

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ ІМЕНІ ІВАНА ФРАНКА
ФАКУЛЬТЕТ УПРАВЛІННЯ ФІНАНСАМИ ТА БІЗНЕСУ

**Кафедра цифрової економіки та бізнес-
аналітики**

КУРСОВА РОБОТА
з навчальної дисципліни
“Проектування та адміністрування БД і СД”

Тема:

«Інформаційна система для салону краси»

Науковий керівник:

к.ф.-м.н., доц. Депутат Б.Я.

(прізвище, ім'я, по-батькові)

_____ (підпис)

Виконавець:

Пономаренко М.І.

(прізвище, ім'я, по-батькові)

УФЕ-31с група

_____ (підпис)

“ ____ ” _____ 2020 р.

“ ____ ” _____ 2020 р.

Загальна кількість балів _____

(підписи, ППП членів комісії)

Львів 2020

ЗМІСТ

ВСТУП.....	3
РОЗДІЛ 1. АНАЛІЗ ВИМОГ.....	5
1.1 Постановка завдання.....	5
1.2 Аналіз засобів.....	6
1.3 Інформаційний ресурс салону краси.....	8
РОЗДІЛ 2. СТРУКТУРА БАЗИ ДАНИХ.....	10
2.1 Опис моделі даних.....	10
2.2. Нормалізація відношень.....	11
2.3 Визначення типів даних.....	13
2.4 Організаційна структура бази даних.....	15
2.5 Проектування бази даних.....	20
2.6 Запити бази даних.....	21
РОЗДІЛ 3. РОЗРОБКА ВЕБ-ДОДАТКУ.....	25
3.1 Структура веб-сайту.....	25
3.2 Макети сторінок веб-сайту.....	26
3.3 Програмування серверної та клієнтської частини.....	28
3.4 Розміщення веб-сайту на локальному віртуальному середовищі або в Інтернеті.....	31
ВИСНОВКИ.....	37
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	38

ВСТУП

В теперішньому світі неможливо уявити розвиток будь-якого бізнесу без сучасних інформаційних технологій, які створені для прискорення обміну інформацією та доступу до різних інформаційних джерел.

Розвиток інформаційних технологій у сфері надання послуг поставив на якісно новий рівень конкурентоспроможність підприємств, зокрема і салонів краси. Розроблення функціонального та структурованого інформаційного ресурсу формує позитивний образ салону у WWW, що сприяє залученню нових клієнтів та інформуванню постійних відвідувачів. Для успішної роботи і утримання високих позицій в індустрії краси необхідні не тільки якісно надавати послуги, але й проводити рекламну кампанію для розширення клієнтської бази.

Основою сучасних інформаційних технологій є те, що всі важливі дані повинні зберігатись в базі даних. За допомогою неї вся актуальна та важлива інформація завжди буде у близькому доступі, це економить дуже багато часу та запобігає майбутнім помилкам від дезінформації. Для створення і функціонування таких баз даних є спеціальні програмні комплекси, які називаються системами управління базами даних. Модель даних – це ядро будь-якої бази даних. Це безліч структур даних, які обмежені маніпулюванням і цілісністю даними. За допомогою неї можуть бути представлені об'єкти предметної області та взаємозв'язки між ними.

Метою курсової роботи є обґрунтування теоретичних основ та реалізація інформаційної системи для салону краси. У цьому проекті наведені короткі теоретичні відомості про засоби, використані для реалізації практичної задачі, конкретна постановка задачі, опис її функціональних можливостей. Основним етапом практичного завдання є проектування бази даних на основі системи управління базами даних MYSQL, складання запитів і взаємозв'язків між різними таблицями.

Завданням курсової роботи є:

1. Визначити особливості сфери бізнесу салону краси
2. Проаналізувати теоретичні засади проектування та реалізації систем на основі баз даних.
3. Визначити інформаційні потреби предметної області дослідження.
4. Проектування бази даних за визначеною предметною областю.

Об'єктом проведеного дослідження даної роботи є практичне використання такого засобу розробки, як MySQL.

Предметом є теоретичні, методичні та практичні аспекти розробки інформаційної системи для зберігання потрібної інформації салону краси.

Практичне значення отриманих результатів. База даних може використовуватись власником та робочим персоналом салону, які хочуть знайти та обрати інформацію, яка їх цікавить.

Використане програмне забезпечення. Для створення бази даних використовувалось середовище розробки MySQL.

Структура роботи. Курсова робота складається з трьох розділів («Аналіз вимог», «Структура бази даних» та «Розробка веб-додатку»), висновків, списку використаних джерел та додатків. Загальний обсяг роботи – сторінок.

РОЗДІЛ 1. АНАЛІЗ ВИМОГ

1.1. Постановка завдання

Розвиток інформаційних технологій у сфері надання послуг поставив на якісно новий рівень конкурентоспроможність підприємств, зокрема і салонів краси. Розроблення функціонального та структурованого інформаційного ресурсу формує позитивний образ салону у WWW, що сприяє залученню нових клієнтів та інформуванню постійних відвідувачів. Для успішної роботи і утримання високих позицій в індустрії краси необхідні не тільки якісно надавати послуги, але й проводити рекламну кампанію для розширення клієнтської бази.

Розроблення інформаційного ресурсу салону краси є комплексною задачею, яка полягає у вивченні потреб та побажань власника салону краси щодо функціональних можливостей та дизайну веб-ресурсу. Адже, мережевий ресурс ефективно використовується як інструмент популяризації салону краси у глобальній мережі. Звертаючи увагу на вже сформовані вимоги до інформаційного ресурсу, варто адаптувати їх саме до потреб салону краси.

Створення такого інформаційного ресурсу для салону краси надасть великий спектр можливостей для популяризації салону та покращення рівня надання послуг, зокрема: - інформування клієнтів про новинки, цікаві акції, знижки та спеціальні пропозиції, що є способом утримання постійних клієнтів та ефективним засобом для залучення нових клієнтів (наприклад, "приведи з собою подругу і отримай 50% знижки на манікюр");

- запис на послугу до майстра у зручний для клієнта час (24/7);
- розміщення необмеженої кількості фото/відео з роботами майстра та пропозицій для клієнтів;

- автоматичний розрахунок клієнтом вартості послуги на основі заданих ним параметрів;
- віртуальний тур по салону;
- актуалізація та структурування даних клієнтів, наприклад, дата народження з метою привітання, електронної адреси для інформування про зміни в салоні, а також акції та знижки;
- замовлення подарункових сертифікатів;
- оплата послуг он-лайн;
- розроблення декількох комплексів обслуговування.

1.2. Аналіз засобів

Інформаційна технологія - це цілеспрямована організована сукупність методів, процесів та програмно-технічних засобів, об'єднаних у технологічний ланцюг, який забезпечує збір, зберігання, оброблення та передачу інформації з метою ефективної організації діяльності людей.



Рис.1.1. Складові інформаційної технології

Інформаційна технологія досить тісно пов'язана з інформаційними системами, тобто її основним середовищем.

