ЗМІСТ

[ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ 8](#_Toc419402179)

[ВСТУП 9](#_Toc419402180)

[1 ФОТОГРАФІЯ. АНАЛІЗ ТА ДОСЛІДЖЕННЯ ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ ПРЕДСТАВЛЕННЯ ФОТО РОБІТ 10](#_Toc419402181)

[1.2 Мистецтво фотографії та процес її виникнення 10](#_Toc419402182)

[1.2 Аналіз веб ресурсу BEST-WEDDING 17](#_Toc419402183)

[1.3 Огляд персональних сайтів, переваги та їх недоліки 20](#_Toc419402184)

[1.4 Аналіз спеціалізованих інтернет спільнот 24](#_Toc419402185)

[2 ОГЛЯД ІНСТРУМЕНТІВ ТА ЗАСОБІВ WEB РОЗРОБКИ 26](#_Toc419402186)

[2.1 Базові відомості про Internet та WEB 26](#_Toc419402187)

[2.2 Скриптова мова програмування PHP 32](#_Toc419402188)

[2.3 Каскадні таблиці стилів CSS 43](#_Toc419402189)

[2.4 Twitter Bootstrap 45](#_Toc419402190)

[3 РОЗРОБКА програмного забезпечення веб-ресурсу формування рейтингу професійних фотографів та їх робіт 47](#_Toc419402191)

[3.1 Розробка прототипу інтерфейсу, функціоналу та БД розроблюваного проекту 47](#_Toc419402192)

[3.2 Верстання адаптивного інтерфейсу засобами веб розробки 47](#_Toc419402193)

[3.3 Створення та програмування БД ресурсу 47](#_Toc419402194)

[3.4 Розробка функціональної частини проектованого продукту 47](#_Toc419402195)

[3.5 Опис функціональних можливостей та ілюстрація роботи створеного веб додатку 47](#_Toc419402196)

[4. ЕКОНОМІЧНА ДОЦІЛЬНІСТЬ РОЗРОБКИ ТА ВИКОРИСТАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ 48](#_Toc419402197)

[4.1. Обґрунтування ринкової доцільності розробки веб ресурсу фотографів 48](#_Toc419402198)

[4.2. Розрахунок собівартості й ціни розробки (вказати назву розробки) 50](#_Toc419402199)

[4.3. Розрахунок можливого прибутку фірми-розробника та терміну окупності розробки. 52](#_Toc419402200)

[4.4 . Аналіз конкурентоспроможності розробки (вказати розробку) 55](#_Toc419402201)

[5 ОХОРОНА ПРАЦІ 58](#_Toc419402202)

[5.1 Значення охорони праці для користувачів ПК 58](#_Toc419402203)

[5.2 Забезпечення нормальних умов праці та Розрахунок освітленості робочого місця 62](#_Toc419402204)

[5.3 Забезпечення електробезпеки 68](#_Toc419402205)

[5.4 Пожежна безпека 71](#_Toc419402206)

[ВИСНОВКИ 73](#_Toc419402207)

**СПИСОК ПОСИЛАНЬ НА ДЖЕРЕЛА………………………….............**

**ДОДАТКИ**

**БІБЛІОГРАФІЧНА ДОВІДКА**

# ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

PHP – Hypertext Preprocessor

HTML – Hyper Text Markup Language

CSS – Cascading Style Sheets

jQuery -JavaScript-бібліотека

ORM – Object-relational mapping

URL – Uniform Resource Locator

UML – Unified Modelling Language

MVC – Model-view-controller

Kohana- Веб-фреймворк на мові PHP

Auth – Модуль авторизації

СУБД – Система управління базами даних

ПК – Персональний комп’ютер

ЕОМ – Електронно-обчислюванні машини

phpMyAdmin – Веб-застосунок з відкритим кодом на мові PHP

Git – Розподілена система керування версіями файлів

# ВСТУП

На сучасному рівні розвитку інформаційних технологій використання комп’ютера для збереження та представлення будь-яких видів інформації стає єдиним засобом, що надає широкі можливості керування інформацією. Важливу роль у процесі отримання інформації відіграє мережа Інтернет. Сьогодні в Україні послугами Інтернет з різною періодичністю користуються близько 9 млн. жителів України. Серед яких є і велике коло фотографів. Internet сьогодні це найбільш розвинена у світі інформаційна система, за допомогою якої здійснюється комунікація між мільйонами користувачів. За допомогою мережі Internet забезпечується доступ до більш як п'яти мільйонів інформаційних Web-сайтів. Якщо прийняти до уваги кількісні показники українського сегменту Internet, то вони налічують 200-300 тисяч користувачів, загальна кількість Web-серверів на сьогодні досягла позначки 4,5 тисяч.

З самого початку розвитку Internet, а особливо з появою Web-технологій, мережа орієнтована на інформаційне забезпечення своїх користувачів. Цим вдало навчилися користуватися і фотографи, які за допомогою Web-сайтів можуть представляти свої роботи та мати можливість підтримувати он-лайн зв'язок з потенційними замовниками.

Метою даного дипломного проекту є “Розробка алгоритмічного та програмного забезпечення веб-ресурсу формування рейтингу професійних фотографів та їх робіт ”.

На сьогодні представлення своїх робі фотографами у мережі Internet є необхідним для підвищення кількості клієнтів та отримання замовлень на виконання даного роду робіт. Розроблюваний ресурс покращить обмін інформацією між усіма учасниками ринкового процесу та способом заявити про свою діяльність великому загалу користувачів глобальної мережі.

# 1 ФОТОГРАФІЯ. АНАЛІЗ ТА ДОСЛІДЖЕННЯ ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ ПРЕДСТАВЛЕННЯ ФОТО РОБІТ

## 1.2 Мистецтво фотографії та процес її виникнення

Бажання зберегти красу швидкоплинною життя створило дивовижний інший вид мистецтва – фотографію. Історія фотографії – це захоплююча історія зародження і впровадження мрії про фіксації і тривалому збереженні зображень навколишніх явищ і предметів, одне із найбільш яскравих та буремних етапів розвитку сучасної інформаційної технології.

Історія фотографії [1] почала розроблятися лише останнім часом, в основному в післявоєнні роки. У різних країнах саме в цей час стали виходити книги, присвячені як вітчизняної фотографії, так і зарубіжним фотографічним школам. Довгий час за фотографією заперечувалося право на естетичне творчість. Художники і мистецтвознавці минулого століття розглядали світлопис як механічну копію дійсності, здатну бути лише сурогатом живопису. Для того щоб хоч якось подолати бездушність технічного засобу - фотоапарата, багато фотографів минулого витрачали чимало сил і уяви, створюючи знімок неадекватний фіксованій натурі. Вони широко використовували монтаж і печатку з декількох негативів, чаклували з пензлем у руках над обробкою відбитка, складали і ставили, подібно театральним режисерам, сцени на міфологічні сюжети.

Історія фотографії починається в глибокій старовині. Люди завжди знали, що від сонячних променів темніє людська шкіра, іскряться опали і аметисти, псується смак пива. Оптична історія фотографії налічує приблизно тисячу років. Найпершу камеру-обскуру можна назвати «кімнатою, частина якої освітлена сонцем». Арабська математик і вчений X століття Альгазена з Басри, який писав про основні принципи оптики і вивчав поведінку світла, зауважив природний феномен перевернутого зображення. Він бачив це перевернуте зображення на білих стінах затемнених кімнат або наметів, поставлених на сонячних берегах Перської затоки, - зображення проходило через невеликий круглий отвір в стіні, у відкритому запоні намети або драпірування. Альгазена користувався камерою-обскура для спостережень за затемненнями сонця, знаючи, що шкідливо дивитися на сонце неозброєним оком.

Першою людиною, хто довів, що світло, а не тепло робить срібну сіль темною, був Йоганн Генріх Шульце (1687-1744), фізик, професор Галльського університету в Німеччині. У 1725 році, намагаючись приготувати світиться речовина, він випадково змішав крейду з азотною кислотою, в якій містилося трохи розчиненого срібла. Він звернув увагу на те, що коли сонячне світло потрапляло на білу суміш, то вона ставала темною, тоді як суміш, захищена від сонячних променів, абсолютно не змінювалася. Потім він провів декілька експериментів з буквами і фігурами, які вирізав з паперу і накладав на пляшку з приготованим розчином, - виходили фотографічні відбитки на посріблені крейді. Професор Шульце опублікував отримані дані в 1727 році, але у нього не було й думки постаратися зробити знайдені подібним чином зображення постійними. Він збовтував розчин у пляшці, і зображення пропадало. Цей експеримент, тим не менш, дав поштовх цілій серії спостережень, відкриттів і винаходів в хімії, які через трохи більше століття привели до винаходу фотографії. Поступово, однак, все чіткіше і ширше стала поширюватися думка, згідно з якою фотомистецтво, якщо таке може існувати, повинно виникнути не на шляхах проходження Наприклад живопису, а на власній основі. На основі неповторних, тільки їй, фотографії, властивих творчих здібностей. І тут знову дало себе знати незвичайне властивість камери запам'ятовувати життя в її документально-справжньої формі. Стало очевидним, що та мова, якою так добре користується фотожурналістика, і є специфічний для всякої фотографії, в тому числі і претендує на естетичне визнання.

Такий рішучий поворот в уявленнях про фото творчість змусив істориків фотографії переглянути поняття про становлення нового виду творчості, по-новому оцінити роль окремих фотографів, переглянути список імен тих майстрів, про які можна говорити як про художників світлопису. Тепер в історію фотографії як персони першорядної важливості увійшли ті, кого ще вчора вважали далекими від естетичної творчості журналістами.

Перше закріплене зображення було зроблено в 1822 році французом Жозефом Нісефор Ньєпс (Nicéphore Niepce), але воно не збереглося до наших днів. Тому першою в історії фотографією вважається знімок «вид з вікна», отриманий Ньепсом в 1826 році за допомогою камери-обскури на олов'яної платівці, покритої тонким шаром асфальту. Експозиція тривала вісім годин при яскравому сонячному світлі. Перевагою методу Ньепса було те, що зображення виходило рельєфним (після протруєння асфальту), і його легко можна було розмножити в будь-якому числі примірників.

У 1839 році француз Луї-Жак Манда Дагер (Jacques Daguerre) опублікував спосіб одержання зображення на мідній пластині, покритої сріблом. Після тридцяти хвилинного експонування Дагер переніс пластину в темну кімнату і якийсь час тримав її над парами нагрітої ртуті. В якості закріплювача зображення Дагер використовував кухонну сіль. Знімок вийшов досить високої якості - добре пророблені деталі як у світлі, так і в тінях, однак, копіювання знімка було неможливо. Свій спосіб одержання фотографічного зображення Дагер назвав дагеротипія.

Незабаром вчені, художники і любителі поліпшили процес Дагера. Вони скоротили час експозиції до декількох хвилин. Застосування призми дало можливість зробити зображення на дагеротипією не дзеркальна, а нормальним. Дагеротипією передавали найдрібніші деталі орендованих об'єктів. До 1841 була створена камера меншого розміру, а її вага зменшилася в 10 разів. Були створені засоби, що оберігали поверхню дагеротипів від пошкоджень і подряпин.

Слава і визнання Дагера росли в міру того, як поширювався по світу його спосіб отримання зображень. Але після опублікування даних про своє процесі він вже не вніс нічого нового в фотографію. До своєї смерті в 1851 він жив в самоті недалеко від Парижа.

Практично в той же самий час англієць Вільям Генрі Фокс Тальбот винайшов спосіб отримання негативного фотографічного зображення, який назвав калотипією. В якості носія зображення Тальбот використав папір, просочений хлористим сріблом. Ця технологія поєднувала в собі високу якість і можливість копіювання знімків (позитиви друкувалися на аналогічній папері). Експозиція тривала близько години, на знімку - гратчасте вікно будинку Тальбота.

Сам термін «фотографія» [2] з'явився в 1839 році, його використовували одночасно і незалежно два астронома - англійська, Вільям Гершель, і німецька, Йоганн фон Медлер.

На сьогоднішній час виділяють такі основні види фотографії.

Чорно-біла фотографія - історично перший вид фотографії. Після появи кольоровий, а потім і цифрової фотографії, чорно-білі знімки зберегли свою популярність. Найчастіше кольорові фотографії перетворюються в чорно-білі для отримання художнього ефекту.

Рання кольорова фотографія була зроблена в 1912 році. Кольорова фотографія з'явилася в середині XIX століття. Перший стійкий кольоровий фотознімок був зроблений в 1861 році Джеймсом Максвеллом за методом триколірної фотографії (метод кольороподілу).

Для отримання кольорового знімка з цього використовувалися три фотокамери з встановленими на них кольоровими світлофільтрами (червоним, зеленим і синім). Отримані знімки дозволяли відтворити при проекції (а пізніше, і у пресі) кольорове зображення.

Поряд з методом кольороподілу з початку XX століття стали активно розвиватися й інші процеси (методи) кольорової фотографії. Зокрема, в 1907 році були запатентовані і надійшли у вільний продаж фотопластини «Автохром» Братів Люм'єр, що дозволяють відносно легко отримувати кольорові фотографії. Незважаючи на численні недоліки (швидке вицвітання фарб, крихкість пластин, зернистість зображення), метод швидко завоював популярність і до 1935 р в світі було вироблено 50 млн.автохромних пластинок.

Альтернативи цієї технології з'явилися тільки в 1930-х роках: Agfacolor в 1932 році, Kodachrome в 1935, Polaroid в 1963 нею стала цифрова фотографія.

Цифрова фотографія - відносно молода, але популярна технологія, що зародилася в 1981 році, коли компанія Sony випустила на ринок камеру Sony Mavica з ПЗС-Матрицею, записуючої знімки на диск. Цей апарат не був цифровим в сучасному розумінні (на диск записувався аналоговий сигнал), проте дозволяв відмовитися від фотоплівки. Перша повноцінна цифрова камера - DCS 100 - була випущена в 1990 році компанією Kodak.

Принцип роботи цифрової камери полягає у фіксації світлового потоку матрицею і перетворення цієї інформації в цифрову форму.

В даний час цифрова фотографія повсюдно витісняє плівкову в більшості галузей.

Особливе місце і значення фотографії в художній культурі пов'язано з технічної, наукової сутністю фотографії. Найважливішою властивістю фотографії є ​​її достовірність, справжність відображених подій. Одночасно з цим, зображення, як і в живописі чи малюнку, несе в собі художнє узагальнення, розкриття внутрішнього сенсу показаної ситуації, характер зображуваного людини та багато іншого.

По своїй суті, фотограф є художником, котрий володіють певними «фарбами» - фототехнікою та фотоматеріалами.

Фотограф використовує образотворчі засоби фотографії (точка зйомки, ракурс, лінійна композиція, план, перспектива, освітлення), родинні образотворчим засобам живопису. Додатковим інструментом є хіміко-фотографічна обробка.

Також одним із основних видів фотозйомки є репортажна зйомка, яку широко використовують при фотографування весіль, бенкетів та інших урочистих подій. До репортажної фотозйомки можна віднести роботу фотографа на концертах, виставках, корпоративних вечірках, спортивних матчах, святах, презентаціях, загалом кажучи - найрізноманітніших подіях. Про будь таку подію зберегти пам'ять допоможе фото. Тому що саме в цьому і полягає основна мета репортажного фото: відобразити певну подію в хронологічному порядку. Іншими словами, якщо Вам необхідна правдива об'єктивна фотозйомка, то Вам необхідний репортажний фотограф.

На перший погляд може здатися, що робота репортажного фотографа не складає особливих труднощів - знай собі спостерігай за подіями і тисни на кнопку. Насправді це зовсім не так. У фотографії репортажна зйомка - одна з найскладніших. Почнемо з того, що в такому вигляді фотозйомки, як репортажна, важливі дуже багато умінь фотографа. Крім вмілого поводження з фотокамерою, необхідні: швидка реакція і вміння бачити. І напевно, друге - найважливіше. Уміння бачити життя через об'єктив ... Репортажне фото - це якийсь розповідь про подію, який не просто викладає подію, а передає його атмосферу.

Особливості справжньою репортажної фотозйомки полягають у тому, що фотограф не може сказати своїй моделі: «Станьте, будь ласка, так. Поверніть голову». Репортажна зйомка - жива, у ній все по-справжньому. Ловити потрібно кожен момент. Секунда раніше чи секунда пізніше - і фоторепортаж вже не буде так вдалий, точніше фотографії стануть просто безглуздою пачкою паперу, де Ви не побачите самого цікавого.

У фоторепортажі розрізняють репортажну фотожурналістику і комерційну репортажну зйомку. У цій статті пропоную поговорити про другий.  
Отже, сьогодні з'явилося поняття комерційної репортажної фотографії. Тобто це - не просто вільний фоторепортаж, а фоторепортаж на замовлення, де фотографу виставляють певні рамки. Завдання такої репортажної зйомки ставляться замовником. Це може бути фотозйомка будь-якої події: весільної церемонії, презентації, корпоративу чи іншого заходу.

У чому особливості комерційної репортажної фотозйомки? Якщо це комерційна репортажна фотозйомка, то фотограф повинен знімати так, як цього хоче клієнт, як він це бачить. Звичайно, це дуже складно. Саме тому перед фотозйомкою фотограф обговорює деталі фотозйомки із замовником: на чому зробити акцент, хто повинен стати фаворитом зйомки і т.д. Очевидно одне, оскільки найчастіше, це фотозйомки будь-яких позитивних подій, то і фотографії повинні випромінювати позитив. Якщо це репортажна фотозйомка бізнес-семінару чи конференції, то відповідно фотографії повинні передати їх атмосферу. У будь-якому випадку перед подібною фотозйомкою фотограф обумовлює акценти із замовником.

Як би не дивно це звучало, в репортажної комерційної фотографії повинні бути присутніми елементи художності. Це стосується більшою мірою фотозйомки весіль, свят і т.д. Більше того, комерційна репортажна фотозйомка увазі не просто позитивні кадри, а й якісні різкі фотографії. Тут смислове навантаження кадру також важлива, як і його якість. І зробити це не завжди легко (думаю, багато фотографів з цим погодяться, знаючи з досвіду). Адже коли Ви намагаєтеся зловити приголомшливий емоційно-позитивний момент, складно встежити і за різкістю або виділити композицію. Так що в комерційній репортажної фотозйомки не так все легко, як може здатися на перший погляд.

З початком широкого застосування фотографії з’явилася необхідність у появі професії фотографа. Професія фотографа романтична, захоплююча і, безсумнівно, творча. Основне завдання - зафіксувати певний об'єкт за допомогою фотоапарата. Щоб фотографії виходили досить цікавими, виразними і, звичайно, якісними, необхідно володіти не тільки певними навичками зйомки і виготовлення знімків, але, в першу чергу, фантазією, багатою уявою і гарним смаком. У функції фотографа входить і безпосередній процес зйомки, та обслуговування обладнання, а також вибір оптимального освітлення, пози, фону і пошук «ідеї» фотографії.

Початком професії фотографа вважають середину 19 століття, коли з'явився фотографічний апарат, і, відповідно, людина, що вміє з ним поводитися. Сьогодні вже ніхто не очікує, коли ж, нарешті, вилетить «пташка» з громіздкого апарату, сучасна техніка дозволяє робити професійні знімки швидко і якісно.

