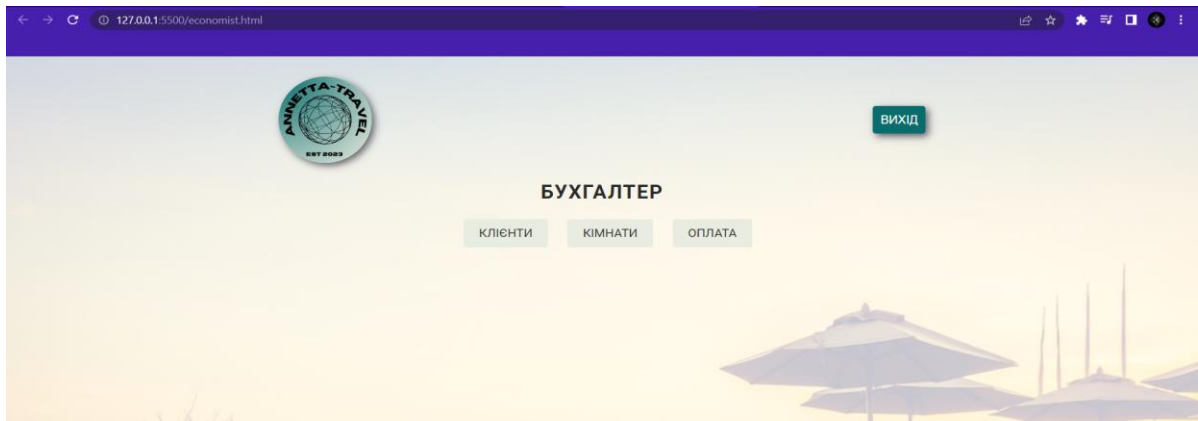


Рисунок 4.7 – Інтерфейс бухгалтера

Першою вкладкою є список клієнтів. Бухгалтер користується даним списком для того, щоб скласти чек та звіт у сторінці оплата. Таблиця включає у себе такі стовбці, як: id, прізвище, ім'я, стать, статус, номер, дата заїзду, дата виїзду, кількість ночей, дата народження, телефон та електронна пошта. Також дана таблиця може редагуватися за потребою, якщо використовувати кнопки, які знаходяться під таблицею (рис. 4.8).

id	прізвище	ім'я	стать	статус	номер	дата заїзду	дата виїзду	кіль. ноч	дата народж.	телефон	email
1	Бочаров	Володимир	ч		102	02.05.2023	08.05.2023	6	15.02.1969	+380675264871	bocharov@gmail.com
2	Деркач	Іван	ч		301	01.05.2023	07.05.2023	6	16.05.1979	+380963609015	derkach@gmail.com
3	Захаров	Віктор	ч		202	12.05.2023	15.05.2023	3	01.03.1982	+380680023401	zacharov@gmail.com
4	Захарченко	Деніс	ч		101	01.05.2023	05.05.2023	4	18.01.2000	+380972653410	zaharchenko@gmail.com
5	Колесникова	Олена	ж		102	15.05.2023	19.05.2023	4	25.06.1980	+380663695842	kolesnikova@gmail.com
6	Максимова	Жанна	ж		201	24.05.2023	28.05.2023	4	13.04.1985	+380933658412	maksimova@gmail.com
7	Миклушевский	Кирило	ч		201	18.05.2023	22.05.2023	4	02.11.1976	+380684875269	miklushev@gmail.com
8	Тичина	Антон	ч		202	01.05.2023	05.05.2023	4	30.01.1990	+380677771405	tychina@gmail.com
9	Чернобров	Олексій	ч		201	04.05.2023	10.05.2023	6	20.05.1989	+380632565498	chernobrov@gmail.com
10	Шадріна	Іріна	ж		101	11.05.2023	15.05.2023	4	14.03.1965	+380639436108	shadrina@gmail.com

Рисунок 4.8 – Сторінка списку усіх клієнтів

Наступною сторінкою розглядатися буде список кімнат. Дана сторінка містить діапазон у вигляді періоду від початкової дати до кінцевої дати. У даному випадку була обрана дата з періодом від 01.05.2023 року до 31.05.2023. Після того, як користувач задав період перегляду кімнат, він може дивитися таблицю або змінювати її. Таблиця допомагає бухгалтеру у складанні чеків та звітів. Дана таблиця складається з наступних стовбців: id, номер, категорія, прайс, статус, дата заїзду, дата виїзду, кількість ночей, клієнт, дорослі, діти та головний клієнт (рис. 4.9).