Неможливо використовувати інформаційну систему без знання відповідної інформаційної технології. З появою персональних комп'ютерів інформаційні технології отримали новий імпульс для розвитку, все більше задовольняючи інформаційні потреби людей у професійній та побутовій діяльності.

Властивості інформаційних технологій:

- цілеспрямованість;
- доцільність;
- наявність компонентів та структури;
- взаємодія з зовнішнім середовищем;
- системна повнота;
- регулярність процесів;
- динамічність.

Технологічний процес обробки інформації - це комплекс взаємопов'язаних операцій перетворення інформації, виконаних у певній послідовності, від її виникнення до використання користувачами для досягнення поставленої мети.

Упродовж останнього десятиліття відбулося становлення нової науки, що вивчає інформаційні технології, - ітології, її предмет – інформаційні технології та процеси їх створення й застосування. Ітологія - це така ж сама фундаментальна наука, як філософія і математика.

Інформаційна технологія поєднує об'єкти, дії, правила обробки інформації в індивідуальній та масовій виробничій діяльності. До складу інформаційних технологій входять мікроелектроніка, виробництво комп'ютерів та програмного забезпечення, зв'язок і телефонія, послуги мобільного зв'язку, забезпечення послуг Інтернет, автоматизація виробництва.

Інформаційні технології - це сукупність методів та способів розв'язання типових задач обробки інформації, їх програмна реалізація з детальним описом технології вирішення цих задач.

Інформаційні технології можна розглядати у концептуальному плані як методологічний базис формалізації, аналізу та синтезу знань, а в технологічному плані - як інструмент підвищення інтелектуальних можливостей людини.

Саме інформаційні технології дають змогу перекинути міст між гуманітарними та природничими дисциплінами, здійснити інтеграцію різних галузей знань, духовного світу та матеріального виробництва.

1.3. Інформаційний ресурс салону краси

Інформаційний ресурс салону краси має відповідати всім вимогам створенням інформаційного забезпечення для підприємства, таким як: уніфікація, достовірність інформаційного наповнення, контроль, цілісність системи та ін.

Особливу увагу треба приділити достовірності персональних даних працівників та інформація салону краси (освіта, стаж роботи, нагороди, відзнаки, участь у майстер-класах, стажування). Обов'язковими елементами цього інформаційного ресурсу є:

- надання контактної інформації салону краси (телефонів, адреси, карти проїзду та е-мейл);
- поширення інформації через соціальні мережі;
- форум для обговорень, статті, рекомендації;
- мобільна версія інформаційного ресурсу;

-веб-ресурс повинен бути багатомовним (англомовний варіант є обов'язковим);

-гостьова книга для висловлення пропозицій, побажань, скарг та зауважень клієнтів.

Створення власного мережевого ресурсу салону краси виступає одним з найбільш ефективних способів залучення та інформування клієнтської аудиторії.

Запропоновано нові функціональні можливості веб-ресурсу, що сприяють покращенню бізнес процесу салону краси. Грамотне та добре продумане розроблення інформаційного ресурсу надасть вагому перевагу салону краси з-поміж конкурентів, розширить клієнтську базу, створивши позитивний імідж у глобальній мережі.

Підвищить довіру потенційних клієнтів до салону краси, слугуючи візитівкою не тільки для відвідувачів, але також і для потенційних співробітників.

РОЗДІЛ 2. СТРУКТУРА БАЗИ ДАНИХ

2.1. Опис моделі даних

У штат салону краси входять:

Директор - обов'язки директора салону краси різноманітні: від взаємодії з державними органами до облаштування приміщення та забезпечення співробітників необхідними матеріалами.

Заступник директора - заступник директора відноситься до категорії керівників, приймається на роботу і звільняється з неї наказом директора підприємства. Під час відсутності директора виконує його обов'язки

Секретар - секретар призначається на посаду і звільняється з посади в установленому чинним трудовим законодавством порядку наказом керівника підприємства. Секретар повинен знати: постанови, розпорядження, накази та інші керівні матеріали і нормативні документи, що стосуються діяльності підприємства і ведення діловодства.

Головний бухгалтер - відноситься до категорії керівників. Головний бухгалтер призначається на посаду і звільняється з неї наказом генерального директора підприємства. Головний бухгалтер підпорядковується безпосередньо директору.

Бухгалтер - призначається на посаду і звільняється з неї наказом директора підприємства за поданням головного бухгалтера. Бухгалтер безпосередньо підпорядковується головному бухгалтеру. На час відсутності бухгалтера його права і обов'язки переходять до іншої службової особи. Виконує роботу з ведення бухгалтерського обліку майна, зобов'язань і господарських операцій.

Масажисти - призначення на посаду масажиста і звільнення з неї здійснюється наказом директора підприємства. Масажист підпорядковується заступнику директора підприємства. На посаду масажиста призначається особа, яка має середню медичну освіту за спеціальністю «Медичний масаж» без пред'явлення вимог до досвіду роботи.

Перукарі - призначення на посаду перукаря і звільнення з неї здійснюється наказом директора підприємства. Перукар підпорядковується заступнику директора. На посаду перукаря призначається особа, яка має середню освіту і вчинила відповідні курси, без пред'явлення вимог до досвіду роботи.

Косметологи - призначення на посаду косметолога і звільнення з неї здійснюється наказом директора підприємства. Косметолог підпорядковується заступнику директора. На посаду косметолога призначається особа, яка має середню медичну освіту за спеціальністю «Медичний масаж» без пред'явлення вимог до досвіду роботи.

Технічний персонал - приймається на роботу і звільняється з роботи директором підприємства, за поданням зам. директора. Технічний персонал підпорядковується безпосередньо заступнику директора. У своїй роботі керуються правилами санітарії та гігієни щодо утримання приміщень салону краси: інструкціями щодо призначення устаткування, що обслуговується і пристосувань.

2.2. Нормалізація відношень

Нормалізація відношень- покроковий процес розділення (декомпозиції) початкових відношень БД на простіші. Кроки цього процесу переводять схему відношення БД в послідовні нормальні форми. Кожна наступна форма володіє кращими властивостями ніж попередня. Кожній нормальній формі відповідає

певний набір обмежень. При переведенні структури відношення у форми вищого порядку досягають видалення з таблиць надмірної описової інформації. Процес нормалізації заснований на понятті функціональної залежності атрибутів.

Перша нормальна форма. Відношення відповідає 1NF тоді, коли на перетині кожного стовпця і кожного рядка знаходяться тільки елементарні (неподільні) значення атрибутів і не містяться групи, що повторюються.

Друга нормальна форма. Відношення знаходиться в 2NF, якщо виконуються обмеження 1NF і кожен описовий атрибут функціонально повно залежить від первинного ключа (у тому числі і складеного).

Третя нормальна форма. Відношення знаходиться у 3NF, якщо виконуються обмеження 2NF і всі описові атрибути відношення взаємно незалежні і повністю залежать від первинного ключа, тобто кожний описовий атрибут не транзитивно залежить від ключа.