Соціальна значущість професії в суспільстві: Знайти для себе роботу фотографи можуть в самих різних сферах діяльності. Це може бути журналістика та модельне агентство, традиційне фотоательє і робота в музеях або всіляких галереях. Рекламні агентства, різного роду лабораторії і, звичайно, криміналістика також потребують їх послугах, але й це ще не все. Адже свій день народження, шкільний випуск, весілля, корпоративну вечірку - все це ми хочемо відобразити на пам'ять. Крім того, нам потрібні якісні фотографії для різного роду документів - це, знову ж, робота для професійного фотографа.

Велике значення в цій професії відводиться терпінню, адже для отримання бажаного знімка часом доводиться провести кілька сесій, особливо це стосується випадків, коли ведуться зйомки непосидючих дітей або диких тварин. Крім того, фотографу потрібно вміти знаходити індивідуальний підхід до кожного свого клієнта, до кожного об'єкту.

Працювати фотографом сьогодні престижно і прибутково, але спочатку потрібно добре себе зарекомендувати і «завоювати» своїх клієнтів в досить серйозному конкурентному оточенні. Особлива краса цієї професії - вільний графік роботи і відсутність рутинних обов'язків.

Отже з розвитком фотографії та широким попитом на них є необхідність створення ресурсів, які допомагають фотографам представляти свої роботи для загалу. З розвитком цифрових технологій та розширення мережі internet стало доцільно розміщати свої роботи на різних веб ресурсах.

## 1.2 Аналіз веб ресурсу BEST-WEDDING

Одним з найпоширеніших в нашому регіоні веб порталом який дозволяє фотографам представити свої роботи є портал «best-wedding», який знаходиться за адресую <http://www.best-wedding.com.ua/>. Даний сайт є комплексним представленням всіх даних про:

* Фотографів;
* Відеооператорів;
* Ведучих на весілля;
* Салони суконь;
* Музикантів;
* Ресторани;
* Тощо.

Загальний вигляд цього ресурсу представлений на рисунку 1.1



Рисунок 1.1 – Загальний вигляд веб сайту

На даному рисунку ми можемо побачити загальний вигляд головної сторінки, з чого можемо судити що таке представлення не дозволяє сконцентрувати увагу безпосередньо на викладених на нього роботах, а подається тільки загальна інформація про деякі послуги. Ми бачимо що на даному сайті є ще безліч інших розділів а не лише розділ про фотографів, що можна сказати що даний сайт є універсальний, а не спеціалізований на певну групу користувачі. А це ускладнює процес пошуку саме фотографа на своє урочисте свято для збереження всіх незабутніх моментів у вигляді фотокарток.

При переході в розділ фотографи ми потрапляємо на сторінку де у випадковому порядку представлені фотографи та фотостудії та є короткий опис про них рисунок 1.2.

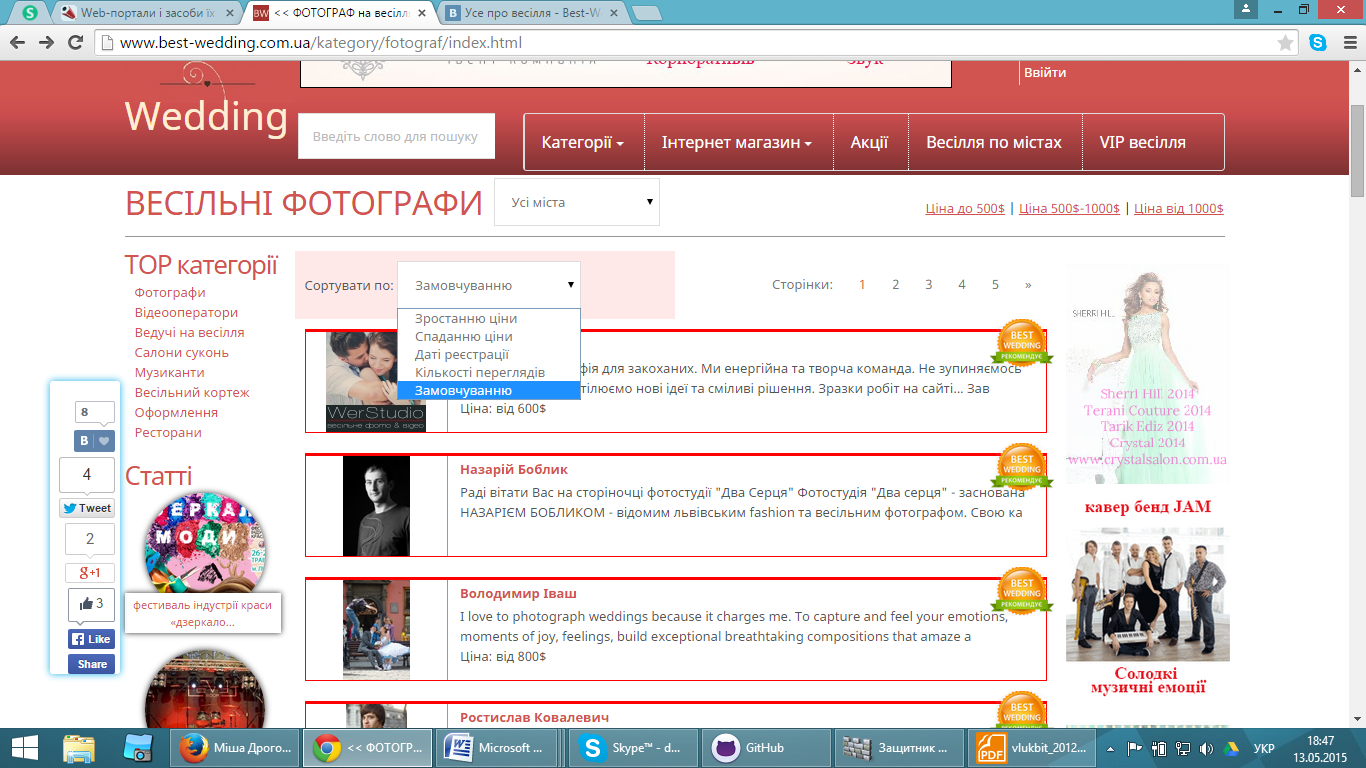


Рисунок 1.2 – Список фотографів та фотостудій

На цій сторінці ще є присутній фільтр, що дозволяє формувати список фотографів за ціною, датою реєстрації, кількістю переглядів та за замовчуванням. Але даний фільтр не дозволяє зробити регіональний пошук чи пошук популярністю. Асами ці фактори саме більше цікавлять потенційних замовників фотографів.

Після вибору фотографа ми потрапляємо у галерею де представлено фотороботи та контактна інформація того чи іншого фотографа рисунок 1.3.

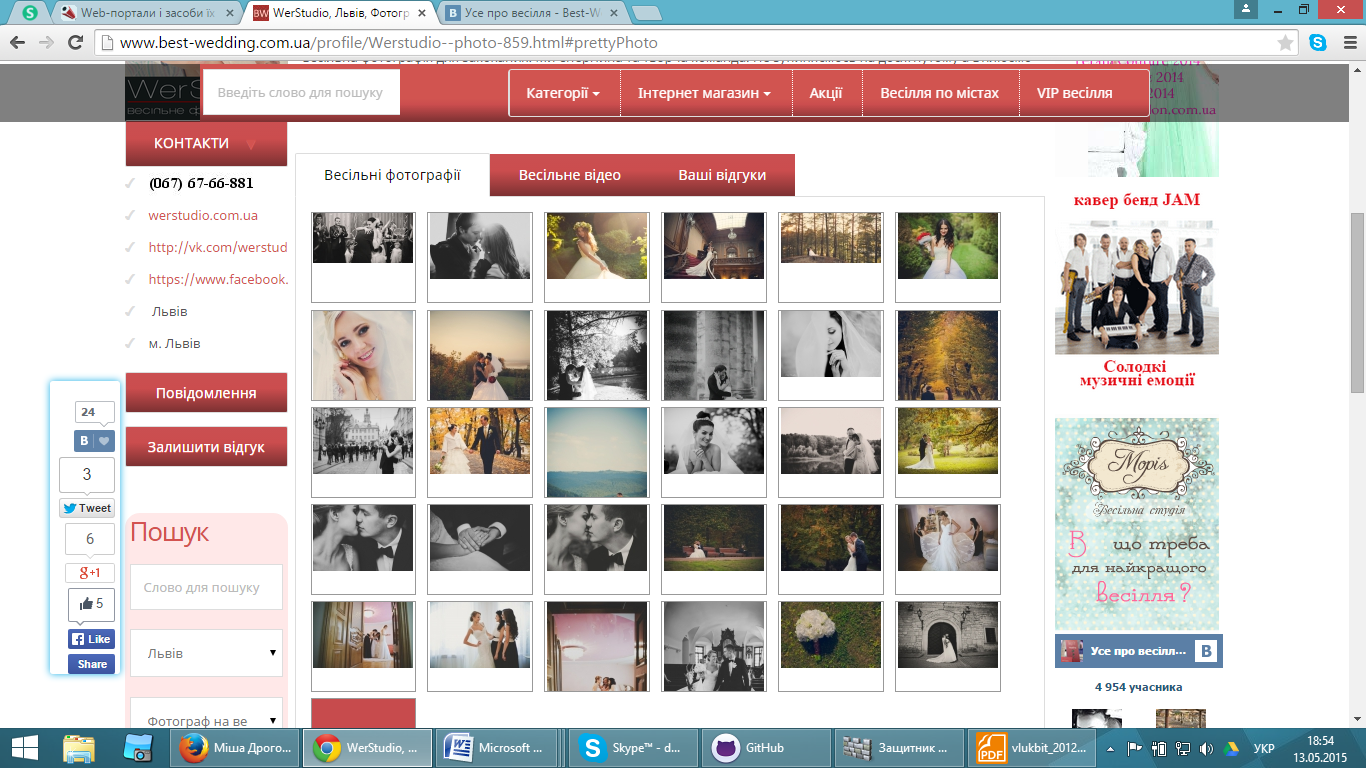


Рисунок 1.3 – Представлення фото робіт фотографа

Також одним із важливих факторів роботи фотографів з ресурсом є особистий кабінет фотографа, завдяки якому він може добавляти, редагувати та видаляти всю інформацію про себе та свої роботи. Для потрапляння в особистий кабінет необхідно зареєструватися або залогінитися на сайті best-wedding. Після чого ми потрапляємо в особистий кабінет де можемо керувати своєю інформацією. Особистий кабінет виглядає наступним чином так як представлено на рисунку 1.4.

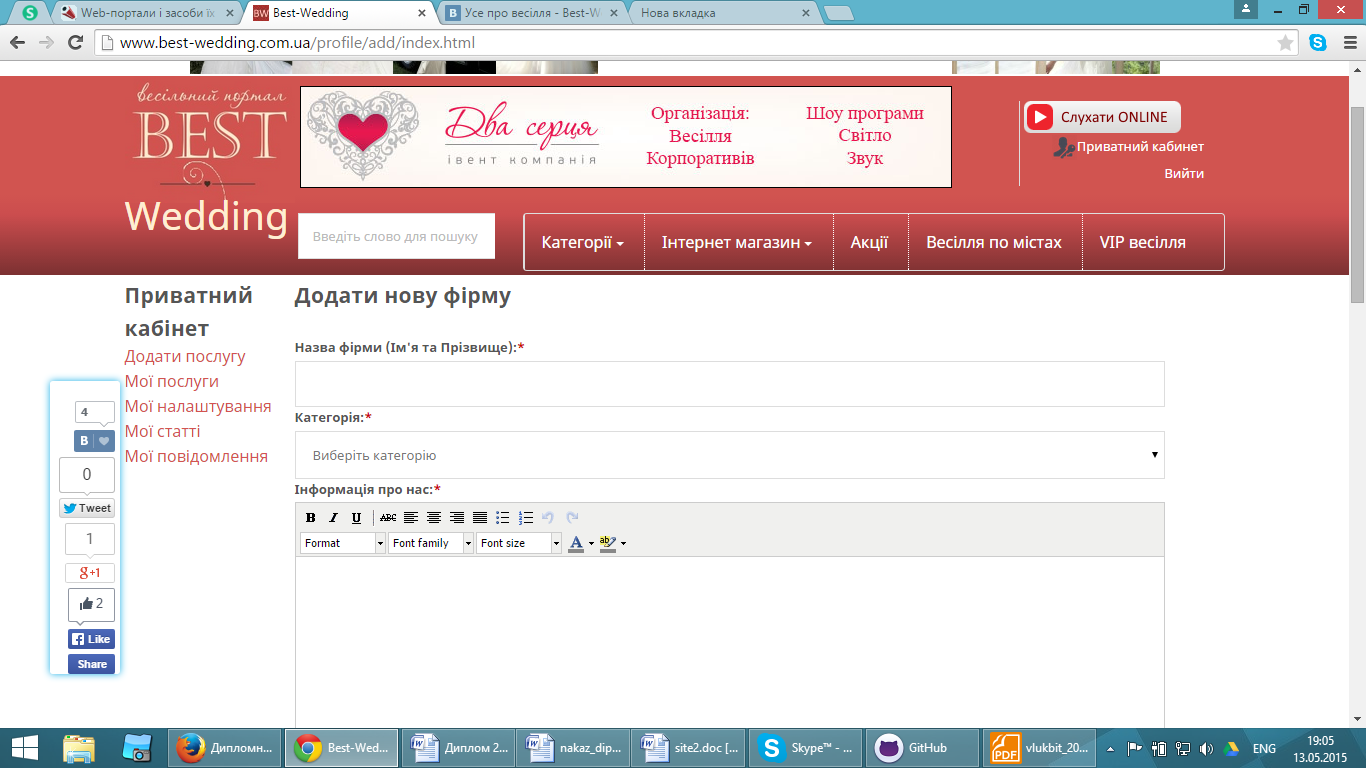


Рисунок 1.4 – Особистий кабінет

Отже провівши аналіз веб ресурсу “BEST-WEDDING” можна відзначити, що даний ресурс дозволяє фотографам представляти свої роботи широкому колу відвідувачів цього порталу. Але так як даний ресурс є не спеціалізований для вузького кола користувачів, то він містить безліч мінусів. До таких мінусів можна віднести:

* Безліч інших різних категорій користувачів;
* Відсутня можливість робити оцінку конкретної фотографії чи фотографа;
* Не великий перелік правил для фільтрування;
* Не впорядковане представлення фото робіт та можливість розподілу їх по каталогам;
* Випадкове формування списку фотографів;
* Відсутня можливість зв’язку між фотографом та відвідувачем який інтересуються отриманням послуг.

Підвівши підсумки можна сказати що даний ресурс не дозволяє повною мірою обрати найкращого фотографа на свою подію.

## 1.3 Огляд персональних сайтів, переваги та їх недоліки

За останній час сайти-візитки або(персональні сайти) придбали досить велике значення. Причина цього - відносно низька вартість їх розробки. Такий тип сайтів - прекрасні інвестиції, адже на наступний же день, якщо правильно спланувати рекламну кампанію, сайт зможе привести нових клієнтів. А якщо грамотно провести пошукове просування, сайт зможе стати стабільним джерелом доходу.

Сайт-візитка потрібний тим людям, які хочуть мати в мережі Інтернет візитну картку та можливість представлення широкому колу відвідувачів свої роботи, та надати всю необхідну інформацію про свою діяльність. Ця картка містить звичайно інформацію про фотографа, сферу діяльності, перелік основних послуги, ціни і т.д. Зазвичай сайт-візитка носить виключно рекламний характер і дає можливість отримувати інформацію про компанію або власника, знайомитися з її діяльністю, а також з наданими товарами і послугами.

Створити сайт-візитку - лише на перший погляд просто. За уявною простотою розробки сайту-візитки ховається багато кропіткої роботи. Найпростіші веб-сайти, які створюють на безкоштовному хостингу, є сторінками з персональною інформацією або рекламою якогось товару чи послуги. Такий сайт коштує лише часу, на нього витраченого. Але й віддача від такого сайту буде невелика. Адже навіть серед сайтів-візиток існує конкуренція, лідирують же сайти-візитки, що створюються в професійних студіях або агентствах. Чому? Як і в будь-якому іншому справі, в веб-дизайні діють певні стандарти та вимоги, яким повинен відповідати будь-який сайт. Ці стандарти допоможуть дотримати професійні веб-розробники.

Ціни на послуги зі створення сайту-візитки досить прийнятні. У той час як віддача від класно зробленого і нехай маленького, але цікавого та потрібного сайту не змусить себе довго чекати. Тому на сьогоднішній день багато фотографів створюють та використовують у своїх цілях персональні сайти для представлення своїх робіт.

У чому переваги сайту-візитки? Головна перевага такого ресурсу - невисока ціна, невеликий обсяг робіт і простота підтримки. Сайт-візитка зазвичай складається з декількох сторінок, в ньому немає складних програмних модулів, тому він не вимагає підтримки. Резервне копіювання такого сайту можуть виконувати самі власники, а також оновлювати і наповнювати свіжою інформацією. Тому сайт-візитка обійдеться його власнику лише в ціну розробки та оплату хостингу раз на місяць.

Тим не менш, сайт-візитка підходить для просування маленького приватних послуг або просто як персональна сторінка, а для організації корпоративного представництва середньої або великої компанії така форма сайту буде недостатньою. Сайт-візитка не може витримувати конкуренцію з більш потужними сайтами і у випадку пошукового трафіку. Причина тому - занадто невелику кількість сторінок, які можуть бути проіндексовані пошуковими роботами, невеликий обсяг даних і відсутність деяких маркетингових інструментів.

Як правило, сайти-візитки мають дуже мало функцій і складаються всього з 3-4 (іноді 5-7) сторінок, тому-то і він так недорого коштує. Тим не менш, деякі базові модулі в ньому використовують.

Наприклад:

* Модуль новин. Якщо необхідно регулярно розміщувати на сайті новинні матеріали, існує можливість сильно спростити завдання, встановивши новинний модуль. Додавання новин стане просто і легко.
* Модуль зворотного зв'язку. Такий модуль потрібний, щоб мати зв'язок між власником сайту та клієнтами.
* Версія для друку. Кожна сторінка буде містити спеціальну кнопку, при натисканні на яку сторінка буде відображатися в зручному для друку вигляді, без зайвих елементів сайту.

Для прикладу проведемо аналіз приватного сайту фотографа Наталі Каблюк [10], який виступає в ролі візитки для представлення її робіт та зв’язку з нею. Для цього необхідно в адресному рядку набрати адресу цього сайту, який знаходиться заадресую <http://nataliakabliuk.com/>. Після чого ми потрапимо на головну сторінку рисунок 1.1.