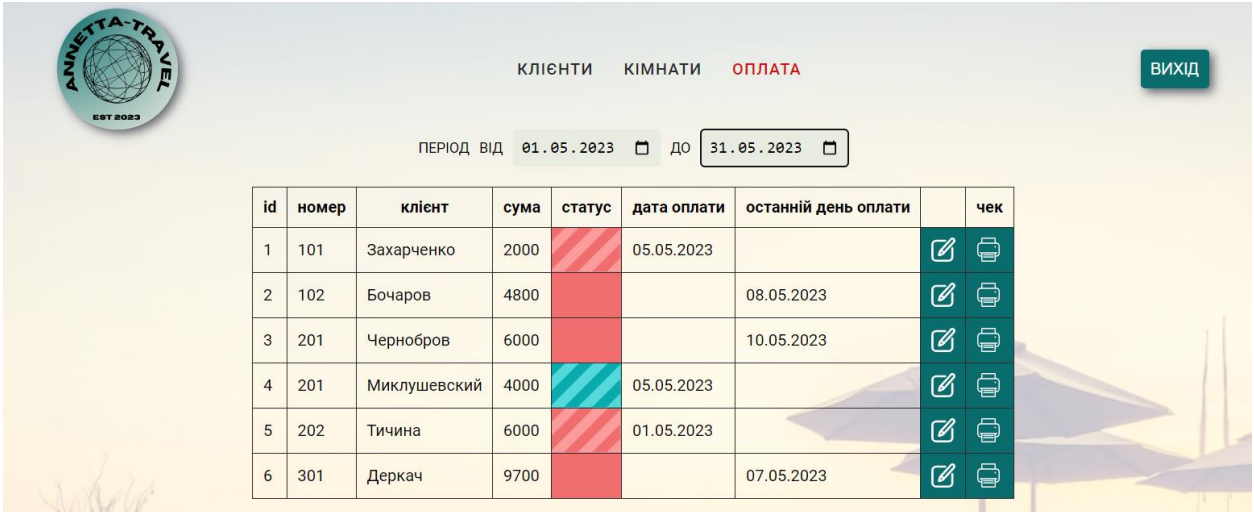
ПЕРІОД ВІД 01.05.2023 ДО 31.05.2023

id	номер	категорія	прайс	статус	дата заїзду	дата виїзду	кіл. ноч	клієнт	дорослі	діти	
1	101	Одномісний стандартний	500		01.05.2023	05.05.2023	4	Захарченко	1	0	
2	101	Одномісний стандартний	500		11.05.2023	15.05.2023	4	Шадріна	1	0	
3	102	Одномісний сімейний	800		02.05.2023	08.05.2023	6	Бочаров	1	0	
4	102	Одномісний сімейний	800		15.05.2023	19.05.2023	4	Колесникова	1	1	
5	103	Одномісний люкс	1000		08.05.2023	14.05.2023					
6	201	Двомісний стандартний	1000		04.05.2023	10.05.2023	6	Чернобров	2	0	
7	201	Двомісний стандартний	1000		18.05.2023	22.05.2023	4	Миклушевський	2	0	
8	201	Двомісний стандартний	1000		24.05.2023	28.05.2023	4	Максимова	2	0	
9	202	Двомісний сімейний	1500		01.05.2023	05.05.2023	4	Тичина	2	0	
10	202	Двомісний сімейний	1500		12.05.2023	15.05.2023	3	Захаров	2	1	
11	301	Тримісний стандартний	1500		01.05.2023	07.05.2023	6	Деркач	2	1	
12	302	Тримісний сімейний	2000		15.05.2023	21.05.2023					
13	303	Тримісний люкс	2500		01.05.2023	07.05.2023					