Четверта нормальна форма (4НФ, 4NF) вимагає, аби в схемі баз даних не було нетривіальних багатозначних залежностей множин атрибутів від будь чого, окрім надмножини ключа-кандидата. Вважається, що таблиця знаходиться у 4НФ тоді, і тільки тоді, коли вона знаходиться в НФБК, та багатозначні залежності є функціональними залежностями. Четверта нормальна форма усуває небажані структури даних — багатозначні залежності.

П'ята нормальна форма (5НФ, 5NF, PJ/NF) вимагає, аби не було нетривіальних залежностей об'єднання, котрі б не витікали із обмежень ключів. Вважається, що таблиця в п'ятій нормальній формі, тоді, і тільки тоді, коли вона знаходиться в 4НФ, та кожна залежність об'єднання зумовлена її ключами-кандидатами.

Шоста нормальна форма. Таблиця знаходиться у 6nf, якщо вона знаходиться у 5nf та задовольняє вимозі відсутності нетривіальних залежностей. Зазвичай 6nf ототожнюють з dknf.

Цілі нормалізації наступні: Виключити дублювання інформації в таблицях. Забезпечити можливість змін у структурі таблиць. Зменшити вплив структурних змін бази даних на роботу додатків, які забезпечують користувачам доступ до даних.

Основні правила нормалізації:

- В кожній таблиці повинен бути первинний ключ;
- Зміна значень в полях не повинна впливати на інформацію в інших полях;
- В кожній базі даних не повинно бути повторюваних полів;
- Кожному значенню первинного ключа повинна відповідати інформація про тип суті або про об'єкт таблиці.

2.3. Визначення типів даних

Типи даних - це множина припустимих значень цих даних і множина дозволених операцій над ними. Водночас тип даних визначає і розмір пам'яті, що займають змінні і константи даного типу. Пам'ять виділяється не для типу даних, а виділяється для розміщення змінної або константи заданого типу.



Рис.2.1. Типи даних

Стандарт мови SQL визначає ряд типів даних. Хотілося б відзначити, що деякі СУБД можуть не підтримувати частина з перерахованих типів або навпаки - додавати свої типи даних.

Найбільш часто використовувані типи даних:

INTEGER, SMALLINT - цілі числа зі знаком. Точність уявлення і розмірність визначається реалізацією, точність SMALLINT не повинна перевищувати точність INTEGER. Багато СУБД дозволяють скорочувати запис INTEGER до INT.

NUMERIC (p, s), DECIMAL (p, s) - задають в загальному вигляді формат точного числа, цілого або з дробовою частиною. NUMERIC представляє число з точністю p (число знаків) і масштабом s (число знаків після коми). Якщо опущений масштаб, то він вважається рівним 0 (тобто число ціле), а якщо опущена точність, то її значення за замовчуванням визначається реалізацією. Тип DECIMAL вимагає, щоб використовувалася залежить від реалізації точність t, така, що $t \leq p$. Припустимо скорочення DEC.

FLOAT (p), REAL, DOUBLE PRECISION - числа в поданні з плаваючою точкою (наближені числові типи). Точність REAL визначається реалізацією, точність DOUBLE PRECISION повинна бути більше точності REAL. Точність FLOAT задається параметром p, якщо параметр відсутній - визначається реалізацією. Приклади допустимих значень: 34, -11.23, 23.45E-15 (запис числа 23,45 15).

CHARACTER (n) - рядок з n символів, допускається скорочення CHAR (n). Запис "CHAR" відповідає CHAR (1). Якщо довжина зберігається значення менше n, в кінці буде додано відповідне число прогалів. Тип NATIONAL CHARACTER (n) зберігає дані в кодуванні UNICODE (2 байта на символ для підтримки національних алфавітів). Допустима запис NCHAR. Строкові константи полягають в одинарні або подвійні лапки (наприклад, 'example', "example 2").

CHARACTER VARYING (n) - рядок змінної довжини не більше ніж з n символів, допускається скорочення VARCHAR (n). Якщо довжина зберігається значення m, де m < n, буде зберігатися рівно m символів. Аналогічно попередньому випадку, існує тип NATIONAL CHARACTER VARYING (n) з синонімами NCHAR (n).

CHARACTER LARGE OBJECT - символічний тип, що дозволяє зберігати великі обсяги тексту. Допустима запис CLOB. Також існує тип NCLOB.

XML - документ у форматі XML (тип даних введений у версії стандарту SQL: 2003).

BIT (n) - бітова рядок довжиною n біт. Запис "BIT" аналогічна BIT <n>.

BIT VARYING (n) -бітова рядок змінної довжини не більше n біт.

BINARY LARGE OBJECT - великий двійковий об'єкт, допустима запис blob.

DATE - дата в форматі 'YYYY-MM-DD', наприклад '1961-04-12' для 12 квітня 1961 р

TIME - тип даних для зберігання відміток часу, що включають поля <годинник>: <хвилини>: <секунди>: <частки секунди>.

TIMESTAMP - тип даних для спільного зберігання дати та часу.

2.4. Організаційна структура бази даних

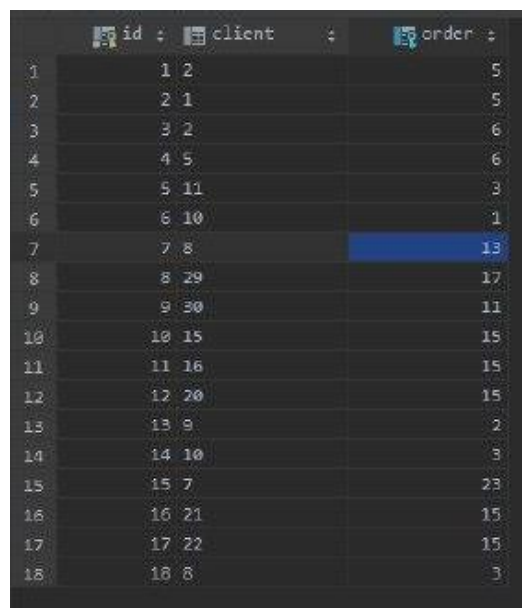
Таблиця містить дані по певній темі, наприклад, відомості про співробітників або клієнтів. Кожен запис в таблиці включає дані про один елемент, наприклад про конкретного співробітника. Запис складається з полів і включає такі відомості, як ім'я, адреса та номер телефону. Крім того, запис зазвичай називається рядком, а поле - стовпцем.

У даній роботі всього 5 таблиць:

- Workers – таблиця, яка містить інформацію про співробітників;
- Orders – таблиця, яка містить інформацію про замовлення послуг у салоні;
- Clients – таблиця, яка містить інформацію про клієнтів;
- Salary – таблиця, яка містить інформацію про заробітну плату співробітників, що працюють у салоні;
- Services – таблиця, яка містить інформацію про послуги салону.

У таблиці «Замовлення» містяться такі поля:

- Табельний номер співробітника;
- Код клієнта;
- Код послуги;
- Вартість послуги.



	id	client	order
1	1	2	5
2	2	1	5
3	3	2	6
4	4	5	6
5	5	11	3
6	6	10	1
7	7	8	13
8	8	29	17
9	9	30	11
10	10	15	15
11	11	16	15
12	12	20	15
13	13	9	2
14	14	10	3
15	15	7	23
16	16	21	15
17	17	22	15
18	18	8	3

Рис 2.2. Таблиця «Замовлення»

У таблиці «Клієнти» містяться такі поля:

- Код клієнта;
- ім'я;

- Прізвище;
- Телефон.