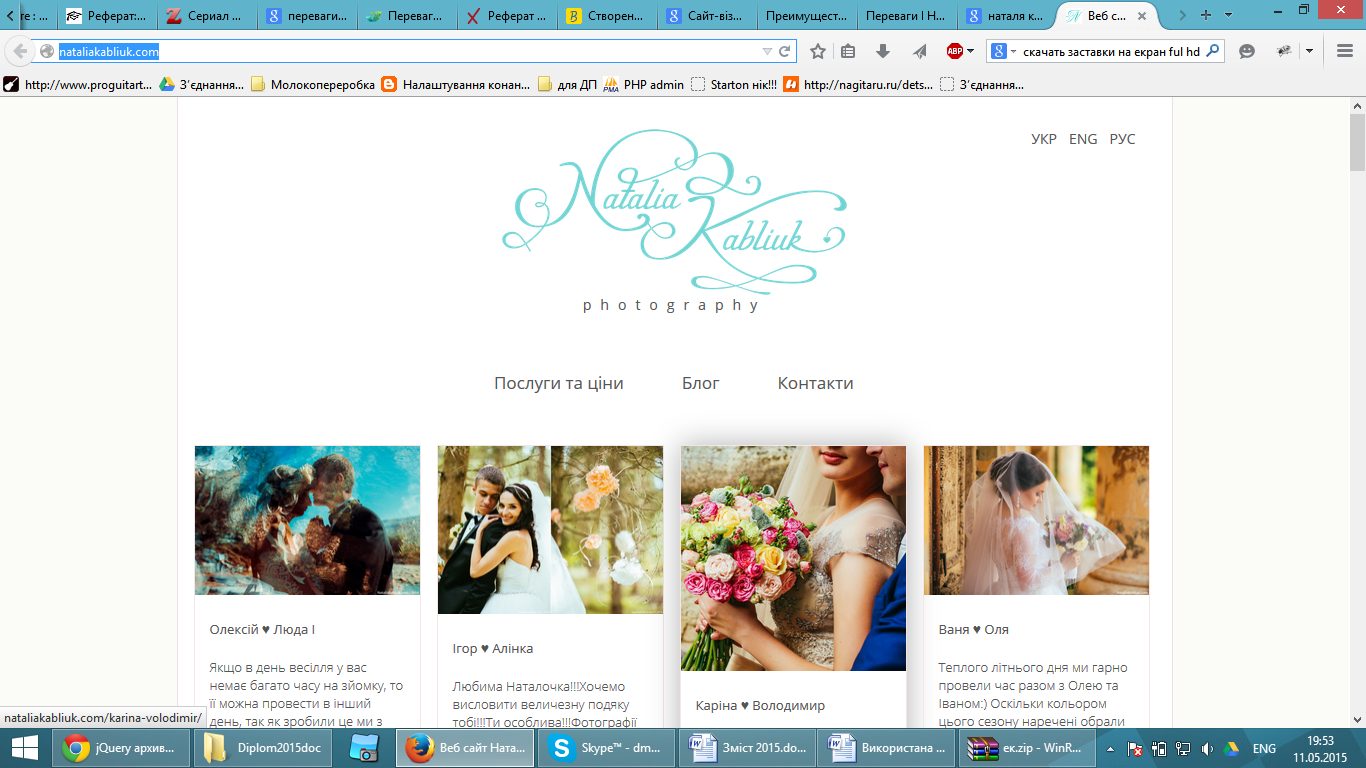


Рисунок 1.1. Головна сторінка сайту-візитки Наталі Каблюк

Що стосується загального вигляду сторінки, то відразу можна побачити що на ній представлені фотографії зроблені Наталею. Загальний вигляд сторінки зроблений продумано: є можливість переглянути роботи, ознайомитися з цінами на послуги. Почитати персональний блог та дізнатися контакти, за якими можна зв’язатися з фотографом. До недоліків можна віднести те що на цій сторінці не має можливості переглядати фотографії як фото галереє, а приходиться промотувати сайт за допомогою рядка прокрутки рисунок 1.2.

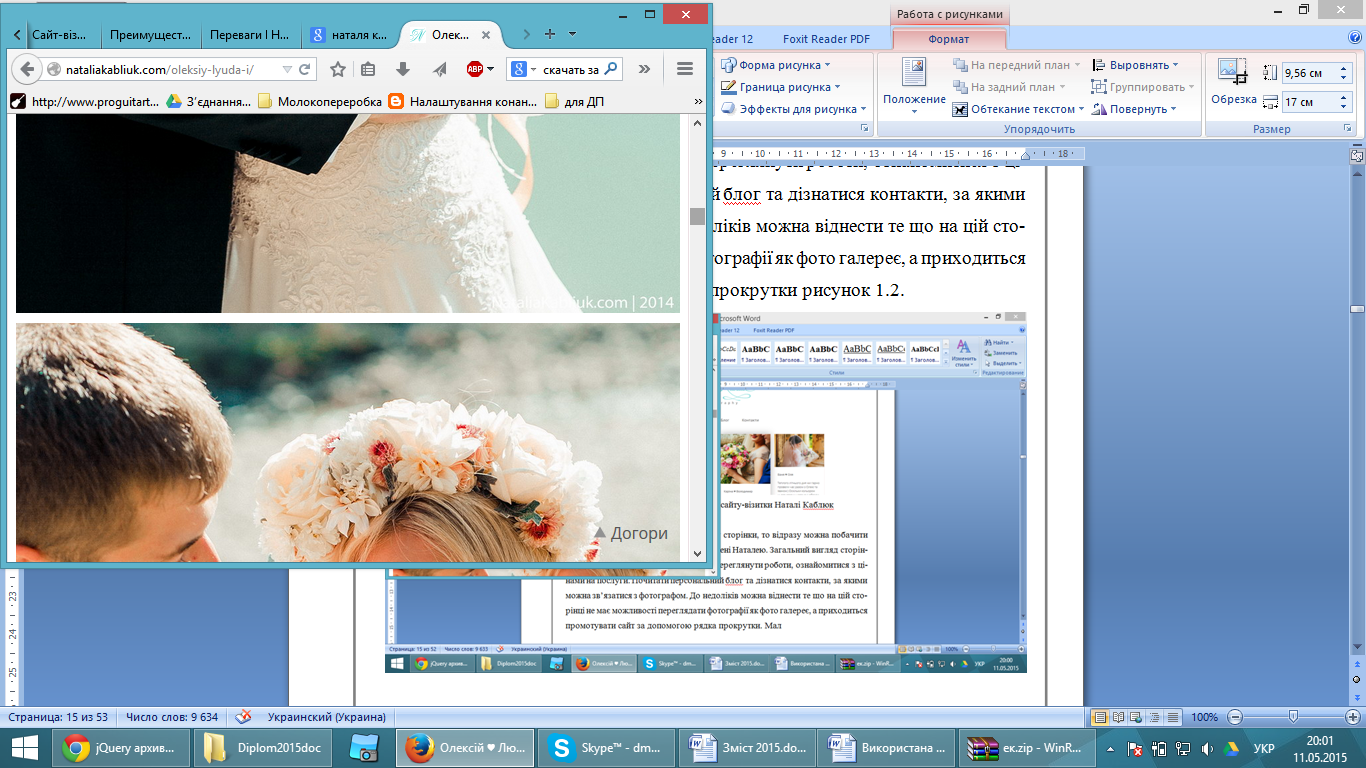


Рисунок 1.2. Перегляд галереї фотографії з допомогою рядка прокрутки

Також на даному веб ресурсі відсутня можливість зворотного зв’язку міх власницею та відвідувачами сайту. Це можна побачити на рисунку 1.3 на якому представлений лише список контактів.

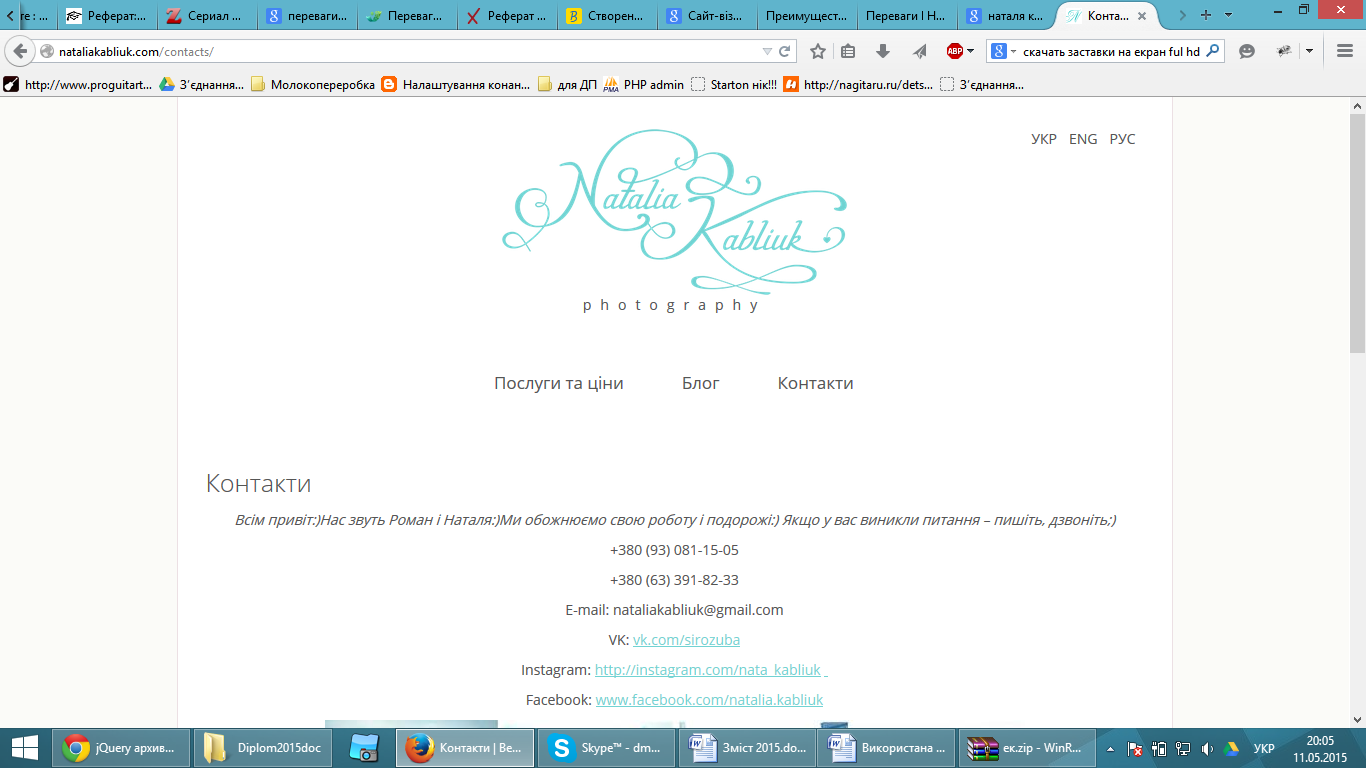


Рисунок 1.3. Сторінка контактів

Отже, переваг персональних сайтів над комплексними немає, так як кожен персональний сайт візитка, не має всіх необхідних функцій для представлення фото робіт та зв’язку відвідувачів з фотографами. Також пошук необхідного персонального сайту в пошуковій системі займає багато часу а інколи і не дає результату, що не можна сказати про такий проект який був розроблений на дипломну роботу. За допомогою розробленого проекту вирішуються проблеми які присутні персональним сайтам, а це не структурованість відображених даних, можливість зворотного зв’язку з фотографом, об єднання безліч фотографів та їх робіт на одному ресурсі, та можливість попереднього пошуку за вказаними параметрами значно коротить час на пошук потрібного працівника. Розроблюваний ресурс допоможе максимально якісно підібрати фотографа на ваше свято без затрати безлічі часу на відвідування сотні приватних сайтів-візиток.

## 1.4 Аналіз спеціалізованих інтернет спільнот

Фотографія сьогодні набуває все більшого інтересу в Україні, особливо у молодих людей. Однак для того, щоб стати справжнім фотографом замало одного фотоапарата і художнього таланту, необхідні ще й спеціальні знання та досвід. Де ж придбати подібні знання? В даний час на просторах глобальної мережі з'являється все більше спільнот фотографів, тобто об'єднань людей, що мають спільний інтерес і займаються фотографією.

Метою відкриття Інтернет-спільнот фотографів є об'єднання всіх, хто має відношення до світу фотографії, створення галерей авторських робіт, обмін досвідом для підвищення професійного рівня учасників спільноти і, звичайно ж, постійне спілкування між фотографами.

Інтернет-спільнота дає можливість вивчати прийоми і способи роботи інших фотографів, отримувати допомогу та цінні поради від професіоналів в області фотомистецтва, а також підвищувати свій власний імідж фотографа.

Брати участь або не брати участь у спільнотах фотографів - справа суто індивідуальна. Зрештою, кожен фотограф повинен бути індивідуальністю зі своїм власним сприйняттям навколишнього світу, і в цьому він цілком самодостатній. Інтернет-спільноти фотографів зазвичай організовуються за територіальним принципом (фотографи, які проживають в одному місті чи регіоні), за жанром творчості (наприклад, весільні фотографи) або ж з яких-небудь подібним поглядам на фотомистецтво.

В цілому, Інтернет-спільнота являє собою своєрідну соціальну мережу, яка об'єднує професійних фотографів, а також всіх тих, кому цікава фотографія.

Основу такої спільноти складають зазвичай блоги фотографів, які представляють їх роботи, історії з життя та поради новачкам, тематичні групи та авторські фотогалереї.

Ставши учасником Інтернет-спільноти, Ви отримуєте можливість додавати свої роботи в фото галереї, отримувати коментарі та критику, публікувати статті та замітки, що стосуються всіх аспектів фотографії. На форумі Ви можете поспілкуватися зі своїми колегами по ремеслу на будь-яку цікаву для Вас тему, обговорити сучасні тенденції, події та технічні новинки в світі фотографії.

Товариства фотографів дозволяють через спеціалізовані сайти на просторах Інтернету ділитися досвідом і знаннями один з одним, тим самим сприяючи підвищенню свого рівня майстерності. Ви можете подивитися, як працює в тому чи іншому жанрі професійні майстри. Це своєрідний майстер-клас без особистого спілкування. Знайомство з роботами професійним фотографів та обмін досвідом допомагає початківцям любителям фотографії виробляти нові ідеї, формувати власний художній смак і постійно вдосконалюватися.

Інтернет-спільноти також дозволяють заявити про себе, даючи кожному учаснику можливість опублікувати своє портфоліо. Учасники спільноти фотографів також нерідко організовують зустрічі в реальному світі з метою просто добре провести час, повеселитися і розширити своє коло спілкування.

Створюючи співтовариство фотографів на розроблюваному сайті, я хотів об'єднати всі ті переваги, що дають фотографам Інтернет - співтовариства. Я передбачив все. Або майже все.

# 2 ОГЛЯД ІНСТРУМЕНТІВ ТА ЗАСОБІВ WEB РОЗРОБКИ

## 2.1 Базові відомості про Internet та WEB

Неможливо уявити собі світ без спілкування людей між собою. Люди спілкуються між собою, як при особистих зустрічах так і з допомогою різних технічних і технологічних засобів – з допомогою пошти, телефону, телеграфу, радіо телебачення тощо. Відповідно поява комп’ютерів не могла не викликати появи засобів зв’язку їх між собою, оскільки комп’ютери були створені для обробки інформації, а інформацію як відомо потрібно якимось чином одержувати і після обробки передавати. Таким чином комп’ютери спочатку об’єднували в локальні мережі, потім в глобальні мережі, а потім глобальні мережі поєднались між собою. Поєднання глобальних комп'ютерних мереж, що поєднують комп'ютери в усьому світі в єдиному інформаційному просторі, носить назву - Інтернет. Інтернет багатогранний і не можна чітко визначити, що це таке. Він не вирішив проблему збереження й упорядкування інформації, але вирішив проблему її передачі, він дав можливість одержати будь-яку інформацію де завгодно, коли завгодно.

Інтернет можна розглядати в широкому та вузькому розумінні. Найбільш вузький погляд – Інтернет, це мережа мереж, всесвітня комп'ютерна мережа в вузькому розумінні, але більш ширший погляд – Інтернет це кіберпростір, що народжує кіберкультуру зі своїми думками, своєю мовою, своєю етикою.

Отже Інтернет це мережа мереж, яка містить велике число серверів, таких як WWW [4], електронна пошта, мережні новини USENET, пошукова система GOOGLE, сервіси: Gopher, FTP, IRC, MUD, MOO, ICQ та інші. Найбільший з цих серверів це WWW – World Wide Web (Всесвітня павутина), який коротко ще називають Web. Web настільки популярний серед людей, що деякі люди, думають, що Web - це і є Internet. Але це невірно, тому що Web - тільки одна з багатьох служб, що використовуються в Internet.

Web була винайдена в 1980 році в CERN (європейської лабораторії фізики елементарних часток). Співробітник цієї лабораторії Тім Бернерс Лі створив програму за назвою Enquire Within Upon Everything (Enquire), щоб відстежити зв'язки між документами, включивши в них посилання один на одного. І, як сказав Тім у 1994 році, даючи інтерв'ю кореспондентам видання Internet World, він хотів знайти більш зручний і логічний спосіб представлення інформації. У 1989 році Бернерс Лі запропонував глобальне узагальнення цієї ідеї - зв'язати гіпертекстові документи в усьому світі. Інформація повинна міститися на серверах, а для її перегляду потрібно використовувати особливі програми-браузери. З тих пір почався бурхливий розвиток Web.

Для роботи в Web необхідна спеціальна програма-клієнт, що називається Web-браузером. Одним з перших браузерів була програма Mosaic, розроблена засновником фірми Netscape Марком Ендрісеном (Магc Andreesen). Перша версія Mosaic була випущена центром NCSA (National Center for Supercomputing Applications - Національний центр суперкомп'ютерних додатків) у 1993 році. У 1994 році Ендрісен залишив NCSA, щоб заснувати фірму Mosaic Communications, на якій почали розробляти браузер Netscape Navigator, який в даний час є найбільшим конкурентом браузера Internet Explorer фірми Microsoft.

Теоретично гіпертекст - це усього лише зручний спосіб представлення інформації. Але на практиці гіпертекст - це можливість зробити посилання на інші документи за допомогою слів, фраз, малюнків. Ім’я кожного з цих місць можна зв'язати з іншим документом, у якому міститься більш докладна інформація. Коли користувач вибирає посилання в першому документі, браузер відкриває другий документ із більш докладними даними.

У гіпертексту є дві важливі особливості:

* інформація ніяк не впорядковується - документи просто зв'язуються один з одним за допомогою посилань. Хоча головною метою багатьох методів є саме впорядкування інформації тим або іншим способом (наприклад, у виді ієрархії), у гіпертексті основна увага приділяється створенню інформаційних зв'язків. Таким чином, гіпертекст - це спроба створення моделі, що описує спосіб представлення інформації в мозку людини.
* Інформаційні зв'язки можуть існувати між самими різними документами. Створюючи впорядкований список або схему, ви поміщаєте на кожне місце в списку або ієрархії (тобто в структурі) тільки один елемент. А в гіпертексті кожен інформаційний фрагмент (або елемент) може знаходитися в багатьох, причому зовсім різних, місцях структури.

Термін гіпермедіа (hypermedia) використовується для опису того, що ви знаходите в Web. Гіпермедіа - це природне узагальнення поняття гіпертексту, що відноситься до документів, у яких розміщується не тільки текст, але і мультимедіа, тобто зображення, відеозаписи і звук. Ці елементи також можна зв'язувати з іншими документами гіпермедіа. Наприклад, на Web-сторінці можна зв'язати зображення з документом таким чином, що якщо користувач клацне на зображенні, браузер відкриє відповідний документ.