Рисунок 4.9 – Сторінка списку усіх заброньованих кімнат

Далі на черзі йде сторінка оплата. Вона виконує важливу роль у даного користувача та у готельного-бізнесу у цілому. Бухгалтер спочатку обирає необхідний період з яким буде працювати. Потім він може змінювати дані, робити звіт або друкувати чек для клієнта, який вже оплатив бронювання. Таблиця складається з таких стовбців: id, номер кімнати, клієнт, сума, статус,

дата оплати, останній день оплати та чек. Також за необхідністю є можливість редагувати дані (рис. 4.10).



id	номер	клієнт	сума	статус	дата оплати	останній день оплати	чек
1	101	Захарченко	2000	Red	05.05.2023		✎ 🖨
2	102	Бочаров	4800	Red		08.05.2023	✎ 🖨
3	201	Чернобров	6000	Red		10.05.2023	✎ 🖨
4	201	Миклушевский	4000	Green	05.05.2023		✎ 🖨
5	202	Тичина	6000	Red	01.05.2023		✎ 🖨
6	301	Деркач	9700	Red		07.05.2023	✎ 🖨

Рисунок 4.10 – Сторінка оплати

Було розглянуто всі дії та сторінки інформаційної системи керуванням готелем з правами доступу користувач-бухгалтер. Він відповідає за надходження платежів. Також має доступ до таких сторінок, як: список клієнтів, кімнат та оплати.

4.3 Інтерфейс користувача-адміністратора

У цьому підрозділі буде розглядатись користувач- адміністратор. Він має повний доступ до всіх операцій. Тобто адміністратор має права менеджера та бухгалтера разом, що він може робити наведено ніше:

- редагувати або дивитися список кімнат;
- робити бронювання;
- реєструвати клієнта;
- робити звіт та чек;
- дивитися список клієнтів та редагувати його;

- працювати з календарем резервувань;
- працювати з працівниками;
- скласти список, які кімнати вільні на обрану дату.

Розглядаючи інтерфейс адміністратора можна розгледіти такі складові: логотип, надпис «Адміністратор», кнопка «Вихід» та вкладки (такі, як реєстрація клієнта, бронювання, календар, клієнти, кімнати, оплата, список вільних кімнат та працівники) (рис. 4.11).

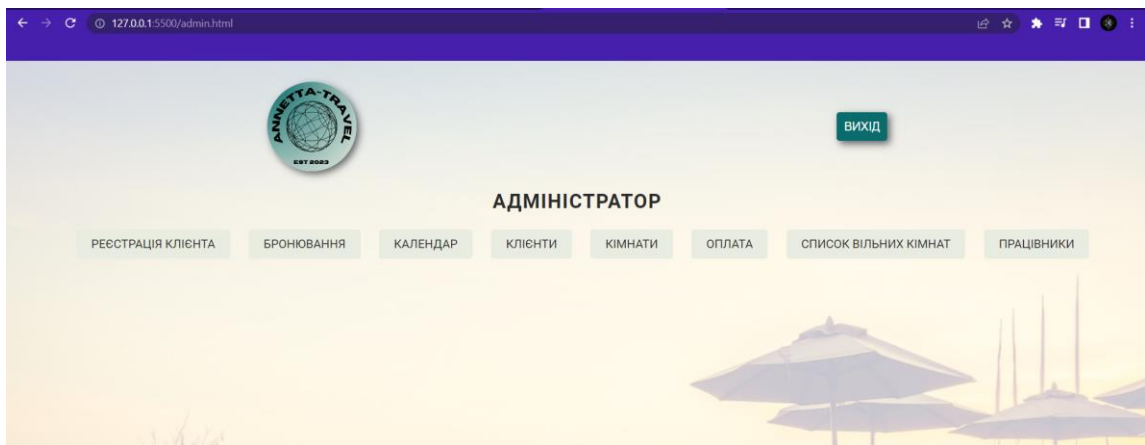


Рисунок 4.11 – Інтерфейс адміністратора

Було коротко розглянуто всі дії та сторінки інформаційної системи керуванням готелем з правами доступу користувач-адміністратор, який має доступ до всіх процесів даної інформаційної системи. Він має право працювати як менеджер, так і як бухгалтер, а також права до редагування працівників у системі.

Отже, даний розділ був присвячений практичній реалізації інформаційній системі керуванням готелем. Розділ складається з того, що працювати у даному проекті можуть користувачі з певними правами доступу. Інформаційна система включає у себе три різновиди цих повноважень, а саме: менеджер, бухгалтер та адміністратор.