	Id	Name	Phone
1	1	Olesia Kushnir	380961230001
2	2	Natalia Vorona	380961230002
3	3	Yana Kostluk	380961230003
4	4	Chrystyna Kotyk	380961230004
5	5	Maria Reznik	380961230005
6	6	Liuba Shynkar	380961230006
7	7	Galina Belei	380961230007
8	8	Viktorla Krestenchuk	380961230008
9	9	Anna Kravchuk	380961230009
10	10	Daryna Laba	380961230010
11	11	Oksana Shtiba	380961230011
12	12	Marta Melnyk	380961230012
13	13	Iryna Romaniuk	380961230013
14	14	Nina Shved	380961230014
15	15	Alina Vorona	380961230015
16	16	Vesyl Sharp	380961230016
17	17	Sesha Savka	380961230017
18	18	Andrii Ivashkiv	380961230018
19	19	Ostap Girnyk	380961230019
20	20	Vitalii Dykyj	380961230020
21	21	Nazar Maslyuk	380961230021
22	22	Oleksandr Polamar	380961230022
23	23	Andrii Shymenskyi	380961230023
24	24	Ivan Vyval	380961230024
25	25	Mykola Myshchak	380961230025

Рис 2.3. Таблиця «Клієнти»

У таблиці «Співробітники» містяться такі поля:

- Табельний номер;
- Посада;
- ім'я
- Прізвище;
- Вік;
- Номер телефону.

	Id	FirstName	LastName	Age	Position	phone
1	1	Marta	Balyk	30	director	0670000001
2	2	Tetiana	Klish	25	administrator	0670000002
3	3	Viktoria	Potatkovska	24	administrator	0670000003
4	4	Diana	Rybachova	22	secretary	0670000004
5	5	Sofia	Borkovska	35	accountant	0670000005
6	6	Alisa	Vanilke	28	female hairdresser	0670000006
7	7	Oksana	Gutsiv	23	female hairdresser	0670000007
8	8	Veronika	Bastyn	20	female hairdresser	0670000008
9	9	Anastasia	Kovajska	34	female hairdresser	0670000009
10	10	Maksim	Batevol	27	male hairdresser	0670000010
11	11	Andrian	Kovalshyk	22	male hairdresser	0670000011
12	12	Solomia	Korchar	19	manicurist	0670000012
13	13	Evangelina	Chebot	25	manicurist	0670000013
14	14	Natoliia	Steslavsko	24	manicurist	0670000014
15	15	Niko	Fedorov	40	maid	0670000015
16	16	Viko	Melova	26	make-up artist	0670000016
17	17	Alina	Navorova	36	make-up artist	0670000017
18	18	Oksana	Yrchilo	31	make-up artist	0670000018
19	19	Julia	Matskiv	29	eyebrow master	0670000019
20	20	Margaryta	Shpakovska	32	epilation master	0670000020
21	21	Olena	Sloboda	30	lashes master	0670000021

Рис 2.4. Таблица «Співробітники»

У таблиці «Послуги» містяться такі поля:

- Код послуги;
- Табельний номер послуги;
- Послуга;
- Вартість послуги.

	Id	Name	praise
1	1	women's haircut and washing	400
2	2	women's haircut	300
3	3	shampoo	150
4	4	hair styling	250
5	5	hair polishing	300
6	6	men's haircut (clipper)	200
7	7	men's haircut (scissors)	300
8	8	men's haircut (scissors + clipper)	250
9	9	hair coloring short	450
10	10	hair coloring medium	650
11	11	hair coloring long	850
12	12	highlighting hair	600
13	13	hair extensions	2700
14	14	baby haircut	150
15	15	make-up	600
16	16	evening make-up	800
17	17	eye brow tattoo	650
18	18	threading	300
19	19	waxing	400
20	20	hair removal	700
21	21	nail cleaning	200
22	22	manicure	400
23	23	pedicure	500
24	24	complex(make-up + manicure)	900

Рис 2.5. Таблица «Послуги»

У таблиці «Заробітна плата» містяться такі поля:

- Табельний номер співробітника;
- Зарплата.

	id	size salary
1	1	40000
2	2	20000
3	3	20000
4	4	12000
5	5	10000
6	6	30000
7	7	30000
8	8	30000
9	9	30000
10	10	25000
11	11	25000
12	12	20000
13	13	20000
14	14	29000
15	15	8000
16	16	25000
17	17	27000
18	18	25000
19	19	20000
20	20	30000
21	21	26000

Рис. 2.6. Таблиця «заробітна плата»

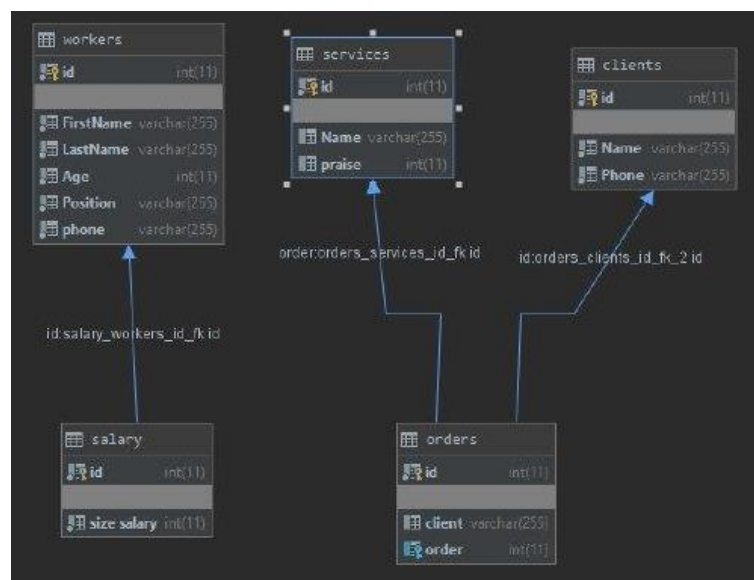


Рис. 2.7. Концептуальна модель бази даних

2.5. Проектування бази даних

Основною метою даної курсової роботи є проектування бази даних «Салон краси», яка підтримує структуровану обробку даних про клієнтів, співробітників, послуги і т.д. Завдання, які необхідно вирішити з використанням БД «Салон краси»:

- скорочення надмірності збережених даних;
- збір і зберігання інформації про клієнтів, співробітників, наданих послугах;
- обробка даних (вивід потрібної інформації в звітах).

На основі даних можна відстежувати інформацію про надані послуги і послуги, які заплановані на майбутнє.

Вимоги до бази даних:

- цілісність бази даних;
- багаторазове використання даних;
- швидке отримання інформації при пошуку по запитах користувачів;
- простота оновлення даних;
- адекватність відображення даних.