Документи HTML.HTML [5] (Hypertext Markup Language - мова гіпертекстової розмітки) служить для опису Web-сторінки, що зберігається у виді звичайного текстового файлу з розширенням \*.htm або \*.html. Головна мета HTML - описати формат вмісту Web-сторінки, він описується з допомогою HTML. Дескриптори визначають способи форматування тексту, служать розпізнавальними знаками зображень або таблиць, дозволяють зв'язувати слова або фрази з іншими документами в Internet.

Якщо дати коротке означення Web-сторінки, то це комбінація тексту і дескрипторів HTML, що описують способи форматування цього тексту.

Web-сторінки зберігаються у файлах з розширенням \* .htm або \*.html. Але якщо на Web-сервері використовується технологія ASP (Active Server Page - активні сторінки сервера) чи технологія PHP(Personal Home Page), то замість HTML-сторінок на сервері зберігаються ASP-файли чи PHP файли. ASP-файл чи PHP-файл, містить сценарій, по якому Web-сервер динамічно створює HTML-сторінку в момент звертання до нього браузера. Технології ASP і PHP дозволяють представляти інформацію в більш динамічному виді, даючи можливість звертатися до баз даних або до іншої інформації, що неможливо представити у виді звичайних HTML-документів.

Технологія ASP створена фірмою Microsoft на основі мови програмування Visual Basic і відповідно вбудована в офісні додатки Microsoft Office. Тому для людей, які не знайомі з Web-програмуванням, але знайомі з принципами роботи додатків Microsoft Office кращим рішенням при створенні Web сторінок використовувати HTML в поєднанні з технологією ASP.

HTML - це стандарт, прийнятий Консорціумом Всесвітньої інформаційної мережі (W3C - World Wide Web Consortium). Остання рекомендація HTML (W3C не використовує термін "стандарт") має номер 4.1. (Багато фірм-виготовлювачів, наприклад Microsoft і Netscape, визначають власні розширення для HTML-файлів.) Асоціація W3C несе також відповідальність за визначення таких стандартів, як HTTP (Hypertext Transport Protocol - протокол передачі гіпертексту) і PICS (Platform for Internet Content Selection - платформа вибору інформації в Internet). Діяльністю W3C керує Тім Бернерс Лі, винахідник гіпертексту. Більш докладну інформацію про цю організацію можна знайти на її Web-сервері за адресую http//www.w3.org.

Термін Web-сторінка позначає документи в Web, у яких для опису атрибутів форматування використовується мова HTML. HTML-документ - це ще одна назва Web-сторінки, причому більш точне, оскільки HTML використовується не тільки в Web.

Однією з головних понять HTML є також URL. Для звертання до ресурсу використовується його адреса, або URL. URL - це стандарт, прийнятий для визначення місцезнаходження будь-якого ресурсу в Internet, будь це документ або служба.

Термін "Web-сторінка" в однаковій мірі підходить для всіх HTML-документів у Internet. A термін "початкова сторінка" має два різних визначення. По-перше, початкова сторінка - це сторінка самого верхнього рівня на великому Web-сервері. Саме ця Web-сторінка відкривається у випадку, якщо користувач уводить URL без вказівки шляху або імені файлу. У подібному випадку Web-cepвep відкриває файл з ім'ям lndex.htm або default.htm і відображає початкову сторінку. По-друге, термін "початкова сторінка" використовується для опису персональних Web-сторінок, що поміщають у Internet окремі користувачі.

HTTP розшифровується як Hyper Text Transport Protocol (протокол передачі гіпертексту) і використовується в Web з 1990 року. Він служить для визначення способів формування запитів клієнтів і способів відповіді Web-серверів на ці запити. Коротше кажучи, HTTP описує способи передачі Web-сторінок.

HTTP - це простий текстовий протокол, тобто ви можете прочитати і зрозуміти кожну команду, що посилається на сервер (спробуйте зв'язатися з портом 80 Web-сервера за допомогою telnet). Броузер посилає на Web-сервер запит, а той, у свою чергу, видає відповідь. Першим рядком запиту є команда, звичайно GET (Одержати) або POST (Послати), що містить URL ресурсу (файлу, сценарію, програми і т.д.) і версію HTTP, використовувану браузером. За командою випливають заголовки, у яких утримується додаткова інформація, необхідна Web-серверові. Усі заголовки мають однаковий формат: ім'я: значення.

Відповідь сервера починається з указівки версії HTTP і коду повернення. Код повернення час від часу з'являється на екрані вашого браузера, коли сервер повертає помилку.

Протокол HTTP не підтримує функції збереження інформації про користувача або про його браузері. Для постійного збереження інформації про користувача використовуються розширення, прийняті фірмою Netscape (або службові індексні файли) cookies. Зараз ця технологія визнана в усім світі. Файли cookies зберігаються на комп'ютері користувача. Сервер визначає cookie у заголовку відповіді Set-cookie. І з кожним наступним запитом браузер повертає cookie на сервер у заголовку запиту Cookie. Таким чином, сервер зберігає інформацію на комп'ютері користувача, а потім прочитує неї при одержанні чергового запиту.

При обробці запиту Web-сервер аналізує URL, у якому може утримуватися звертання до файлові або програмі. Іноді розробники Web-сторінок доповнюють їх службовими програмами, що виконують, наприклад, перевірку даних, введених користувачем, або обробку замовлення в режимі online. HTTP не визначає способи виконання сервером подібних операцій. Якщо ж URL указує на файл, сервер просто повертає його вміст браузеру.

Якщо URL указує на програму, то сервер запускає її з параметрами, переданими браузером. У деяких випадках на сервері є спеціальні каталоги, такі як CGI-BIN. Це віртуальний каталог із програмами. Запис у URL типу host/cgi-bin/prog змушує Web-сервер викликати програму prog з каталогу cgi-bin. В інших випадках у URL указується конкретний шлях до програми, наприклад, host/prog. dll, що є ISAPI-фільтром, що використовується разом з Internet Information Server.

Програма-клієнт, що використовується для звертання до Web-сервера, називається Web-броузером. Два самих популярних браузера на сьогоднішній день - це Internet Explorer і Netscape.

По своїй суті Web-браузери - неймовірно складні програми. Вони підтримують безліч стандартів, що визначають усе, що тільки можна: починаючи від способів передачі Web-сторінок за допомогою HTTP і відображення HTML-документа і закінчуючи захистом транзакцій за допомогою SSL. Але зовні Web-браузери здаються надзвичайно простими, тому що ними легко користуватися. Розглянемо можливості, якими володіє більшість Web-браузерів.

Ідентифікація Web-сторінок за допомогою URL. В усіх Web-браузерах реалізовані однакові методи відкривання Web-сторінок. Ви можете клацнути на посиланні, набрати URL-адресу Web-сторінки в адресному полі, скористатися меню, вибрати сервер у списку Обране (Favorites) і т.д.

Можливості Web-браузерів не обмежуються одним тільки відображенням HTML-документів. За допомогою Web-браузера можна, наприклад, переглянути Gopher-сервер або завантажити файли з FTP-сервера.

Динамічний HTML користується в даний час величезною популярністю, оскільки дозволяє розроблювачеві створити дійсно інтерактивну Web-сторінку, а не статичний документ. Подібна можливість досягається за рахунок того, що для кожного елемента Web-сторінки можна написати сценарій. Наприклад, можна написати сценарій зміни кольору слова при приміщенні на нього покажчика миші, чи написати сценарій імітації феєрверку при натисненні кнопки миші на деяких елементах Web-сторінки.

## 2.2 Скриптова мова програмування PHP

PHP (англ. PHP:Hypertext Preprocessor - PHP: гіпертекстовий препроцесор), попередньо: Personal Home Page Tools - скриптова мова програмування, була створена для генерації HTML-сторінок на стороні веб-серверу. PHP є однією з найпоширеніших мов, що використовуються у сфері веб-розробок (разом із Java, .NET, Perl, Python, Ruby). PHP підтримується переважною більшістю хостинг-провайдерів. Проект за яким був створений PHP - проект з відкритими програмними кодами.

PHP інтерпретується веб-сервером в HTML-код, який передається на сторону клієнта. На відміну від таких скриптових мов програмування, як JavaScript, користувач не має доступу до PHP-коду, що є перевагою з точки зору безпеки але значно погіршує інтерактивність сторінок. Але ніщо не забороняє використовувати РНР для генерування і JavaScript-кодів які виконаються вже на стороні клієнта.

PHP - мова, яка може бути вбудована безпосередньо в html-код сторінок, які, в свою чергу коректно будуть оброблені PHP -інтерпретатором. Механізм РНР просто починає виконувати код після першої екрануючої послідовності (<?) і продовжує виконання до того моменту, коли він зустріне парну екрануючу послідовність (?>).

Велика різноманітність функцій PHP дають можливість уникнути написання багаторядкових призначених для користувача функцій на C або Pascal.

Наявність інтерфейсів до багатьох баз даних:

* в PHP вбудовані бібліотеки для роботи з MySQL, PostgreSQL, mSQL, Oracle, dbm, Hyperware, Informix, InterBase, Sybase.
* через стандарт відкритого інтерфейсу зв'язку з базами даних (Open Database Connectivity Standard - ODBC) можна підключатися до всіх баз даних, до яких існує драйвер.

Мова РНР здаватиметься знайомою програмістам, що працюють в різних областях. Багато конструкцій мови запозичені з С, Perl. Код РНР дуже схожий на той, який зустрічається в типових програмах на С або Pascal. Це помітно знижує початкові зусилля при вивченні РНР. PHP - мова, що поєднує переваги Perl і С і спеціально спрямована на роботу в Інтернеті, мова з універсальним і зрозумілим синтаксисом. І хоча PHP є досить молодою мовою, вона здобула таку популярність серед web-програмістів, що на даний момент є мало не найпопулярнішою мовою для створення web-додатків (скриптів).

Стратегія Open Source, і розповсюдження початкових текстів програм в масах, безсумнівно справили благотворний вплив на багато проектів, в першу чергу - Linux хоч і успіх проекту Apache сильно підкріпив позиції прихильників Open Source. Сказане відноситься і до історії створення РНР, оскільки підтримка користувачів зі всього світу виявилася дуже важливим чинником в розвитку проекту РНР.

Ухвалення стратегії Open Source і безкоштовне розповсюдження початкових текстів РНР надало неоціниму послугу користувачам. Додатково, користувачі РНР по всьому світу є свого роду колективною службою підтримки, і в популярних електронних конференціях можна знайти відповіді навіть на найскладніші питання.

Ефективність є дуже важливим чинником при програмуванні для середовищ розрахованих на багато користувачів, до яких належить і web. Важливою перевагою PHP є те, що ця мова належить до інтерпретованих. Це дозволяє обробляти сценарії з достатньо високою швидкістю. За деякими оцінками, більшість PHP-сценаріїв (особливо не дуже великих розмірів) обробляються швидше за аналогічні їм програми, написані на Perl. Проте, щоб не робили розробники PHP, виконувані файли, отримані за допомогою компіляції, працюватимуть значно швидше - в десятки, а іноді і в сотні разів. Але продуктивність PHP цілком достатня для створення цілком серйозних web-додатків.

З точки зору системи типізації, PHP є мовою програмування з динамічною типізацією. Немає необхідності явного визначення типу змінних, хоча така можливість існує. В разі звернення до змінної, інтерпретатор PHP трактує її тип відповідно до контексту. За необхідності можливе приведення змінної до певного типу за допомогою відповідних конструкцій мови. Це може знадобитись, якщо зважити, що значення змінної можуть трактуватись по-різному в залежності від її типу. Також можливе визначення типу відповідної змінної на певному етапі виконання сценарію. Імена змінних чутливі до регістру символів.

Що стосується функцій в PHP, то замість прийнятого в багатьох мовах принципу перевантаження функцій, що дозволяє змінити хід виконання певної функції в залежності від типу та кількості переданих параметрів, використовується метод динамічних аргументів. Це дає змогу не визначати кількість параметрів для функцій при їх оголошенні, а працювати із тими аргументами, які були отримані на момент виклику функції. У тілі функції можливо отримати кількість переданих їй аргументів і проводити відповідні маніпуляції. При оголошенні функції звичайним чином, можливе задання значень аргументів за замовчуванням. Функції можуть повертати лише одне значення, проте це обмеження можна оминути, використавши не лише масиви, а й посилання. Передача аргументів за посиланням неможлива під час виконання та оголошення функції.

Після виконання сценаріїв, простір пам'яті, займаної ними очищується збирачем сміття. Проте, за необхідності можливе виконання очищення пам'яті від надлишкових сегментів даних під час виконання скриптів. Використання функцій очищення пам'яті є невиправданим, хоча така можливість існує.

Для побудови програмних комплексів можна використовувати модульний підхід, виконуючи розділення різнорідного коду. При потребі, можливе виконання під'єднання необхідних модулів, причому операція виконання може бути і умовною. Під'єднані до скрипта файли можуть повертати значення.

Історія PHP починається з 1995 року, коли Расмус Лердорф (Rasmus Lerdorf) створив простий додаток мовою Perl, що аналізував відвідування користувачами його резюме на веб-сайті. Потім, коли цим додатком вже користувалися декілька чоловік, а число охочих одержати його постійно збільшувалося, Лердорф назвав своє творіння Personal Home Page Tools версія 1 і виставив для вільного завантаження. З цієї миті почався небувалий зліт популярності PHP.

Як це завжди буває, терміново було потрібно доопрацювання і нові доповнення. Для їхньої реалізації Расмус створює нову версію пакету, тепер уже написану С. Отриманий таким чином інструмент набуває робочої назви PHP/FI (Personal Home Page / Forms Interpreter - Персональна Домашня сторінка / Інтерпретатор Форм), надалі він також буде відомий під назвою PHP 2. Ця версія вже більшою мірою схожа на сьогоднішній PHP. Вона мала синтаксис і спосіб іменування змінних в стилі мови Perl, можливість вбудовування PHP операторів в html-код сторінки, автоматичну інтерпретацію форм, інтеграцію з базами даних. При цьому все працювало досить швидко, оскільки PHP прокомпелювалося до веб-серверу Apache. До 1997 року PHP використовувався вже на 50,000 доменах (не більше 1% всіх веб-серверів).

У тому ж 1997 році до проекту PHP підключилися Зеєв Сураський (Zeev Suraski) і Енді Гутманс (Andi Gutmans). Ці студенти одного з ізраїльських університетів намагалися використовувати PHP/FI для одного з комерційних університетських проектів. При цьому їм довелося зіткнутися з багатьма труднощами і обмеженнями цієї технології. Вивчаючи початковий код PHP 2, Зеєв і Енді дійшли висновку про необхідність доопрацювання, а точніше істотної переробки PHP, особливо в плані синтаксису мови. Протягом декількох місяців вони блискуче справилися з цим завданням.

Закінчивши роботу Зеєв і Енді домовились з Расмусом про співпрацю в галузі розвитку та вдосконалення мови. З цієї миті з'являється PHP Group - група однодумців, що працюють над розвитком технології PHP. Одержаний продукт з’явився на світ у 1998 році під назвою PHP 3.

При цьому головною особливість PHP 3 була можливість розширення ядра, що привернуло до роботи над PHP безліч сторонніх розробників, що створюють спеціалізовані модулі. Їх наявність дала PHP можливість працювати з величезним кількість баз даних, протоколів, підтримувати велике число API. До кінця 1998 кількість користувачів PHP перевалила за 100000, а PHP був вже встановлений на не менше ніж 10% серверах Інтернету. У той ж час значному поширенню даної мови сприяли публікації в електронній пресі та вихід книжок по вивченню PHP.

Відразу ж після виходу PHP 3, Енді Гутманс і Зеєв Сураський почали переробку ядра PHP. В першу чергу належало вирішити проблему підвищення продуктивності. Новий продукт, названий Zend Engine (від імен творців: Zeev і Andi), успішно справлявся з поставленим завданням і був реалізований в 1999 році. Основними реалізованими ідеями є можливість компіляції сценарію у виконуваний модуль, за рахунок чого продуктивність можна було підняти на порядок.

PHP 4, що працює на цьому ядрі, вийшов в 2000 році. На додаток до поліпшення продуктивності, PHP 4 мав нові можливості по підтримці сесій, буферизацію виводу, безпечні способи обробки інформації, що вводиться користувачем, і нові мовні конструкції. З виходом 4 версії PHP став використовуватися вже на більш ніж 20% доменів Інтернету.

За час з 2000 по 2004 рік продовжувалися активні роботи по поліпшенню 4 версії, але майже відразу PHP Group приступила до продумування можливостей нової версії. В першу чергу було вирішено підсилити об'єктні можливості мови, що дозволяло використовувати його для реалізації масштабних проектів. Роботи із створення версії 5 велися тривалий час, в них брало участь рекордна кількість фахівців, зокрема Стерлінг Хьюз (Sterling Hughes ) і Маркус Бергера (Marcus Boerger ).

У липні 2004 року виходить офіційний реліз PHP 5. В першу чергу, як і планувалося, було перероблено весь механізм роботи з об'єктами. І якщо в попередніх версіях об'єктно-орієнтоване програмування на PHP було можливе в мінімальному ступені, а тому і використовувалося на практиці не часто, то PHP 5 володіє прекрасним потенціалом реалізації об'єктного програмування. Окрім цього, PHP збагатився рядом цінних розширень для роботи з XML, різними джерелами даних, генерації графіки і інше.

Серед інших украй корисних доповнень в PHP 5 слід зазначити нову схему обробки виключень. Конструкція try/catch/throw дозволяє весь код обробки помилок локалізувати в одному місці сценарію.

Всі основні бібліотеки для роботи з XML, запозичені в PHP 4, були піддані серйозній переробці. Такі популярні розширення, як SAX, DOM і XSLT, тепер використовують інструмент libxml2, що робить їх ще ефективнішими.

У PHP 5 також включені два нові модулі для роботи з протоколами - SimpleXML і SOAP. SimpleXML дозволяє значно спростити роботу з XML-даними, представляючи вміст XML-документа у вигляді PHP-об'єкту. Розширення SOAP дозволяє будувати на PHP сценарії, що обмінюються інформацією з іншими додатками за допомогою XML-повідомлень поверх існуючих веб-протоколів, наприклад HTTP. Модуль для роботи з SOAP для PHP 5 надає розробникам засіб для достатньо швидкого створення ефективних SOAP-клієнтів і SOAP-серверів.

Новий модуль PHP 5 MySQLi (MySQL Improved) призначений для роботи з MySQL-сервером версій 4.1.2 і вище, реалізовуючи не тільки процедурний, але і об'єктно-орієнтований інтерфейс до MySQL. Додаткові можливості цього модуля включають - SSL, контроль транзакцій, підтримка реплікації і ін. Очевидно, що, на цьому історія PHP не закінчується. Слід очікувати наступних версій мови із розширеними можливостями.