Кожен з користувачів має певні права та обов'язки. Менеджер у даному проекті виконує такі обов'язки, як: реєстрація клієнта та бронювання

номерів. Бухгалтер працює з платіжками, тому він має доступ до перегляду списків кімнат та клієнтів. Адміністратор має більш за усіх права. Тобто він може керувати списками кімнат або клієнтів, робити бронювання, оформляти клієнтів, а також редагувати платіжки. Даний проект був створений саме для того, щоб їм могли ефективно користуватися в управлінні готельним бізнесом.

ВИСНОВКИ

У результаті виконання дипломної роботи було здійснено проектування та програмна реалізація спеціалізованої інформаційної системи для керування готелем, яка є зручним і ефективним засобом управління для персоналу всіма бізнес-процесами готельного підприємства.

Інформаційна система управління готелем надає змогу всім категоріям користувачів здійснювати, в залежності від прав доступу до інформації, реєстрацію, бронювання номерів, перегляд гостей готелю, кімнат, зроблених платежів, перегляд списку усіх працівників. У системі визначені передбачені наступні категорії користувачів: адміністратор, бухгалтер, менеджер.

В результаті виконаного проектування інформаційної системи керування готелем проведено аналіз існуючих систем та визначені основні переваги та недоліки, сформульовані вимоги до підсистеми адміністратора, бухгалтера та менеджера. Виконано проектування усіх бізнес-процесів інформаційної системи керування готелем та створена база даних системи.

У розробці інформаційної системи використовувалися наступні програмні засоби: дворівнева клієнт-серверна архітектура, СУБД PostgreSQL, сервер Tomcat, об'єктно-орієнтована мова програмування Java, мова гіпертекстової розмітки HTML та каскадні таблиці стилів CSS.

Завдяки використанню сучасних технологій розробки інформаційна система управління готелем має зручний та зрозумілий інтерфейс і може бути легко розширена та доповнена новими функціональними можливостями.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Левицька І. В, Корж Н. В., Онищук Н. В. Готельна справа: Навч. посібник. К.:Київ.нац.торг.-екон.ун-т, 2015. 580 с.
2. І. К. Нестерчук, С.В. Тищенко, Т.М. Чернишова, Б. Л. Шевчук, А.С. Осіпчук. Технологія готельної справи: Навч. посібник. Житомир: ЖНАЕУ, 2019. 223 с.
3. Cenium Hospitality ERP. URL: <https://cenium.com/> (дата звернення 10.05.2023).
4. Система управління готелем jSolutions. URL: <https://jsolutions.ua/> (дата звернення 10.05.2023).
5. Програма автоматизації підприємств Рарус. URL: <https://rarus.com.ua/> (дата звернення 10.05.2023).
6. Марченко А. В. Проектування інформаційних систем: Навч. посібник. К.: ДУТ, 2016. 90 с.
7. М. Ю. Карпенко, Н. О. Манакова, І. О. Гавриленко. Технології створення програмних продуктів та інформаційних систем: Навч. посібник. Харків: ХНУМГ, 2017. 94 с.
8. В. Basham, В. Bates. Head First Servlets and JSP: Passing the Sun Certified Web Component Developer Exam Second edition: O'Reilly Media, 2008. 914 p.
9. J. Brittain, I. F. Darwin. Tomcat: The Definitive Guide: The Definitive Guide: O'Reilly Media, 2007. 496 p.
10. Єрьоміна Н.В. Проектування баз даних: Навч. посібник. К.: КНЕУ, 1998. 208 с.
11. Hans-Jurgen Schonig. Mastering PostgreSQL 15: Advanced techniques to build and manage scalable, reliable, and fault-tolerant database applications: Packt Publishing, 2023. 522 p.
12. Васильєв О. Програмування мовою Java: Навч. книга. Богдан, 2020. 696 с.

13. P. McFedries. Web Design Playground: HTML & CSS the Interactive Way: Manning, 2019. 440 p.
14. Yuliu C., Li Q. Modeling and Analysis of Enterprise and Information Systems: Springer Berlin, Heidelberg, 2009. 400 p.