База даних «Салон краси» проста в застосуванні і може бути використана навіть людиною, яка володіє тільки основами знань з інформатики. Людина, яка працює з базою даних «Салон краси», може вести списки Клієнтів, співробітників, також і мати Можливість простежувати оплату за виконані послуги, виводити інформацію по клієнтах і співробітниках, робити звіти та інше.

2.6. Запити бази даних

Запит - це засіб вибору необхідної інформації з бази даних. Питання, сформований по відношенню до бази даних, і є запит. Запити в MySQL дозволяють управляти даними: додавати, видаляти, змінювати.

Є три види запитів:

1. Інформаційні запити - користувач шукає певну інформацію, не піклуючись про те, на якому саме веб-сайті він її виявить.

2. Навігаційні запити - користувач шукає сайт, де, за його припущенням, міститься цікава йому інформація.

3. Транзакційні запити - у формулюванні запиту користувач висловлює свою готовність зробити якусь дію.

У моїй базі даних були використані наступні запити:

- (SELECT * from workers) показує всіх працівників салону;
- (SELECT * from workers where age >30) показує всіх працівників, які старші за 30 років;
- (SELECT FirstName, phone, Position from workers where Position ='administrator' or Position='director') показує ім'я і телефон адміністраторів, директора салону;
- (SELECT * from clients order by Name asc limit 10) показує перших 10 клієнтів по алфавіту;
- (SELECT * from clients where Phone like '%15') показує клієнтів, чий номер закінчується на 15;
- (SELECT FirstName, phone from workers where FirstName like 'Oksana'and LastName like 'Gutsiv' or FirstName like 'Natalia'and LastName like 'Steslavska') показує імена, номер телефону працівників;

- (SELECT Name, praise from services where praise > 500) показує послуги і ціни, які коштують дорожче 500;
- (SELECT Name, praise from services where praise between 600 and 900) показує всі послуги, які коштують дорожче 600 і дешевше 900;
- (SELECT Sum(price) from services) рахує загальну суму всіх послуг;
- (SELECT Name, Max(praise) from services) показує найдорожчу послугу салону;
- (SELECT FirstName, Position from workers join salary s on workers.id = s.id
where `size salary`>26000
group by FirstName, Position) показує всіх працівників, які мають заробітну плату більше 26000;
- (SELECT * from workers
group by FirstName
order by min(Age)
limit 5) показує 5 наймолодших працівників салону;
- (SELECT * from orders join clients c on orders.id = c.id
join services s on orders.`order` = s.id
group by orders.id) показує інформацію про замовлення;
- (SELECT * from orders join clients c on orders.id = c.id
join services s on orders.`order` = s.id
where praise >= 600
group by orders.id) показує всю інформацію про замовлення на суму рівну 600;
- (SELECT * from services
group by id
order by min(praise) desc) показує всі послуги салону від найдорожчої до найдешевшої;
- (SELECT * from workers

where Position like 'manicurist') показує всіх працівників, які є майстрами манікюру;

- (SELECT *from workers join salary s on workers.id = s.id

where Position like 'female hairdresser') показує, скільки заробляють жіночі перукарі;

- (UPDATE salary

set size salary = `size salary`+ 10%

where id='1,2,3,4,5,6,7,8,9,10') збільшує заробітну плату на 10%;

- (UPDATE services

set praise = praise+5%

where Name like 'make-up') збільшити ціну послуги «макіяж» на 5%.

2.7. Реалізація SQL-скрипту

Приклад створення таблиці:

```
CREATE TABLE `orders` (
  `id` int(11) NOT NULL,
  `client` varchar(255) DEFAULT NULL,
  `order` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `orders_services_id_fk` (`order`),
  CONSTRAINT `orders_clients_id_fk_2` FOREIGN KEY (`id`)
  REFERENCES `clients` (`id`) ON DELETE CASCADE ON UPDATE
  CASCADE,
```

```
CONSTRAINT `orders_services_id_fk` FOREIGN KEY (`order`)
REFERENCES `services` (`id`) ON DELETE CASCADE ON UPDATE
CASCADE
```

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Приклад внесення даних у таблицю:

```
LOCK TABLES `orders` WRITE;
```

```
/*!40000 ALTER TABLE `orders` DISABLE KEYS */;
```

```
INSERT INTO `orders` VALUES
```

```
(1,'2',5),(2,'1',5),(3,'2',6),(4,'5',6),(5,'11',3),(6,'10',1),(7,'8',13),(8,'29',17),(9,'30',11),(10
,'15',15),(11,'16',15),(12,'20',15),(13,'9',2),(14,'10',3),(15,'7',23),(16,'21',NULL),(17,'2
2',NULL),(18,'',NULL);
```

```
/*!40000 ALTER TABLE `orders` ENABLE KEYS */;
```

```
UNLOCK TABLES;
```


РОЗДІЛ 3. РОЗРОБКА ВЕБ-ДОДАТКУ

3.1. Структура веб-сайту

Веб-сторінка— інформаційний ресурс, доступний у мережі World Wide Web, який можна переглянути у веб-браузері. Зазвичай ця інформація записана у форматі HTML або XHTML і може містити гіпертекст із навігаційними гіперпосиланнями на інші веб-сторінки.

Веб-сайт— сукупність веб-сторінок, доступних в Інтернеті, які об'єднані як за змістом, так і навігаційно.

Фізично сайт може розміщуватися як на одному, так і на кількох серверах. Зовнішній вигляд кожного сайту є унікальним, проте в усіх сайтів можна знайти спільні за функціональністю частини. Структура веб-сайтів:

1. Головна сторінка
2. Меню сайту
3. Гіперпосилання на інші сторінки або сайти

На будь-якому сайті першою відкривається головна сторінка. У верхній частині головної сторінки зазвичай розташована так звана шапка, яку дублюють на інших сторінках сайту. Це роблять спеціально, адже ця частина відображається у вікні браузера першою і відвідувач насамперед звертає увагу на неї.

Щоб забезпечити швидкий перехід до основних тематичних розділів сайту, створюють меню сайту — список гіперпосилань на його розділи. Горизонтальне меню зазвичай розташовують у шапці, іноді дублюючи його в нижній частині сторінки, а вертикальне — переважно в лівій частині сторінки, у місці, звідки відвідувач починає її переглядати. Меню є одним із найважливіших компонентів

сайту, користувач постійно звертає на нього увагу, і тому вимоги до нього високі. Меню має бути зручним, помітним і зрозумілим, інакше користувач не знатиме, як перейти до потрібного розділу, і покине сайт. Пункти меню мають бути чітко відділені один від одного.

Гіперпосилання, розміщені в тексті чи у вигляді графічних об'єктів, дозволяють переходити на різні сторінки сайту або навіть на інші сайти. На сайтах із дуже великим обсягом інформації є сторінки третього рівня, а якщо необхідно — то й четвертого, п'ятого тощо.