Всі сценарії оформляються у вигляді блоків коду. Ці блоки можуть бути поміщені в HTML-код, але відділені від нього відповідними обмежувачами. Код PHP в HTML повинен знаходитись між початковим тегом <?php та кінцевим ?> (або між <script language="php"> та </script>) Бажаним варіантом виділення PHP коду є варіант <?php ?>, оскільки саме такі початковий та кінцевий теги дозволять використовувати PHP код в документах, які відповідають правилам XML. Також можна користуватися скороченим записом: <? ?> (в php.ini змінна short\_open\_tag повинна мати значення On) і записом в стилі ASP: <% %> (в php.ini змінна asp\_tags повинна мати значення On). Проте стиль ASP не рекомендується і очікується, що він буде відсутній у PHP6.

Спочатку визначимо сферу застосування PHP. Як і будь-яка інша скриптова мова, PHP використовується на сайтах, зміст (контент) яких є динамічним, тобто залежить від того, хто, коли і яким чином його переглядає, а також має можливість доповнювати та редагувати контент (портали, форуми, гостьові книги). Також PHP може стати у пригоді для організації статичних сайтів великого об’єму, що містять велику кількість документів, які потрібно структурувати та привести до подібного вигляду (електронні бібліотеки, сайти документації).

Джерела інформації. Основні джерела – це посібник з PHP [11] (www.php.net) та MySQL (www.mysql.com). При підготовці статті також використовувалися матеріали з www.webreview.com та www.devshed.com. Деякі пункти є перекладами з цих джерел.

Результатом роботи PHP є звичайні сторінки мовою HTML, тобто людина, що переглядає веб-сайт, може не здогадуватись про те, що веб-сторінка не є статичною, а була згенерована. Код PHP (тобто деякий текст програми, що виконується сервером і результатом виконання якої є веб-сторінка) може бути вбудваний всередину статичного коду HTML.

Розробникам Web-аплікацій немає необхідності говорити, що web-сторінки – це не тільки текст та картинки. Гідний уваги сайт повинен підтримувати деякий рівень інтерактивності з користувачем: пошук інформації, продаж продуктів, конференції, тощо. Традиційно все це було реалізовано CGI-скриптами, написаними на Perl. Але CGI- скрипти дуже погано масштабується. Кожний новий виклик CGI, вимагає від ядра породження нового процесу, а це забирає процесорний час та витрачає оперативну пам'ять. PHP пропонує інший варіант – він працює як частина Web-сервера, й цим схожий на ASP від Microsoft.

#### 2.2.1 PHP Фреймворк Kohana

Фреймворк - це каркас сайту, який дозволяє прискорити процес розробки, тобто не писала якісь повторювані ділянки з проекту в проект, а зробити основний наголос саме на логіці. Якщо ви досить давно займаєтеся веб-розробкою, то у вас точно є якісь готові класи або функції, які ви використовуєте у своїх проектах. Такий ось набір готових до використання бібліотек - це по суті і є фреймворк. Але будь-який розробник, із зростанням досвіду, приходить до висновку, що його старий код (фреймворк) нікуди не годиться і його треба переробляти. Так чому не використовувати відразу вже готовий фреймворк, такий як Kohana, який до того ж написаний дуже грамотно.

Kohana - це веб-фреймворк з відкритим кодом, заснований на PHP5 і використовує концепцію HMVC (Hierarchical Model View Controller - Іерахіческіе Модель-Вид-Контролер). Його основні цілі - бути безпечним, легким і простим у використанні, і це дійсно так.

Kohana був створений як гілка PHP фреймворку CodeIgniter. Спочатку він був відомий під назвою Blue Flame. Основною причиною Форком («розщеплення» на два окремі проекти) був перехід до більш відкритою для громадськості моделі розробки, через те, що багато користувачів були незадоволені швидкістю розробки та виправлення помилок в CodeIgniter. Rick Ellis – розробник і власник CodeIgniter - був щасливий бачити форк свого проекту, але допомагати відмовився. Він підштовхнув новий проект до створення власної документації і порадив перейменувати проект. У липні 2007року Blue Flame був перейменований в Kohana для того, щоб уникнути проблем з авторськими правами в майбутньому.

Назва Kohana було вибрано, коли розробники почали переглядати словники корінних американців, щоб вибрати слово що не порушує авторські права. Kohana мовою Сіу означає «швидкий». Також на японській мові має значення «маленька квітка», а на українському «кохана» і ім'я відомої косатки (Kohana) - все це не має ніякого відношення до назви.

До переваг даного фреймворк можна віднести наступне:

* Висока швидкість роботи;
* Безпека;
* Використання можливостей PHP5;
* Велика кількість вбудованих інструментів;
* Простота розуміння;
* Використання концепції HMVC;
* Повна сумісність з UTF-8.

Для того, щоб уміти працювати з фреймворком, потрібно розуміти його структуру ну і звичайно знати його класи і методи. Абсолютно необхідно, щоб ви знали PHP на рівні ООП (хоча б початковому), а також розуміли що таке MVC.

#### 2.2.2 ORM

ORM або Object-relational mapping (рос. Об'єктно-реляційне відображення) - це технологія програмування, яка дозволяє перетворювати несумісні типи моделей в ООП, зокрема, між сховищем даних та об'єктами програмування. ORM використовується для спрощення процесу збереження об'єктів в реляційну базу даних та їх вилучення, при цьому ORM сама піклується про перетворення даних між двома несумісними станами. Більшість ORM-інструментів значною мірою покладаються на метадані бази даних і об'єктів, так що об'єктам нічого не потрібно знати про структуру бази даних, а базі даних - нічого про те, як дані організовані у додатку. ORM забезпечує повне розділення завдань в добре спроектованих додатках, при якому і база даних, і додаток можуть працювати з даними кожен у своїй вихідній формі.



Рисунок 2.1 Розподіл завдань з допомогою ORM

Парадигма «невідповідності». Говорячи конкретніше, використання ORM вирішує проблему так званої парадигми «невідповідності», яка свідчить про те, що об'єктні та реляційні моделі не дуже добре працюють разом. Реляційні бази представляють дані в табличному форматі, в той час як об'єктно-орієнтовані мови представляють їх як зв'язаний граф об'єктів. Основні проблеми та невідповідності виникають під час збереження цього графа об'єктів в реляційну базу або його завантаження:

* реляційна модель може бути набагато детальніше, ніж об'єктна, тобто для зберігання одного об'єкта в реляційній базі даних використовується декілька таблиць;
* реляційні СУБД не мають нічого схожого на спадкування - природну парадигму об'єктно-орієнтованих мов програмування;
* в СУБД визначений тільки один параметр для порівняння записів - первинний ключ. У той час як ООП надає як перевірку ідентичності об'єктів (a == b), так і їх рівності (a.equals (b));
* для зв'язку об'єктів СУБД використовує поняття зовнішніх ключів, в об'єктно-орієнтованих мовах зв'язок між об'єктами може бути тільки односпрямованої. Якщо ж потрібно організувати двонаправлені відносини, то доведеться визначити дві односпрямовані асоціації. Крім того, немає можливості визначити кратність відносини, дивлячись на модель предметної області;
* принцип доступу до даних в ООП кардинально відрізняється від доступу до даних в БД. Для доступу до даних в ООП використовуються послідовні переходи від батьківського об'єкта до властивостей дочірніх елементів і ініціалізації об'єктів за необхідності. Такий підхід вважається не ефективним способом отримання даних з реляційних баз даних. Як правило, кількість запитів до БД має бути зведено до мінімуму, необхідні суті повинні по можливості завантажуватися відразу з використанням JOIN-ів.

Ключовою особливістю ORM є відображення, яке використовується для прив'язки об'єкта до його даними в БД. ORM як би створює «віртуальну» схему бази даних у пам'яті і дозволяє маніпулювати даними вже на рівні об'єктів. Відображення показує як об'єкт і його властивості пов'язані з однією або декількома таблицями і їх полями в базі даних. ORM використовує інформацію цього відображення для управління процесом перетворення даних між базою і формами об'єктів, а також для створення SQL-запитів для вставки, оновлення та видалення даних у відповідь на зміни, які додаток вносить в ці об'єкти.

Використання ORM в проекті позбавляє розробника від необхідності роботи з SQL і написання великої кількості коду, часто одноманітного і схильного помилок. Весь генерований ORM код імовірно добре перевірений і оптимізований, тому не потрібно в цілому замислюється про його тестуванні. Це безсумнівно є плюсом, але в теж час не варто забувати і про мінуси. Основний з них - це втрата продуктивності. Це відбувається тому, що більшість ORM призначені для обробки широкого спектру сценаріїв використання даних, набагато більшого, ніж будь-яке окремий додаток коли-небудь зможе використовувати. Питання про доцільність використання ORM за великим рахунком зачіпається тільки у великих проектах, які стикаються з високим навантаженням, тут доводиться вибирати що більш пріоритетно - зручність чи продуктивність? Звичайно, робота з БД за допомогою грамотно написаного SQL-коду буде набагато ефективніше, але не варто забувати і про такий параметр, як час - те, що з легкістю пишеться з використанням ORM за тиждень, можна реалізовувати жоден місяць власними зусиллями. Крім того, більшість сучасних ORM дозволяють програмісту при необхідності самому задавати код SQL-запитів. Без сумнівів, для невеликих проектів використання ORM буде куди більш виправдане, ніж розробка власних бібліотек для роботи з БД.

## 2.3 Каскадні таблиці стилів CSS

Cascading Style Sheets (каскадні таблиці стилів) - технологія опису зовнішнього вигляду документа, написаного мовою розмітки. CSS використовується переважно для оформлення HTML- і XHTML-документів, але іноді і для інших XML-структурованих документів (наприклад, в браузерs Mozilla для оформлення елементів графічного інтерфейсу, XUL).

Таблиці стилів в тому чи іншому вигляді існували з зародження SGML в 1970-тих. Каскадні таблиці стилів розроблялися для спрощення процедури додавання інформації про стилі для веб-сторінок.

З розвитком HTML з’явилася можливість використовувати різні механізми для додавання стилів до елементів сторінки. Еволюція HTML дала веб-дизайнерам більше можливостей для створення вигляду сайту, але HTML-код ставав складнішим для написання та зміни. Через різницю у відображення сторінки в різних браузерах збереження стилю сторінки було складним, користувачі мали менше контролю над відображенням контенту.

На розгляд W3C було запропоновано дев’ять різних варіантів таблиць стилів. Після обговорення в спеціальному списку розсилки було обрано два, вони створили основу для того, що стало CSS: англ. Cascading HTML Style Sheets (CHSS) та англ. Stream-based Style Sheet Proposal (SSP). Спочатку в жовтні 1994, Håkon Wium Lie (зараз генеральний технічний директор Opera Software) запропонував Cascading HTML Style Sheets (CHSS), що дещо подібний до сучасного CSS. Bert Bos працював над браузером Argo, що використовував власний варіант таблиць стилів, Stream-based Style Sheet Proposal (SSP). Lie і Bos почали співпрацювати для вироблення стандарту CSS (літера 'H' була виключена з назви, оскільки таблиці стилів могли застосовуватися до інших мов розмітки, не лише HTML).

На відміну від наявних тоді таблиць стилів, таких як DSSSL та FOSI, CSS дозволяв застосування різних таблиць стилів до документу (сторінки). Таблиці стилів могли наслідувати правила з інших (тобто створювати каскади), що дозволило контролювати використання стилів як дизайнером сайту так і користувачем (наприклад в браузері Opera).

Пропозиції до стандарту CSS обговорювалися на конференціях в 1994 та 1995 роках.

В 1994 році було створено World Wide Web Consortium W3C, серед інших питань W3C займався також і CSS. Робочу групу очолив Steven Pemberton, як провідні технічні спеціалісти до неї входили Håkon Wium Lie та Bert Bos. В грудні 1996 було опубліковано CSS рівня 1.

CSS використовується розробниками веб-сторінок для завдання кольорів, шрифтів, розташування і інших аспектів представлення документа. Основною метою розробки CSS було розділення вмісту (написаного на HTML або іншій мові розмітки) і представлення документа (написаного на CSS). Це розділення може збільшити доступність документа, надати велику гнучкість і можливість управління його уявленням, а також зменшити складність і повторюваність в структурному вмісті. Крім того, CSS дозволяє представляти один і той же документ в різних стилях або методах висновку, таких як екранне уявлення, друк, читання голосом (спеціальним голосовим браузером або програмою читання з екрану), або при висновку пристроями, що використовують Шрифт Брайля.

Стандарт CSS визначає пріоритети, у порядку яких застосовуються правила стилів, якщо для якогось елементу підходять деякі правила одночасно. Це називається "каскадом", в якому для правил розраховуються пріоритети або "ваги", що робить результати передбаченими.

Таблиця стилів складається з набору правил. Кожне правило, у свою чергу, складається з одного або декількох селекторів, розділених комами і блоку визначень.

До появи CSS оформлення веб-сторінок здійснювалося безпосередньо усередині вмісту документа. Проте з появою CSS стало можливим принципове розділення змісту і представлення документа. За рахунок цього нововведення стало можливим легке застосування єдиного стилю оформлення для маси схожих документів, а також швидка зміна цього оформлення.

Переваги CSS розмітки:

* + Декілька дизайнів сторінки для різних пристроїв перегляду. Наприклад, на екрані дизайн буде розрахований на велику ширину, під час друку меню не виводитиметься, а на КПК і стільниковому телефоні меню буде слід за вмістом.
  + Зменшення часу завантаження сторінок сайту за рахунок перенесення правил представлення даних в окремий CSS-файл. В цьому випадку браузер завантажує тільки структуру документа і дані, що зберігаються на сторінці, а представлення цих даних завантажується браузером тільки один раз і кешується.
  + Простота подальшої зміни дизайну. Не потрібно правити кожну сторінку, а лише змінити CSS-файл.
  + Додаткові можливості оформлення.

## 2.4 Twitter Bootstrap

Twitter Bootstrap - вільний набір інструментів для створення сайтів і веб-додатків. Включає в себе HTML і CSS шаблони оформлення для типографіки, веб-форм, кнопок, міток, блоків навігації та інших компонентів веб-інтерфейсів, включаючи JavaScript розширення.

Bootstrap використовує найсучасніші напрацювання в галузі CSS і HTML, тому необхідно бути уважним при підтримці старих браузерів.

Основні переваги Twitter Bootstrap є наступні:

* Економія часу - Bootstrap дозволяє заощадити час і зусилля, використовуючи шаблони дизайну і класи, і сконцентруватися на інших розробках;
* Висока швидкість - динамічні макети Bootstrap масштабується на різні пристрої і дозволу екрану без будь-яких змін в розмітці;
* Гармонійний дизайн - всі компоненти платформи Bootstrap використовують єдиний стиль і шаблони за допомогою центральної бібліотеки. Дизайн і макети веб-сторінок узгоджуються один з одним;
* Простота у використанні - платформа проста у використанні, користувач з базовими знаннями HTML і CSS може почати розробку з Twitter Bootstrap;
* Сумісність з браузерами - Twitter Bootstrap сумісний з Mozilla Firefox, Google Chrome, Safari, Internet Explorer і Opera;
* Відкрите програмне забезпечення.

Основні інструменти Bootstrap:

* Сітки - наперед задані розміри колонок, які можна відразу ж використовувати, наприклад ширина колонки 140px відноситься до класу .span2, який можна використовувати в CSS описі документа;
* Шаблони - Фіксований або гумовий шаблон документа;
* Типографіка - Описи шрифтів, визначення деяких класів для шрифтів, таких як код, цитати і так далі;
* Медіа - Представляє деякий управління зображеннями і Відео;
* Таблиці - Засоби оформлення таблиць, аж до додавання функціональності сортування;
* Форми - Класи для оформлення форм і деяких подій відбуваються з ними;
* Навігація - Класи оформлення для табів, Вкладок, сторінковому, Меню і панелі інструментів;

# 3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ВЕБ-РЕСУРСУ ФОРМУВАННЯ РЕЙТИНГУ ПРОФЕСІЙНИХ ФОТОГРАФІВ ТА ЇХ РОБІТ

## 3.1. Основні етапи розробки веб-ресурсу. Налаштування робочого місця

Створення сайту - надзвичайно складний процес, тим більше що сайт розроблявся самостійно. Є багато студій веб-дизайну, які пропонують швидко і недорого зробити сайт. Проте хотілося, щоб сайт був унікальний і не коштував ні копійки – сайт розроблявся самостійно. Етапи створення сайту, приведені нижче.

**Розробка структури сайту:**

* визначення початкових даних для сайту;
* визначення вимог до зовнішнього вигляду і функціональності;
* формування структури сайту - розділів меню;

**Розробка концепції дизайну:**

* створення дизайн-макета головної сторінки сайту;
* затвердження концепції дизайну - макету головної сторінки;
* створення внутрішніх сторінок сайту і визначення змін в дизайні до внутрішніх сторінок;
* **Html-верстка, дизайн і створення внутрішніх сторінок:**
* розробка наповнення внутрішніх сторінок;
* розробка додаткових сторінок;
* оптимізація зображень;

**Програмування:**

* визначення завдань програмування;
* розробка структури баз даних;
* написання скриптів адміністрування.

Всі вище описані етапи проектування та розробки програмного продукту були проведені мною підчас проходження переддипломної практики на підприємстві SoftServe. Одним з початкових етапів був етап постановки завдання. Без цього етапу неможна було б визначити що розробляти, чим розробляти та як розробляти. На даному етапі було прийнято рішення розробити веб-ресурсу формування рейтингу професійних фотографів та їх робіт. В основу чого полягало розробити функціональний веб ресурс засобами веб розробки та розмістити його на сервері, для подальшого використання його користувачами.

Розробка структури сайту розпочалася з визначення початкових даних для сайту. Ними стала інформація про те як повинен виглядати та функціонувати розроблюваний ресурс. Даний етап включив в себе:

* визначення вимог до зовнішнього вигляду і функціональності;
* формування функціоналу сайту.

Для виконання цього пункту було прийнято спроектувати макет зовнішнього вигляду веб сайту, в основу якого було покладено основу побудови зовнішнього вигляду сайту. При розробці дизайну сайту потрібно врахувати безліч чинників. На сайті повинні бути правильно розставлені елементи управління, з урахуванням правил юзабіліті та інших технічних особливостей, необхідних для швидкої і якісної роботи ресурсу. Дизайн веб-ресурсів мав ряд особливостей і кардинально відрізняється від всіх інших напрямків візуального оформлення. Для розробки якісного дизайну, обов’язково використовуватимуться основні мови, що використовуються при написанні сайтів та технології просування в пошукових системах, тільки при дотриманні всіх правил можна розраховувати на успішний розвиток і якісну роботу веб-ресурсу.

Проектування прототипу сайту здійснювалося за допомогою програми Mockup, за допомогою час проектування прототипу був зменшений у декілька разів. За допомогою цієї програми були спроектовані прототипи головної сторінки Рисунок 3.1.