Виділяють три типи структур веб-сайтів — лінійну, деревоподібну та довільну. Подорожуючи сайтом із лінійною структурою, з головної сторінки ви перейдете на другу сторінку, з неї — на третю тощо. На сайті з деревоподібною структурою з головної сторінки можна потрапити на одну зі сторінок другого рівня, звідти — на одну зі сторінок третього рівня тощо. Сайт із довільною структурою видається зовсім неорганізованим, але саме в цьому й полягає принцип його створення. Подорожуючи таким сайтом, ви можете переходити з однієї його сторінки на інші в різні способи, і ваш шлях назад не обов'язково має бути таким самим.

3.2. Макети сторінок веб-сайту

Одним із найбільш важливих етапів в розробці майбутнього сайту вважається розробка його макету. Макет - це графічна схема із зазначенням відтінків, відступів і інших параметрів в статиці.

Макет сайту – це щось середнє між блоковою схемою і прототипом. Це оптимальний варіант для клієнта, якому не доведеться переплачувати за інтерактивні ефекти прототипу. З іншого боку, макет не виглядає занадто простим як блокова схема. З його допомогою простіше зрозуміти, яким саме має

бути сайт. Залежно від домовленості між клієнтом і дизайнером, макет виглядає як блок із зазначенням обсягів або як майже готовий дизайн.

Переваги використання макета:

1. Визначення візуальних недоліків на перших стадіях роботи над проектом.
2. Можливість наочно показати клієнту дизайнерські ідеї.
3. Спрощення роботи програміста.
4. Розуміння того, як буде виглядати готовий сайт.
5. Зведення до мінімуму непорозумінь між дизайнером і замовником.

Вимоги до готового макету:

1. Симетричне відображення візуальних елементів.
2. Вказівка параметрів макету.
3. Створення елементів в різних варіантах відображення.
4. Використання сітки для вирівнювання блоків та інших елементів.
5. Окремий файл для повторів.
6. Використання стандартних шрифтів.
7. Використання окремих шарів для всіх елементів.

Помилки при створення макетів:

1. Занадто багато елементів на сторінці.
2. Незбалансованість колірних відтінків.
3. Наявність непотрібних шарів, які потрібно було видалити.
4. Невідповідний фон і нечитабельним шрифт.
5. Немає адаптації під різні пристрої.
6. Невідповідні розміри макета сайту.

Основні робочі елементи макета: єдиний стиль, структура, кнопки призову, форми підписки і зворотного зв'язку, логотип (за бажанням клієнта), контактні дані, шапка, основний контент, меню, сайдбар, футер.

3.3. Програмування серверної та клієнтської частини

Процес створення сайту розділений на дві частини, клієнтську і серверну (Frontend та Backend). Клієнтська частина це Frontend розробка, до неї можна віднести CSS верстку, створення макета і шаблонів для сайту, а також призначеного для користувача інтерфейсу і спеціальних скриптів, що відповідають за візуалізацію і анімацію. До серверної частини (Backend розробка) відноситься створення ядра сайту, розробка платформи, основного функціоналу та адміністративної частини.

Наприклад, уявіть собі салон краси. Декорації, меблі, а також музика є клієнтською частиною (Frontend), а серверна частина (Backend) включає в себе майстрів, перукарів і менеджерів, які забезпечують безперебійну роботу салону.

Front-end - це та частина програмного забезпечення з якою безпосередньо контактує користувач. Для прикладу у веб-розробці в якості Front-end виступають:

- HTML-верстка;
- Стили CSS;
- JavaScript.

Власне ці технології створюють картинку для користувача і з ними він контактує безпосередньо виконуючи ті чи інші дії на сайті.

Основною задачею front-end розробника є створення привабливого, зручного веб-дизайну та інтерфейсу.

Frontend Developer— це спеціаліст, який має добрі навички у побудові web додатків на стороні клієнта. Щоб представити контент сайту найкращим чином та забезпечити ідеальну взаємодію користувача з сайтом, фронтенд-розробники використовують багато корисних інструментів, найважливіші з яких — HTML, CSS та JavaScript.

HTML (Hypertext Markup Language) відповідає за представлення коду користувачам у зрозумілій та звичній формі.

CSS (Cascading Style Sheets) пропонує чудові можливості із вдосконалення стилів (кольори, фони, блики і т.д.) Javascript робить взаємодію користувача з сайтом швидшою та цікавішою за допомогою слайдів, випадних меню, безкінечного скролінгу та багатьох інших способів — вибір безмежний.

Фронтенд – це робота з тим, що бачить звичайна людина на екрані свого ноутбука чи смартфона, коли заходить на сайт. Усе, починаючи від шаблону, та закінчуючи розташуванням тексту – завдання front-end розробника. Він повинен мати великий досвід у HTML та CSS, гарні знання написання скриптів та володіння Javascript. Основною задачею фронтендника є створення платформи для спілкування з користувачами сайту, передачі та отримання інформації. Це означає, що вам доведеться вивчати також основи дизайну та роботу у графічних редакторах, щоб створювати графіку та персоналізувати шаблони.

Back-end – програмна сторона веб-сторінки, невидима для користувача, яка відповідає за реалізацію відповідних сценаріїв, створення сторінок, власне з нею працюють розробники та адміністратори сайту, пов'язана з написанням скриптів для сервера.

Бекенд — «невидимий двигун» сайту.

Бекенд-розробники пишуть код, використовуючи такі популярні мови програмування, як PHP, Ruby on Rails, Python, .NET та інші. Про трендові новинки у сфері програмування періодично можна спостерігати на сайті Hacker News Hiring Trends. Коли потрібно виконати якусь операцію, бекенд-код

взаємодіє з базою даних (використовуючи MySQL, SQL, Microsoft Access і т. д.). Після цього необхідна інформація повертається користувачу в формі фронтенд-коду.

Який би приклад підійшов у цьому випадку? Коли ви ведете автомобіль, ви використовуєте кермо, щоб привести його в рух і дати інші важливі команди. Але те, що дійсно приводить авто в рух, заховане всередині. Це його двигун. Займаєтесь онлайн-шопінгом? Підписуєтесь на розсилку? Редагуєте контент на сторінці? Яку б операцію ви не виконували, врешті-решт за неї відповідає бекенд-код. Інформація оновлюється, змінюється або видаляється в базі даних.

Створення та редагування даних першим спадає на думку, коли справа стосується бекенд розробки. Такий IT-спеціаліст вивчає мови скриптування: PHP, Ruby та Python. За їхньою допомогою він створює алгоритми та вибудовує логіку даних, щоб налаштувати адекватну роботу додатка чи сайту у відповідь на отриманий від користувача запит. Це означає, що back-end розробник повинен вміти писати код для отримання інформації та її збереження, як у базі даних. Спеціаліст дійсно вдається до створення баз даних – в основному, реляційних (PostgreSQL та MySQL) та іноді – нереляційних (Mongo). Для цього використовується мова SQL. Розуміння серверного менеджменту допомагає розробнику вирішувати проблеми з повільним завантаженням та вносити корективи, щоб залучити до сайту якомога більше користувачів.

Так само з фронтендом і бекендом.

Фронтенд-розробка зосереджена на тих елементах сайту, які користувач бачить у браузері і з якими безпосередньо взаємодіє.

Бекенд-розробка відповідає за функціонал сайту і має справу з речами, яких ви не бачите, — такими як бази даних та сервери.