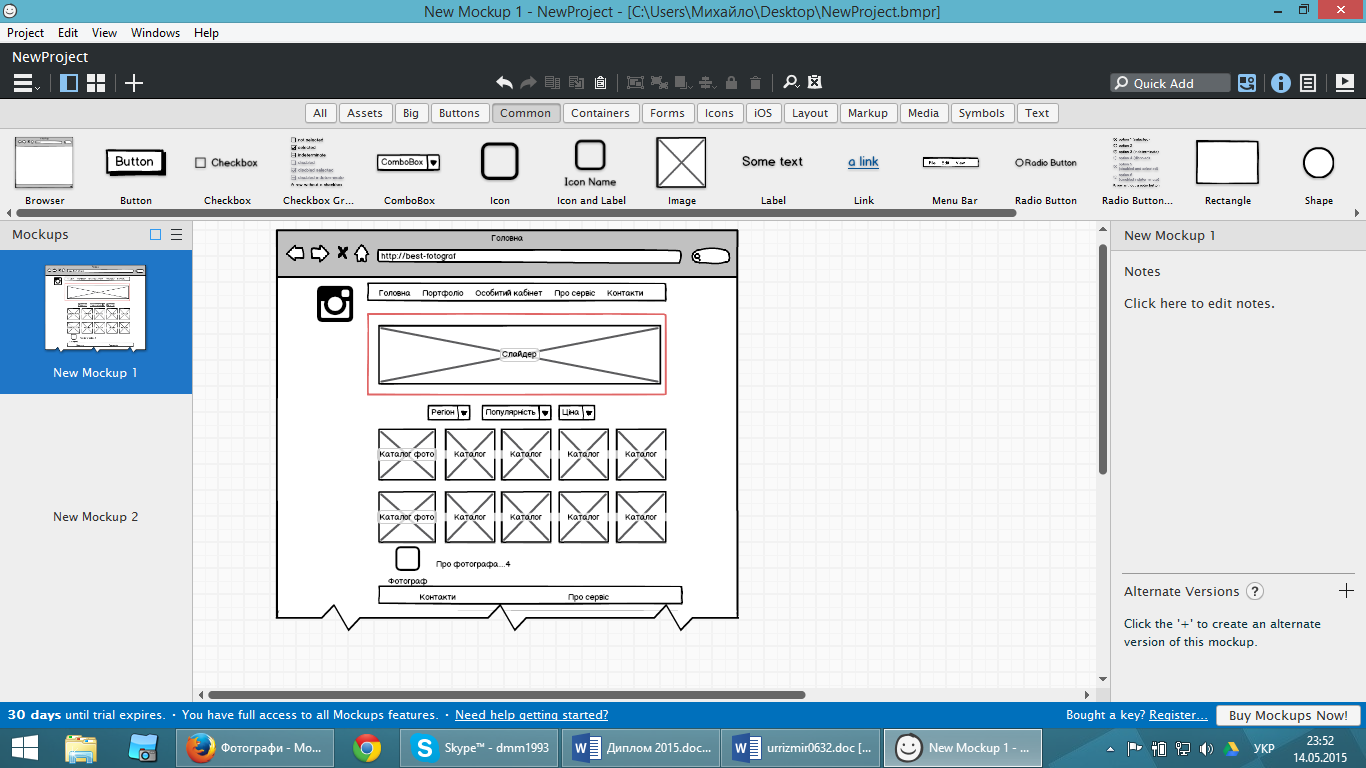


Рисунок 3.1 – Прототип головної сторінки

На якому ми можемо побачити те як будуть розміщатися основні компоненти головної сторінки. Даний прототип був створений для того щоб можна було, ще на етапі проектування підібрати оптимальне розташування компонентів та елементів веб ресурсу, та зменшити час на обдумання розміщення її в процесі верстання веб сторінок. Також за допомогою цієї програми було спроектовані проекти і інших додаткових сторінок які включає в себе веб сайт.

Наступним етапом розробки прототипу ПП був етап проектування функціоналу ресурсу, який показує зв’язки між елементами та частинами сайту рисунок 3.2.

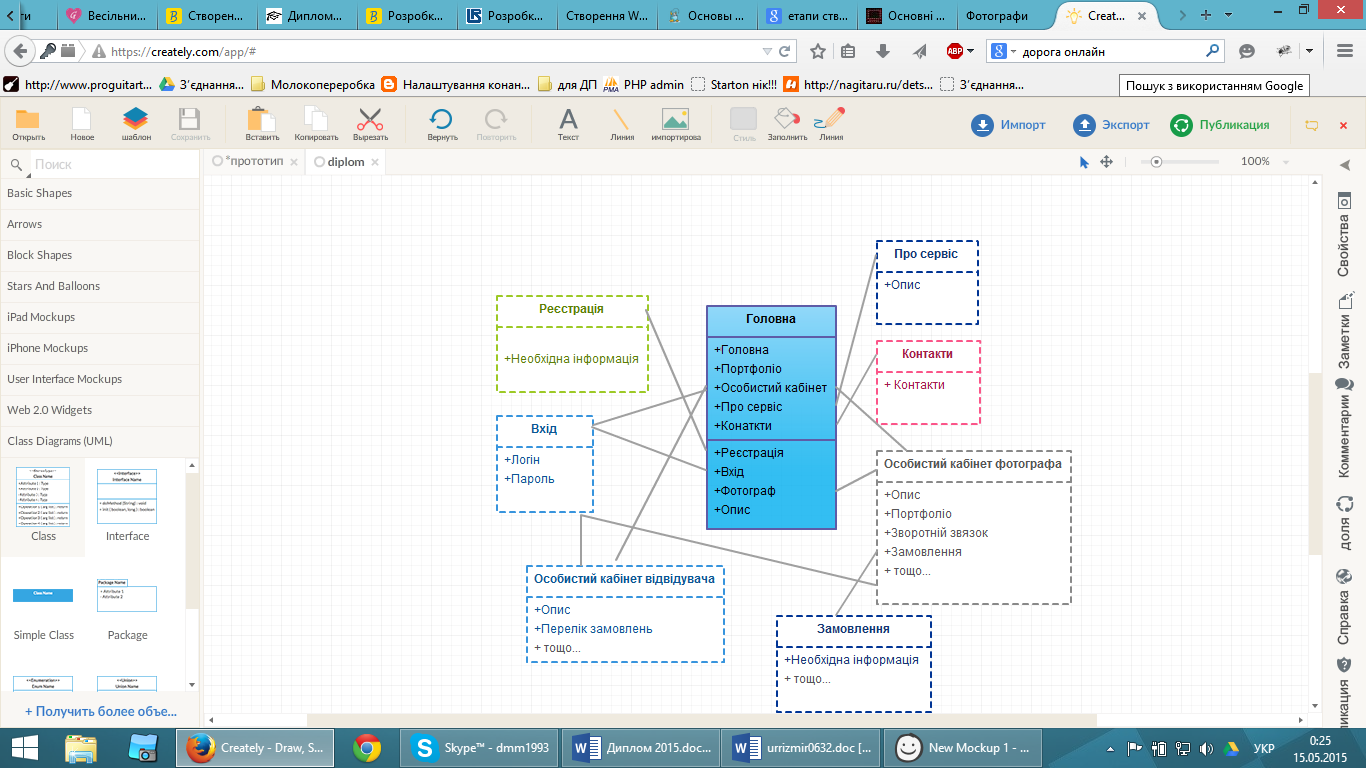


Рисунок 3.2 – Прототип функціоналу сайту

На даному рисунку ми можемо побачити приблизний зв'язок між функціональними частинами сайту та його компонентами. Даний прототип в процесі розробки буде дещо змінюватися, в залежності від подальших прийнятих рішень в процесі розробки.

Також до етапу проектування можна віднести і проектування БД сайту, яка буде підключена до нього та зберігатиме всі необхідні дані, які буде передавати чи зчитувати сайт. Прототип БД сайту розроблявся за допомогою безкоштовного онлайн інструменту creately.com. Даний інструмент є досить зручний, коли необхідно розробити прототип БД чи чогось іншого. Даний інструментарій дозволяє декількома кліками спроектувати БД, та показати зв’язки між її елементами (таблицями), для подальшого використання логіки зв’язку в процесі програмування. Але так як ми будемо розробляти прототип БД, то ми саме для проектування прототипу БД його і використаєм. Прототип БД представлено на рисунку 3.3.

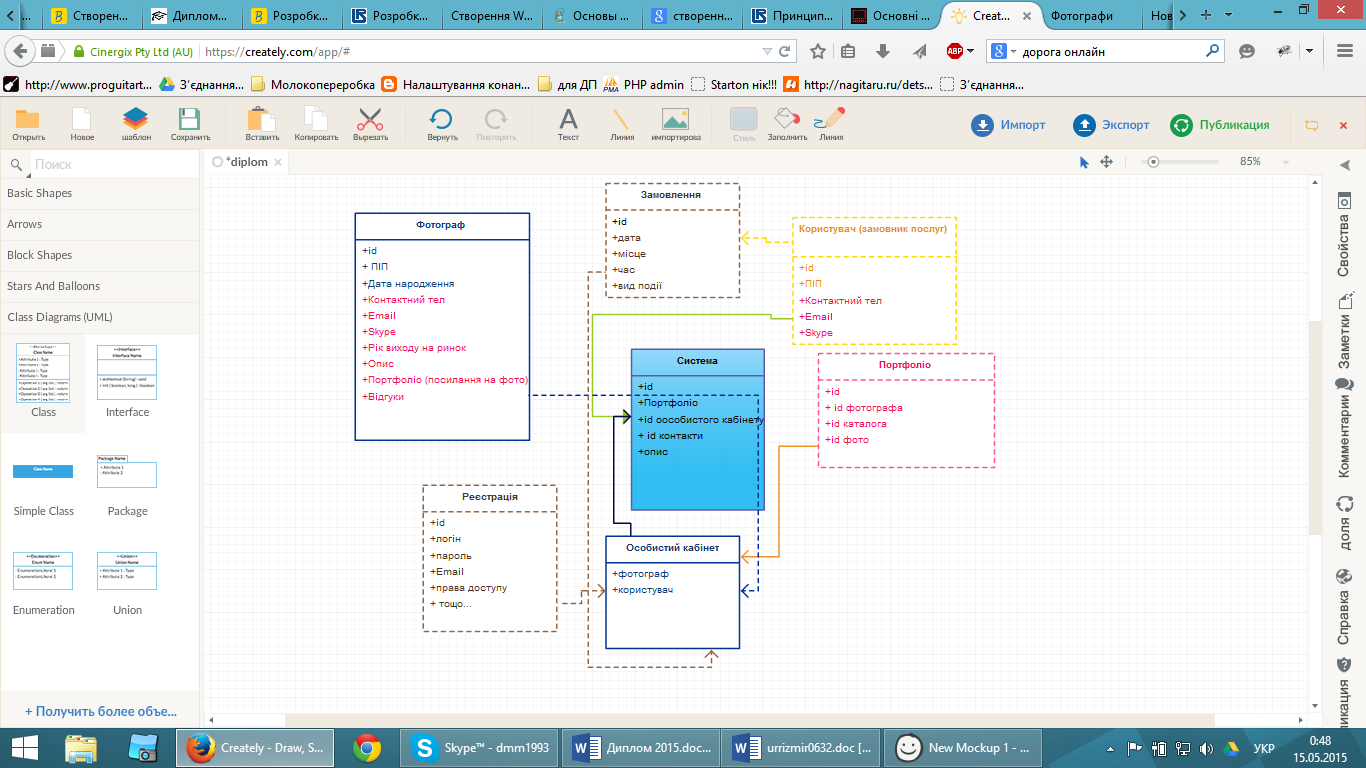


Рисунок 3.3 – Прототип БД

Отже на даному етапі було проведені підготовчі роботи для розробки ПП. Дані роботи полягали в проектування початкових прототипів оформлення, функціонування та зв’язку таблиць в БД розроблюваного сайту. Завдяки виконанню цього етапу було скорочено час на виконання на етапі програмування.

Після завершення розробки прототипів ПП настав час налаштувати робоче місце для подальшого виконання поставленого завдання. Під налаштуванням робочого місця слід розуміти: встановлення та налаштування всі необхідних серверів, фреймворків, редакторів коду та інших програмних засобів за допомого яких проводилася розробка сайту.

Перший крок налаштування робочого місця був крок з установки локального сервера “Denwer”.

Денвер - набір дистрибутивів і програмна оболонка, призначені для створення та налагодження сайтів (веб-додатків, іншого динамічного вмісту інтернет-сторінок) на локальному ПК (без необхідності підключення до мережі Інтернет) під управлінням ОС Windows. Вибір цього сервера полягав в тому що даний дистрибутив вміщає наступні елементи:

* Веб-сервер Apache з підтримкою SSI, SSL, mod\_rewrite, mod\_php;
* Інтерпретатор PHP з підтримкою GD, MySQL, SQLite;
* СУБД MySQL з підтримкою транзакцій (mysqld-max);
* Система управління віртуальними хостами, заснована на шаблонах;
* Система управління запуском і завершенням;
* Панель phpMyAdmin для адміністрування СУБД;
* Інсталятор.

Всі компоненти були необхідні для подальшої роботи над проектом.

Отже для встановлення даного сервера необхідно було перейти на його офіційну сторінку за адресую <http://www.denwer.ru/>, звідки необхідно було скачати установочний пакет та інсталювати на своєму ПК. Вигляд головної сторінки офіційного сайту ДЕНВЕР показано на рисунку 3.4.

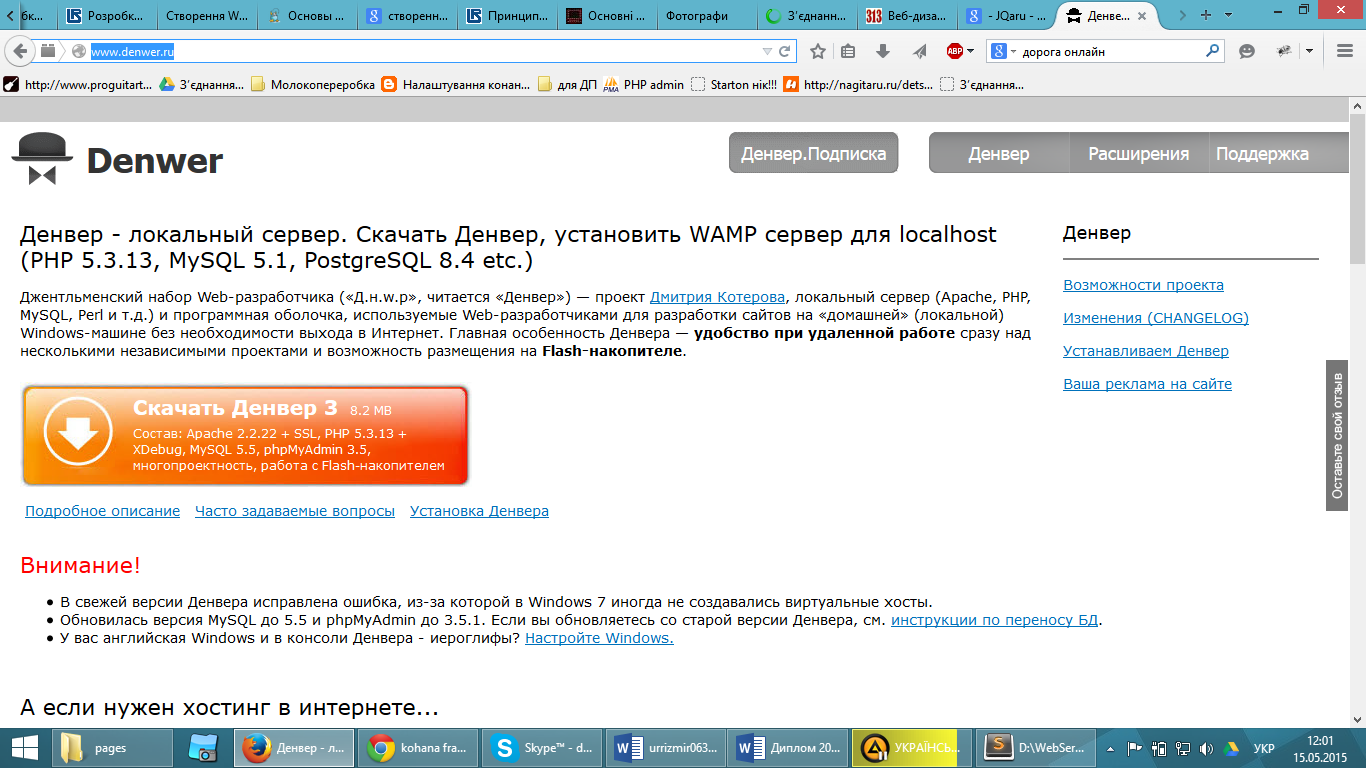


Рисунок 3.4 – Вигляд головної сторінки

Після скачування установочного пакету, необхідно було розпакувати та встановити програму. Для подальшої коректної роботи сервера необхідно провести ряд внутрішніх налаштувань. Після проведення всіх налаштувань ми отримали папку з всіма необхідними файлами сервера, дана папка буде служити місцем подальшого збереження нашого сайту. Структура та вміст кореневої папки сервера наведено на рисунку 3.5.

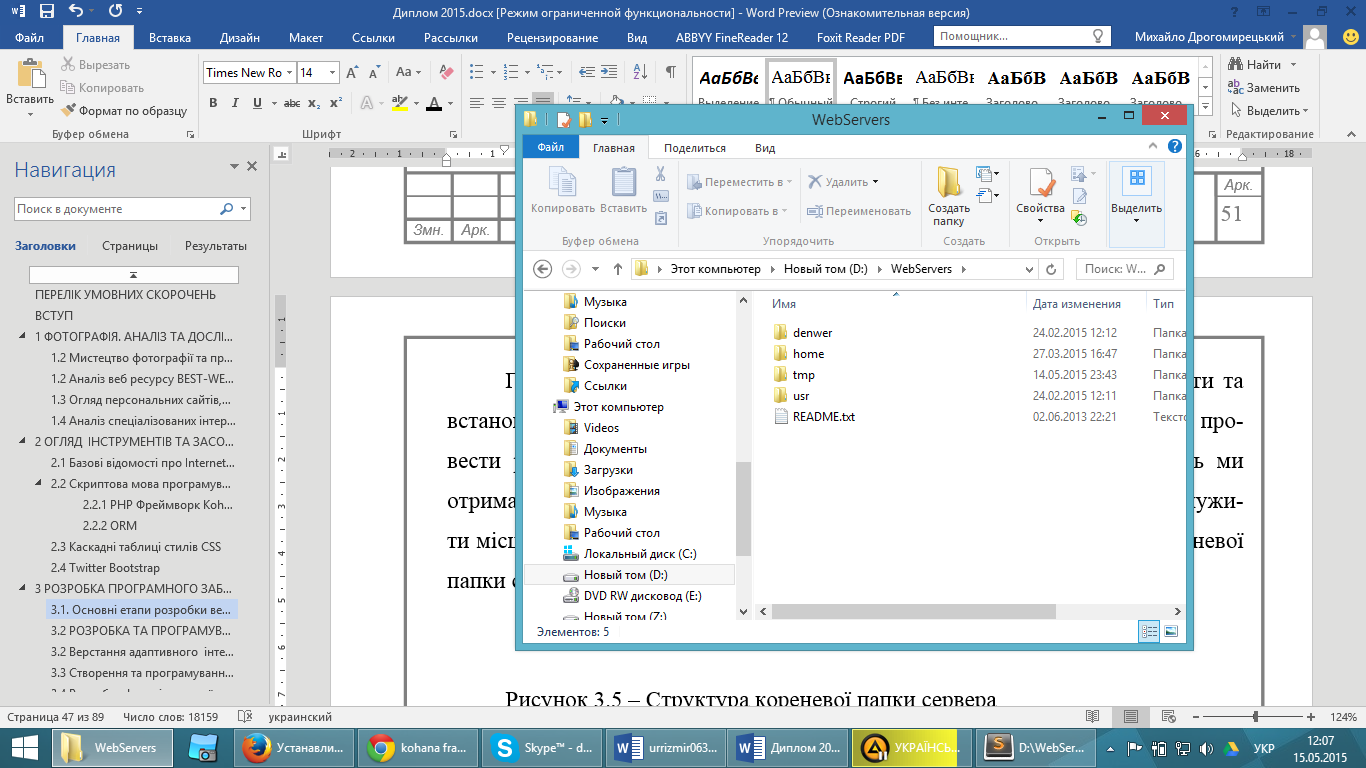


Рисунок 3.5 – Структура кореневої папки сервера

Наступним кроком налаштування робочого місця був крок встановлення та налаштування фреймворку “Kohana”, на основі якого проводилася розробка сайту.

Kohana (раніша назва — Blue Flame) — веб-фреймворк на мові PHP з відкритим кодом, який використовує архітектурну модель MVC (англ. Model View Controller). Його мета — бути безпечним, легким і простим у використанні. Цей фреймворк був вибраний через наступні його особливості:

* Висока безпека;
* Екстремальна легкість;
* Легкий у розумінні;
* Використовує MVC-модель;
* Повна сумісність з UTF-8;
* Наявність технології ORM;
* Дуже легко розширюється.

Також я його використав у проекті для того щоб не було повторень коду та повторного написання деяких функцій таких як авторизація та інших, адже даний фреймворк вже має вбудовані модулі які ці питання поставлені вирішувати. Для цього необхідно підключити ці модулі та використовувати як об’єкти цього модуля.

Для використання фреймворку “Kohana”, необхідно скачати його з офіційного сайту “Kohana” за адресую <https://kohanaframework.org/>, та встановити в себе на комп’ютері. Вигляд сторінки для скачування фреймворку показано на рисунку 3.5.

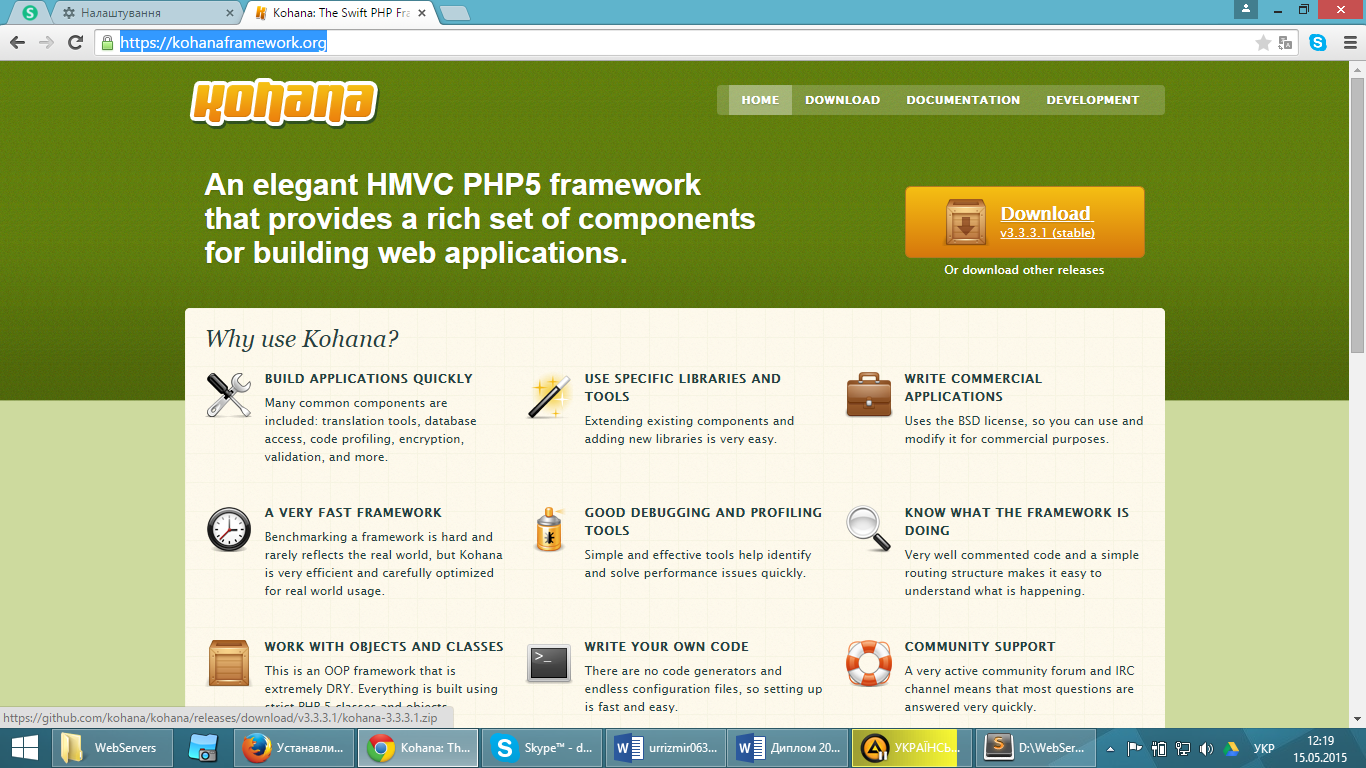


Рисунок 3.5 – Сторінка для скачування фреймворку

Але одного скачування фреймворку недостатньо, далі необхідно його розпакувати з архіву та помістити в корінь папки локального веб сервера ДЕНВЕР рисунок 3.6.

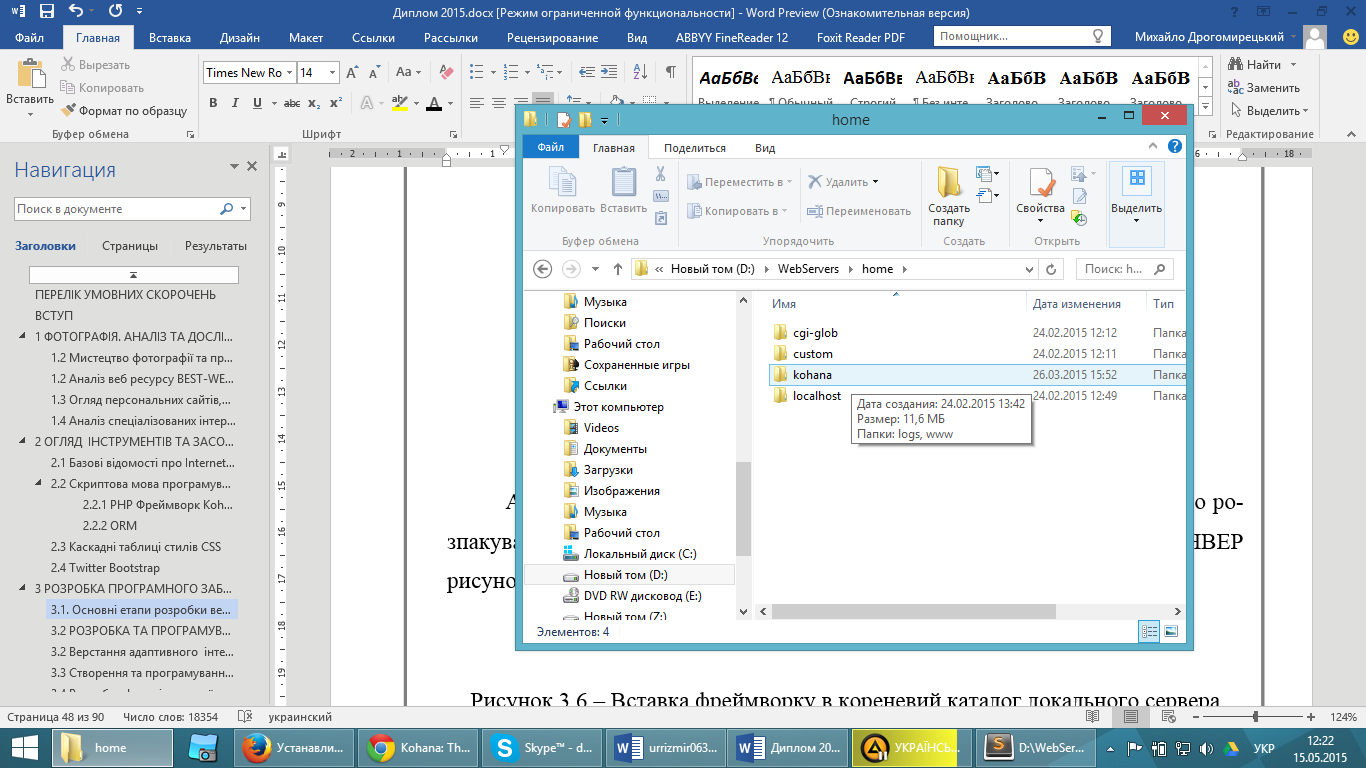


Рисунок 3.6 – Вставка фреймворку в кореневий каталог локального сервера

Після вставки розпакованого фреймворку в папку сервера необхідно провести ряд змін в основному конфігураційному файлі фреймворку “bootstrap”. Для цього необхідно відкрити цей файл та внести змінив його текстовий зміст. Текст з внесеними змінами наведено в (додатку А)

## 3.2 РОЗРОБКА ТА ПРОГРАМУВАННЯ ВЕБ-САЙТУ

Дуже важливий та трудомісткий етап розробки веб-ресурсу, саме від якості програмування залежить, на скільки, якісно буде працювати, і індексуватися пошуковими системами сайт. Як правило, терміни програмування залежать від технічних особливостей і функціональних можливостей сайту (чим більші  функціональні можливості сайту, тим складніше його програмувати). Якість програмування залежить від обраних технологій за допомогою яких і буде розроблятися веб сайт.

Як вже було сказано вище, вся розробка сайту проводилася засобами та інструментами для веб програмування. Для цього було використано наступні мови програмування та інструменти:

* PHP;
* HTML 5;
* JavaScript;
* CSS 3;
* Twitter Bootstrap;
* Kohana framework;
* jQuery;
* Та інше.

Всі веб-сторінки кодувалися на мові гіпертекстової розмітки – HTML та надавалася стилізація сторінок за допомогою каскадних таблиць стилів CSS3. Для зручного написання коду на цих мовах, використовувався редактор коду Subline Text 2. Даний редактор є досить зручний і легкий у використання та підтримує вище перелічені мови та їх методи, що значно скоротило час написання коду. Загальний вигляд редактора представлено на рисунку 3.4.

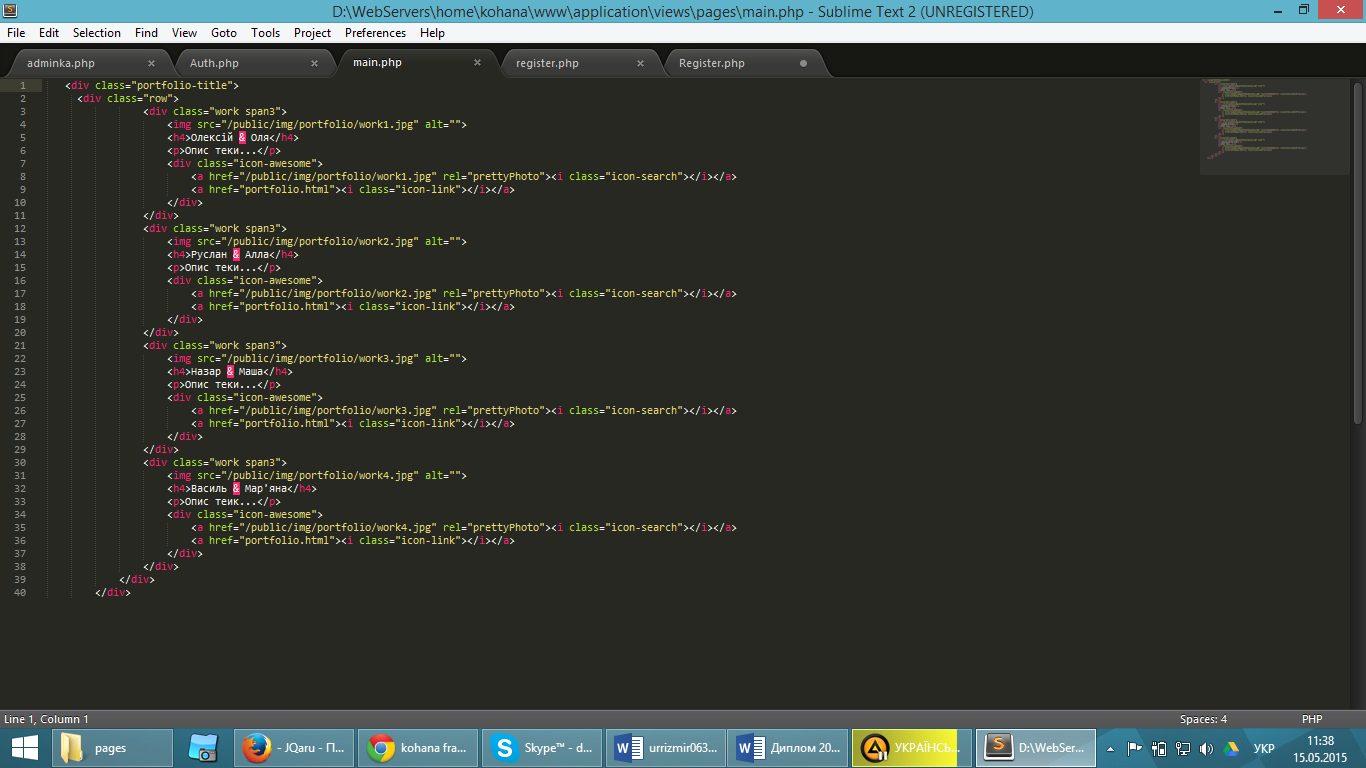
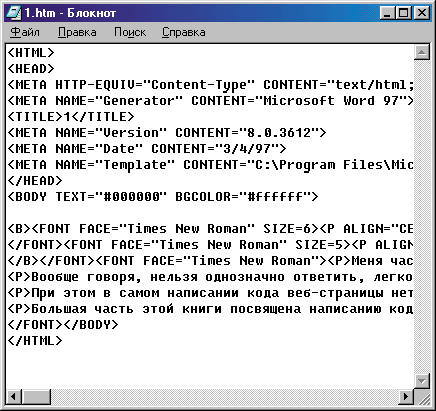


Рисунок 3.4 – Загальний вигляд редактора

Підчас верстання всі

Взагалі кажучи, щоб написати HTML-файл, досить мати будь-який текстовий редактор, лише б він умів не додавати в текст свої спеціальні символи. Найпростіший варіант - це редактор *Notepad (Блокнот)*, що входить в стандартне постачання Windows (мал. 2.1). Власне кажучи, це саме те, що потрібне, - проста програма, що зберігає написаний текст саме в тому вигляді, в якому він був введений, і нічого зайвого. (Тим, хто не працює під Windows, як заміна Блокноту підійде майже “все що завгодно”: Edit - для MS-DOS, vi або jed - для Linux, Kedit - для Linux/KDE і т. д.)



Мал. 2.1. Код HTML в текстовому редакторові Блокнот

Проте в дуже простих текстових редакторах типу Блокнота важ HTML-текст доводиться писати уручну, а багатьом хотілося б якусь частину роботи автоматизувати. Враховуючи це бажання, розробники створили спеціалізовані засоби, покликані полегшити працю веб-сервера-програміста (як іноді називають тих, хто пише код HTML, на відміну від веб-дізайнеров, які придумують зовнішній вигляд і іноді навіть намагаються утілити його, використовуючи програми роботи з графікою). Давайте спочатку стисло опишемо декілька простих програм, а потім зупинимося на кращих з них.

*Веб-сервер-редактор TextPad*

Отже, почнемо. Тим, хто вважає за краще набирати код HTML уручну, але кому не вистачає функціональності Блокнота і подібних до нього програм, можна порадити програму під назвою TextPad, яку можна завантажити, звернувшись за адресою www.textpad.com. Ця програма по суті вельми схожа на Блокнот, проте розробники спеціально передбачили деякі зручності для того, щоб писати код HTML (а також мов Java, З, C++, Perl і ще деяких). Це виражається в тому, що при написанні HTML -документа все теги автоматично підсвічують синім кольором, їх атрибути - темно-синім, а значення атрибутів - зеленим (кольори можна набудувати за власним бажанням, так само, як і шрифт). Це дуже зручно. Наприклад, якщо автор випадково помилиться в імені тега або атрибуту, то воно залишиться чорним, і він відразу зрозуміє, що тут щось не те. Правда, перевірка не є “ інтелектуальною”: програма може спокійно “дозволити” приписати тегу яка-небудь властивість, якої у нього в принципі бути не може (спокійно підсвічує абракадабру типу <BR ALIGN="top"> або </BR> ).

На відміну від Блокнота, TextPad - редактор багатовіконний. У нім можна відкрити відразу декілька документів і працювати, легко перемикаючись між ними за допомогою списку в лівій частині вікна або вкладок в нижній частині.

Веб-сервер-редактор TextPad дозволяє автоматизувати набір багатьох тегов. Якщо не хочеться набирати їх уручну (багато цього не люблять просто через те, що доводиться перемикатися на латинський шрифт), то звернете увагу на ліву нижню частину вікна програми. Там приведений список найбільш поширених HTML-тегов, які можна вставляти в свій основний текст подвійним клацанням миші. Правда, в списку вказані не самі теги, а їх опис. Наприклад, щоб вставити тег <BR>, потрібно вибрати із списку пункт Block > Break Line. Проте до цього швидко звикаєш. Майже всі пункти списку вставляють теги разом з тим, що закриває парним тегом. Наприклад, якщо вибрати пункт Block > Preformatted, в текст автоматично будуть додані теги і <PRE> і </PRE> . Деякі пункти додають відразу цілі блоки-заготовки. Якщо, наприклад, вибрати пункт Table (Таблиця), в текст буде вставлений такий код:

<TABLE ALIGN="left" BORDER=0 CELLSPACING=0 CELLPADDING=0 WIDTH="100%">

<TR ALIGN="left" VALIGN="middle">

<TH></TH>

<TH>?</TH>

<TR ALIGN="left" VALIGN=middle">

<TD> ? </TD>

<TD> ? </TD>

</TABLE>

Значення цих тегов і їх атрибутів ми розглянемо пізніше, а поки звернемо увагу на те, що окрім списку тегов TextPad надає нам також можливість вибирати із списку спеціальні символи (список HTML Characters), а також, якщо буде потрібно, будь-який символ, що управляє, наприклад символи псевдографіки DOS та інші. Ті, хто часто вводять які-небудь послідовності символів, що при написанні веб-сторінок не рідкість, можуть полегшити своє завдання, записавши в TextPad відповідні макроси. Для запису макросу треба натиснути комбінацію клавіш CTRL+SHIFT+R (або вибрати з меню Macros пункт Record). При цьому почнеться запис макросу, тобто все подальші дії запам'ятають. Щоб закінчити запис, треба знову натиснути комбінацію клавіш CTRL+SHIFT+R, після чого привласнити ім'я файлу макросу, а також дати назву для уявлення макросу в меню. Тут можна також дати, якщо потрібно, короткий опис макросу і вказати ім'я його автора. Після натиснення на кнопку ОК назва макросу з'явиться в меню Macros. Вибравши його, можна ввести відразу всю задану послідовність символів.