Отже, як фронтенд, так і бекенд важливі важелі для розробки сайту.

3.4. Розміщення веб-сайту на локальному віртуальному середовищі або в Інтернеті

Розміщення веб-сайту просто означає переконатися, що ваш веб-сайт може бути доступний у всесвітній мережі (WWW). Зазвичай це робиться одним із двох способів. Ви можете заплатити за хостинг у постачальника послуг або самостійно розмістити його на власному сервері.

Файли веб-сайтів, такі як HTML, зображення, відео, зберігаються на серверах, підключених до Інтернету. Коли користувачі захочуть відвідати ваш веб-сайт, вони введуть адресу вашого веб-сайту у свій браузер, а потім їх комп'ютер підключиться до вашого сервера. Потім ваші веб-сторінки будуть доставлені користувачам через Інтернет-браузер.

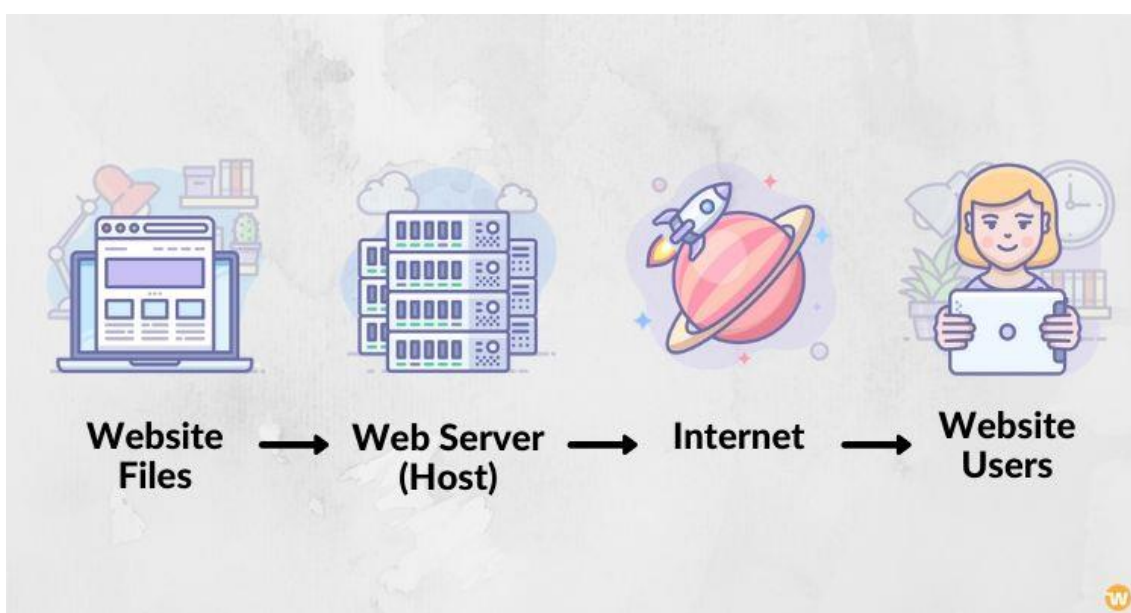


Рис. 3.1. Як працює хостинг

Використання постачальника послуг - найпростіший спосіб розміщення веб-сайту. Ви можете сплатити невелику щомісячну плату і покластися на постачальника послуг, щоб подбати про все обладнання, інфраструктуру та інші пов'язані з цим потреби.

Плюси хостингу у постачальника послуг:

- Зазвичай дешевше;
- Підтримка часто доступна;
- Немає необхідності в технічному обслуговуванні;
- Більш висока надійність.

Мінуси хостингу у постачальника послуг:

- Можуть бути деякі сервісні обмеження;
- Менше варіантів розміщення локацій.

Етапи розміщення веб-сайту з постачальником послуг хостингу:

1. Вирішіть, який тип веб-сайту ви будете.

Існує два основних типи веб-сайтів; статичний і динамічний.

Прості статичні веб-сайти можна створити за допомогою програми «Що ви бачите, що ви отримуєте» (WYSIWYG), а потім перенести на обліковий запис хостингу.

Динамічні сайти в основному керуються додатками і використовують сценарії, бази даних та інші інструменти для генерації деяких частин сайту на ходу. WordPress та Joomla - приклади поширених програм CMS (Content Management System), які популярні сьогодні. Інші, такі як Magento і PrestaShop, використовуються для веб-сайтів електронної комерції.

2. Порівняйте типи веб-хостингу.

Наче багато існує різних категорій автомобілів, хостинг веб-сайтів також випускається з різними смаками. Наприклад, спільний хостинг – це найдешевший і найпростіший в управлінні - вони схожі на компактні машини світу.

Оскільки тип веб-хостингу збільшується, так само зростає пов'язані з цим витрати і часто складність управління обліковим записом хостингу. Наприклад,

в VPS-хостинг вам потрібно буде керувати не лише деталями хостингу, але й середовищем, в якому він розміщений.

3. Виберіть та придбайте план веб-хостингу.

Навіть у межах типів хостингу, у постачальників послуг часто доступні різноманітні плани. Ключова відмінність цих планів часто полягає у кількості ресурсів, які кожен отримує. Чим більше ресурсів має ваш сайт, тим більше відвідувачів він може працювати.

Що стосується ресурсів веб-хостингу, ми зазвичай маємо на увазі три основні елементи - процесор (процесор), пам'ять (оперативна пам'ять) та накопичувач (HDD або SSD). Однак це не завжди означає хорошу роботу веб-хоста.

У минулому не було простого способу оцінити ефективність роботи веб-хостингу. Більшість користувачів доводиться розраховувати на огляди, які, на жаль, зазвичай роблять лише знімки продуктивності хоста та рідко їх оновлюють. Щоб пройти це, спробуйте скористатися HostScore, веб-сайт, який постійно оцінює ефективність роботи веб-хостів на основі поточного збору даних. Це означає, що оцінки їх роботи на веб-хості набагато точніші.

Також слідкуйте за додатковими цінностями, такими як безкоштовний SSL, доменне ім'я, рекламні кредити, доданий розробник веб-сайтів або інші предмети, які можуть допомогти вам створити або продати свій сайт.

4. Придбати домен та план.

Там, де ваш веб-хостинг - це фактичне місце, на якому сидять файли вашого веб-сайту, вам потрібно доменне ім'я, щоб користувачі могли отримати доступ до вашого сайту. Ім'я домену діє як ваша адреса на WWW. Як і справжні адреси, кожен унікальний.

Багато планів веб-хостингу сьогодні матимуть безкоштовне доменне ім'я, тому переконайтеся, що це стосується веб-хостингу, який ви маєте намір

придбати. Якщо це так, ви можете подбати про доменне ім'я одночасно, коли платите за свій веб-хостинг.

Якщо ні, то вам потрібно буде купити доменне ім'я окремо. Це можна зробити з того самого місця, де ви придбали план хостингу, або іншого постачальника послуг. Якщо вам потрібно придбати доменне ім'я окремо, настійно рекомендую заглянути в іншому місці.

Імена домену не є фіксованими ціновими елементами і часто надходять у продаж. Деякі провайдери часто мають дешеві продажі на доменні імена, і якщо пощастить, ви можете забрати його на крадіжку. Наприклад, часто в імені доменних імен пропонувались до 98% знижки.

Виняток із цього випадку, якщо ви власник сайту вперше. У такому випадку покупка доменного імені та хостинг у того самого постачальника послуг може полегшити вам роботу з початківцем.

5. Перемістіть / створіть свій веб-сайт на сервері.

Як тільки ваше доменне ім'я та план веб-хостингу будуть готові, настає час міграції. Міграція сайту може бути складною, тому, якщо ви робите це вперше, зверніться за допомогою до нового хоста. Деякі постачальники послуг хостингу пропонують безкоштовні міграції сайту.

Якщо ви створили свій веб-сайт локально (на своєму комп'ютері), просто перенесіть наші файли на ваш веб-сервер. Для цього ви можете або скористатися файловим менеджером на панелі керування веб-хостингу, або здійснити передачу за допомогою FTP-клієнта.

Процес схожий на копіювання файлів з одного місця в інше на власному комп'ютері.

Розміщення веб-сайту на локальному рівні означає, що ви фактично використовуєте власне місцезнаходження для налаштування веб-сервера з нуля. Це означає, що ви несете відповідальність за все, починаючи від апаратного та

програмного забезпечення аж до забезпечення пропускної здатності та інших інфраструктурних потреб.

Плюси самостійного хостингу:

- Надзвичайний контроль над вашим хостинг-середовищем;
- Потенціал для швидшого часу обслуговування;
- Ваш вибір постачальників обладнання та послуг.

Мінуси самостійного хостингу:

- Може бути дуже дорогим;
- Це не завжди можливо в житлових умовах.

Розміщення веб-серверів на локальному рівні є складним і може бути дуже дорогим. Він також часто менш надійний, ніж хостинг з постачальником послуг.

1. Виберіть обладнання та програмне забезпечення.

Базове обладнання для сервера може бути дуже схожим на обладнання на вашому ПК з деякими незначними відмінностями. Насправді технічно ви можете взяти власний ПК (або навіть ноутбук) і перетворити його на домашній веб-сервер, якщо цього дуже хотіли.

Ключова відмінність полягає в тому, наскільки надійним ви хочете, щоб ваш веб-сервер був, і кількість відвідувачів, з якими він може працювати. Як і у випадку з постачальниками послуг, вам потрібно буде стежити за процесором, пам'яттю та місцем для зберігання.

Якщо ви вибрали серверне обладнання високого класу, наприклад, стійковий сервер, вам також потрібно буде забезпечити задоволення спеціалізованих потреб цього обладнання. Це включає в себе простір, охолодження та потужність.

2. Забезпечте достатню пропускну здатність.

Пропускна здатність Інтернету також важлива для роботи вашого власного сервера. У багатьох випадках більшість із нас чудово застосовують стандартну пропускну здатність Інтернету, оскільки ми використовуємо обмежені підключення до Інтернету. Уявіть, якби 30 людей намагалися одночасно користуватися вашим домашнім Інтернетом - це, а може, й більше, те, що ви, можливо, хочете підтримати.

Що також потрібно врахувати, це ваша IP-адреса. Більшість домашніх інтернет-планів мають динамічні IP-адреси. Для того, щоб запустити веб-сервер, вам потрібен статичний IP. Цим може впоратися постачальник послуг, наприклад DynDNS або придбавши послугу у свого Інтернет-провайдера (ISP).

3. Розробка та розгортання веб-сайту.

Наступна частина схожа на досвід використання провайдера веб-хостингу, за винятком того, що ви не отримуєте жодної підтримки. Ваші веб-файли потрібно перемістити на ваш веб-хост, щоб ваш сайт почав функціонувати.

ВИСНОВКИ

У результаті опрацювання Розділу 1 було досліджено особливості використання інформаційних систем. У сучасному світі необхідність використання інформаційних систем у веденні бізнесу детермінована тим, що ведення оперативного управління, планування та обліку досить складно здійснювати без процесу автоматизації. Накопичення протягом кожного року інформації ускладнює процес прийняття рішень без інформаційних систем. Інформаційні системи забезпечують швидку взаємодію підприємства з контрагентами, що дає можливість здійснювати оцінювання рівня ділової активності, фінансової стійкості та прибутковості. Тобто нині вони є засадничим інструментом, що сприяє веденню бізнесу у будь-якій галузі. Інформаційна система – це система, яка складається із організаційних та технічних засобів з метою збереження та оброблення інформаційних потоків суб'єкта господарювання. Оскільки інформація про зовнішнє, внутрішнє та операційне середовище постійно циркулює у вигляді інформаційних потоків, які характеризуються різною інтенсивністю, різнонаправленістю і наявністю зв'язків і обмінів, то важливо так імплементувати інформаційну систему на підприємстві (установі, організації), щоб забезпечувати безперебійний процес циркуляції інформаційних потоків.

В ході виконання практичного завдання було створено нову базу даних. Було визначено елементи бази даних, між яким встановлено зв'язки. Для реалізації бази даних було обрано систему управління базами даних MySQL. Розроблена система була ретельно протестована. Гідність бази даних полягає в простоті, зрозумілості та зручності. Саме простота і зрозумілість для користувача з'явилися основною причиною їх широкого використання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Dash, Python [Електронний ресурс]. – Режим доступу: <https://dash.plot.ly>.
2. Gosling The Java Language Specification Java SE 8 Edition [Електронний ресурс]. – Режим доступу: <http://docs.oracle.com/javase/specs/jls/se8/jls8.pdf>
3. Єдинак В.С. Розвиток інформаційних технологій в Україні//Наукові доробки молоді - вирішенню проблем європейської інтеграції: збірник наукових статей. В 2 т. Т. 1 – Харків: Континент, 2015. – С. 289 -290.
4. Какорін М.О. Інформаційні технології як фактор інновацій у глобальній фінансовій системі // Формування ринкових відносин в Україні: Збірник наукових праць. – №5. – 2016. – С. 106 -109.
5. Клименко А. Профессии в сфере использования информационных технологий //Менеджер по персоналу. – 2016. – № 11. – С. 46 -56
6. Ковалева Т. Ю. Статистическое исследование динамики и структурных изменений на рынке труда [Электронный ресурс] / Т. Ю. Ковалева // Проблемы современной экономики. – 2011. – № 2 (38). – Режим доступу: <http://www.m-economy.ru/art.php?nArtId=3577>.
7. Львівський ІТ -кластер [Електронний ресурс]. – Режим доступу: <http://itcluster.lviv.ua/about-us/about-cluster/>
8. Пацай Б.Д. Роль інформаційних технологій в управлінні фінансовими ресурсами підприємств// Фінанси України. – № 8. – 2014. – С.82 -84.
9. Сіленко А. Інформаційні технології - новий імпульс для пошуку парадигми майбутнього суспільства //Політичний менеджмент. – 2014. –№ 3. – С. 96 -112.