Для зручності відладки можна встановити прапорець в пункті Line Numbers (Нумерація рядків) в меню View (Вигляд), - в цьому випадку всі рядки тексту будуть пронумеровані. Хочеться відзначити, що якщо в меню Configure (Настройка) включений пункт Word Wrap (Перенос по словах) для автоматичного перенесення кінців довгих рядків у видиму частину екрану, то кожна такий довгий рядок все одно нумеруватиметься одним номером, а не двома або трьома (до речі, така нумерація чомусь недоступна в чудовій програмі Homesite, про яку мова піде нижчим). А якщо в меню View (Вигляд) включити прапорець Visible Spaces (Відображати пропуски), то можна побачити на екрані і “невидимі символи”, такі, як пропуски, символи табуляції та інші.

У програмі TextPad можна легко порівняти два файли, вибравши з меню Tools (Сервіс) пункт Compare Files (Порівняти файли). А якщо є два (або більш) схожі файли, в деяких місцях яких треба внести зміни, зручно використовувати функцію Synchronize Scrolling (Одночасна прокрутка) з меню Configure (Настройка). В цьому випадку можна відкрити відразу декілька файлів, наприклад, вибравши з меню Windows (Вікна) пункт Tile Vertically (Розташувати по вертикалі), і тоді при прокрутці одного з них інші прокручуються синхронно. Серед інших корисних функцій програми TextPad варто відзначити можливість автоматичної зміни клавіатурного регістра командою Edit >

Change Case (Правка > Змінити регістр), автоматичного копіювання в буфер слова або рядка, на якому знаходиться курсор, за допомогою команд Edit ” Cut Other (Правка > Вирізувати) і Edit > Сміттю Other (Правка > Копіювати), а також функцію перевірки орфографії Tools > Spelling (Сервіс > Правопис). І, звичайно, тут присутня можливість проглядання створеного файлу в броузере View > In Web Browser (Вигляд > У броузере).

*Веб-сервер-редактор Arachnophilia*

Завершивши короткий огляд можливостей програми TextPad, давайте розглянемо іншу програму для написання .HTML-кода - Arachnophilia, яку можна отримати за адресою www.arachnoid.com/arachnophilia.

Як і у попередньому випадку, програма автоматично підсвічує синім кольором теги і атрибути, а значення атрибутів - зеленим, що покращує зорове сприйняття, хоча перевірка правильності тегів і відсутній. Зате якщо випадково забути закрити тег, то все подальший вміст файлу підсвітить бордовим кольором, так що помилка відразу кинеться в очі. У програмі Arachnophilia передбачена автоматизація введення тегов HTML. У нижній частині вікна є кнопки, кожна з яких відкриває відповідну кнопкову панель. На цих панелях розташовані кнопки для швидкого введення тегов. Наприклад, якщо натиснути кнопку Styles (Стилі), то відкриється панель управління, що містить кнопки для введення тегов форматування тексту . Натиснення на будь-яку з них вводить тег або блок-заготівку. Наприклад, після натиснення на кнопку BR в тексті з'явиться тег <BR>, а після натиснення на кнопку HR - з'явиться відразу ціла заготівка: <HR WIDTH="95%" ALIGN=CENTER>.

Натиснення деяких кнопок викликає появу діалогових вікон. Наприклад, натиснувши на кнопку Color (Колір), можна відкрити стандартне діалогове вікно вибору кольору, а за допомогою кнопки TableWiz (Майстер таблиць) відкривають діалогове вікно Table Wizard (Майстер таблиць), засобами якого можна заздалегідь задати кількість рядків і стовпців в таблиці, а також визначити її ширину, колір ліній і деякі інші параметри .

Суттєвою особливістю програми є можливість легкого перегляду веб-сторінки в різних броузерах, для чого в меню Preview (Попередній перегляд) передбачений пункт Identify Browser (Вказати Лроузер). Тут можна призначити до шести різних броузеров, в кожному з яких легко відкрити створюваний HTML-файл для перегляду, навіть не зберігаючи його на диску.

Проте найчудовішою властивістю програми, мабуть, є функція Instant View (Негайний перегляд), яка доступна в меню Preview (Попередній перегляд). Якщо вона включена і внутрішній броузер, який також є в програмі, відкритий, то всі зміни, що вносяться до тексту HTML, негайно відображаються на екрані! Правда, в деяких випадках, програма не встигає за введенням даних і зображення у вікні броузера може зникати. Але не хвилюйтеся, а введіть наступний символ, і зображення знову з'явиться.

У програмі Arachnophilia існує безліч додаткових корисних функцій. Наприклад, в меню Selection (Фрагмент) є команда Find/Replace/Count (Поиск/Замена/Пересчет), яка дозволяє швидко знайти або замінити потрібні слова у виділеній області, що часто дуже виручає при створенні веб-сторінок. Команда Tag Delimiters (Обмежувачі тегов) з того ж меню дозволяє перетворити кутові дужки, що є загальноприйнятими обмежувачами тегов HTML, в спеціальні символи &lt і &gt, що необхідне, коли треба показати код HTML на самій веб-сторінці. Можна також здійснити зворотне перетворення. У цьому ж меню є команда Strip all HTML tags (Приховати теги HTML), за допомогою якої можна швидко звільнити текст від HTML-тегов, наприклад, для перенесення його в іншу програму. Програма Arachnophilia, до речі кажучи, здатна читати і записувати файли формату RTF(Rich TextFormat). При відкритті RTF-файлу пропонується конвертувати його у формат HTML, але його можна редагувати і в звичайному вигляді.

Цікаво, що програму Arachnophilia 4.0 можна завантажити як в повному вигляді (це настановний файл розміром півтора мегабайти), так і в скороченому (1 Мбайт), якщо в системі встановлені необхідні бібліотеки. Можна також завантажити тільки виконуваний файл, а решта дозавантажувати по мірі необхідності.

Також існують програми, що дозволяють редагувати веб-сторінки як би в режимі WYSIWYG.

*Веб-редактори типу WYSIWYG*

Насправді зазвичай з цього нічого хорошого не виходить. Це пов'язано з тим, що автор створює не код, а оформлення сторінки, після чого програма автоматично підбирає для неї код, який відповідає тому, що задумав автор. Зазвичай на сторінці виявляється багато абсолютно зайвого коду. Він може залишатися, наприклад, від відмінених проб, не говорячи вже про те, що програма може сама вставляти коментарі, які тільки уповільнюють завантаження сторінки.

Ефективно управляти оформленням сторінки у такий спосіб теж не вдається. Тому ми довго не затримуватимемося на веб-редакторах, що працюють по прнципу WYSIWYG.

Взагалі кажучи, для редагування HTML-тексту в режимі WYSIWYG можна використовувати навіть такий текстовий процесор, як Microsoft Word. Починаючи з версії MS Word 97 він дозволяє набрати деякий текст, відформатувати його і зберегти у форматі HTML. Якщо будете це робити, не забудьте видалити коментарі...

*Редактор Star Office*

Могутніші засоби редагування HTML-файлів має в своєму розпорядженні програма StarOffice. Тут при відкритті або створенні HTML- файлу відповідно міняється вміст деяких меню, що дозволяє достатньо ефективно працювати з HTML кодом. Найприємнішим моментом тут, мабуть, є можливість встановити прапорець HTML Source (Початковий код HTML) в меню View (Вигляд), який відключає режим WYSIWYG і дозволяє працювати з початковим HTML-текстом, в якому всі теги і їх атрибути підсвічуються червоним кольором. На свій розсуд можна редагувати як початковий текст, так і результат, що відображається, перемикаючись між режимами командою HTML Source (Початковий код HTML) з меню View (Вигляд). На відміну від інших WYSIWYG-редакторів, StarOffice досить коректно видаляє непотрібні елементи при відміні користувачем яких-небудь дій і не вставляє зайвих коментарів. При цьому він досить активно використовує каскадні таблиці стилів (CSS).

Перед підсвічуванням теги перевіряються на коректність - помилково написані теги червоним кольором не виділяються. Проте треба мати на увазі, що програма не розуміє нових тегов, таких, як <BUTTON>, <LABEL> і ін.

*Netscape Composer*

Ще одна WYSIWYG-орієнтована програма для редагування .HTML-файлів вбудована в броузер Netscape. Вона називається Netscape Composer значно менше, ніж в попередній, проте, оскільки це модуль популярного броузера, її теж використовують активно. Засоби верхньої інструментальної панелі дозволяють вставити у файл зображення (Image), горизонтальну лінію (H.Line), таблицю, гіперпосилання (Link) або якір (Anchor). Нижче розташована невелика панель для управління зображенням шрифту (напівжирне, курсивне, підкреслене), а також його розміром, відступом і виключенням. Тут же можна вибрати колір фону. За допомогою випадних меню доступні ще декілька HTML-елементів.

Не дивлячись на доступність і популярність програми, ми навряд чи можемо порадити використовувати WYSIWYG-редактори типу Netscape Composer для створення веб-сторінок, за винятком яких-небудь дуже простих випадків. Якщо вам неодмінно хочеться візуально відстежувати всі зміни, що вносяться, то встановіть програму Arachnophilia і скористайтеся функцією Instant View.

Відзначимо, що всі розглянуті в цьому розділі програми розповсюджуються безкоштовно. В принципі, таких засобів, як TextPad і Arachnophilia, цілком достатньо, щоб відчувати себе комфортно при створенні веб-сторінок будь-якої складності. Проте деякі розробники пропонують ще більш просунуті засоби розробки веб-сторінок, прагнучи забезпечити максимальну зручність для користувача і автоматизацію рутинної роботи. На жаль, як правило, такі програми вже не безкоштовними, але із-за зручності у використанні вони теж користуються великою популярністю. Далі ми перейдемо до розгляду таких “просунутих” засобів створення веб-сторінок.

**2.3. Графіка на веб-сторінці**

На сьогоднішній день в Інтернеті залишилися не так багато сторінок без графічних елементів, оскільки за допомогою графіки можна не тільки проілюструвати пропонований матеріал, але і небагато “пожвавити” загальне оформлення, про що мова піде в цьому розділі.

*Фонове зображення*

Отже, давайте розберемося, яким же чином можна використовувати графічні елементи на веб-сторінці. Розглянемо такий приклад. Допустимо, ми хочемо, щоб читачі дізналися достоїнства нашого коханого електронного піаніно - клавінови. Ми вже написали відповідний текст, але хочемо його оформити не дуже строго. І тут найпростішим рішенням буде розмістити цей текст не на одноколірному, а на тематичному фоні.

Допустимо, що у нас заготовлено відповідний фоно вий малюнок - він записаний у файлі під ім'ям back1 .jpg (мал. 2.2). Якщо при окремому перегляді цього малюнка він вам подобається, це ще нічого не означає. Щоб визначити, чи підходить малюнок як фон до тексту, його потрібно перевірити “в справі”.

Щоб помістити малюнок на веб-сторінку як фон, потрібно встановити в теге <BODY> атрибут BACKGROUND= із значенням, що містить ім'я файлу малюнка.

<BODY BACKGROUND""Images/back1.jpg">

Якщо малюнок менше вікна броузера, то він повторюватиметься по горизонталі і по вертикалі. Це відбувається автоматично, і до недавнього часу це не можна було відмінити. Тепер же, якщо ви не хочете повторення



Мал. 2.2. Фоновий малюнок

Можна вказати або відносне місцеположення файлу малюнка (тобто шлях доступу до нього відносне розташування початкового HTML-файлу), або його повну URL-адресу. Тут і далі ми припускаємо, що ви розмістили в тій же теці, що і HTML-файл, теку, у яку складаєте всі графічні файли.

І, нарешті, можна проілюструвати веб-сторінку, вставивши перед описом необхідну фотографію (наприклад, фото працівника організації). Припустимо, що фотографія у нас вже готова і лежить в папці під ім'ям kadr\_1.gif. Вставити її в текст можна з помо щью тега <IMG>, наприклад, от так:

<IMG SRC="Images/kadr\_1.gif">

В принципі, це вже достатньо - на місці цього запису броузер відобразить нашу картинку. Проте краще вказати в теге <IMG> ще декілька атрибутів.

По-перше, треба пам'ятати, що картинки завантажуються з мережі завжди набагато повільніше, ніж звичайний текст. Це я говорю не до того, що сам файл картинки повинен бути не дуже великого об'єму (це вірно, але це питання ми обговоримо окремо), а до того, що при такому записі, як вище, багато броузери, почавши завантажувати картинку, припинять завантаження подальшого тексту до тих пір, поки картинка не буде завантажена повністю. Але для читача було б зручніше, якби текст продовжував завантажуватися gараллельно з картинкою: поки вона довантажується, можна було б щось вже читати. Щоб броузер міг це зробити, йому потрібно явно вказати розміри майбутньої картинки за допомогою атрибутів WIDTH= і HEIGHT=:

<IMG SRC="Images/clavi.gif" WIDTH="300" HEIGHT="200">

Взагалі кажучи, за допомогою цих атрибутів можна задати які завгодно розміри картинки, “розтягнувши” або стиснувши її як завгодно. Проте зазвичай цього не роблять, і ось чому. Представляти картинку меншою, ніж вона є, недоцільно, оскільки навіщо тоді вантажити великий файл? Адже файли для картинок великого розміру мають великий розмір і вантажаться з Інтернету, відповідно, набагато довше. Краще вже зробити маленький файл для картинки маленького розміру і швидко завантажити його. А якщо представити картинку більшою, ніж вона є, то, як правило, вона виглядатиме не вельми цікавою за рахунок появи “ступінчастості” як у формах, так і в кольорі.

Далі, хорошим тоном є наявність так званого альтернативного тексту для тих випадків, коли в броузері відключено проглядання графіки. Тоді на місці нашої картинки буде виведений прямокутник, який відповідає розміру, який супроводжується поясненням, що за картинка на цьому місці повинна бути. Альтернативний текст вводиться як значення атрибуту ALT:

<IMG SRC="Images/clavi.gif" WIDTH="300" HEIGHT="200" АLТ="Працівник Петров В.І.">

Нарешті, навколо малюнка можна відобразити рамку. Товщина цієї рамки задається як значення атрибуту BORDER=. Якщо ми, як в даному випадку, не хочемо відображати рамку навколо малюнка, то можемо нічого і не указувати, оскільки більшість броузеров за умовчанням рамку не малюють. Але деякі броузери все ж таки це роблять, і “для вірності” непогано б вказати і відсутність рамки теж:

<IMG SRC="Images/clavi.gif" WIDTH="300" HEIGHT="200" АLТ="Клавинова" BORDER="0">

Чого ще не вистачає нашому HTML-файлу? Звичайно ж, гіперпосилань! Без них будь-який документ в Інтернеті сприймається набагато нудніше, оскільки відвідувач відразу бачить, що зайшов в безвихідь. Тому обов'язково додайте гіперпосилання скрізь, де це тільки можна. В даному випадку, наприклад, слово “Організація” просто проситься стати гіперпосиланням на сторінку, де розміщено опис відповідної організації. Зробимо це:

<A HREF="http://www.organizaciya.com">Oрганізація</A>

До речі, картинки теж можуть містити гіперпосилання.

Ще слід відмітити, що ми не дарма визначили в нашому прикладі колір фону (за допомогою атрибуту BGCOLOR=). Адже якщо у користувача в броузері відключено завантаження графіки, то білий текст у нього відображатиметься на заданому за умовчанням білому фоні, і в результаті нічого не буде видно.

До речі, тег <IMG> може мати власний атрибут ALIGN=, який управляє розташуванням малюнка щодо тексту. У нашому прикладі це було не актуально, оскільки малюнок знаходився в окремому блоці <DIV> . Але якщо малюнок “вбудований” в текст, тобто декілька варіантів його розташування:

ALIGN="left" - текст обтікає малюнок, який розташовується зліва;

ALIGN="right" - текст обтікає малюнок, який розташовується справа;

ALIGN="top" - малюнок розташовується усередині текстового рядка, і його верхня межа співпадає з верхньою межею букв тексту (можна також застосовувати значення "texttop");

ALIGN="middle" - малюнок розташовується усередині текстового рядка, який вертикально вирівнюється по центру малюнка (можна також при міняти значення "absmiddle");

ALIGN="bottom" - малюнок розташовується усередині текстового рядка, і його нижня межа співпадає з нижньою межею букв тексту (можна також застосовувати значення "absbottom" або "baseline").

Крім того, щоб малюнок не дуже “притискався” до тексту, можна залишити між ними трохи “повітря”. Для цього в теге <IMG> використовують атрибути HSPACE= і VSРАСЕ=, значення яких задають відповідно горизонтальне і вертикальне “поля” малюнка.

І ще невелике зауваження: броузер Internet Explorer (версія 4 і вище) дозволяє за допомогою тега <IMG> завантажувати і проглядати також відеофрагменти у форматі AVI (Video for Windows). Для цього можна встановити атрибут DYNSRC=, як значення якого вказати ім'я AVT-файлу. При цьому можна додатково вказати в атрибуті START=, коли слід починати програвання відео: "FILEOPEN" - відразу після завантаження файлу або "MOUSEOVER" - після наведення покажчика миші на зображення. Атрибут LOOP= дозволяє встановити кількість програвань: "О" означає нескінченне повторення, алюбоє інше число указує точна кількість повторів.

**ВИСНОВКИ**

Отже, зі сказаного вище, можна зробити наступні висновки:

**Розробка веб-сайту йде поетапно. Головні етапи:**

**Розробка структури сайту:**

* визначення початкових даних для сайту;
* визначення вимог до зовнішнього вигляду і функціональності;
* формування структури сайту - розділів меню;

**Розробка концепції дизайну:**

* створення дізайн-макета головної сторінки сайту;
* затвердження концепції дизайну - макету головної сторінки;
* створення внутрішніх сторінок сайту і визначення змін в дизайні до внутрішніх сторінок;

**Html-верстка, дизайн і створення внутрішніх сторінок:**

* розробка наповнення внутрішніх сторінок;
* розробка додаткових сторінок (карта сайту, результати пошуку і т.п.);
* оптимізація зображень;

**Програмування:**

* визначення завдань програмування;
* розробка структури баз даних;
* написання скриптів адміністрування.

Вимоги до контенту сайту досить прості: контент сайту повинен бути написаний простою, зрозумілою мовою, орієнтований на Вашу аудиторію сайту.

Краще доручити створення контента професійним копірайтерам, таким чином, ви захистите свій сайт від неякісних, згенерованих програмами текстів. Оптимізація контенту необхідна для того, щоб ваш сайт піднімався в результатах пошуку по запитах на ключові слова. Оптимізатори допрацьовують контент написаний копірайтерамі, і в результаті скорочуються витрати на платну рекламу в пошукових системах. **Назви програм, які стануть в нагоді при створенні сайту:**

# Html-Reader 1.7 Rus - редактор для перегляду, коректування і створення нових html-сторінок.

# Мини-Сайт - невелика, проста і зручна програма фірми "Kors-Soft" для створення сайту про фірму.

# CatsHtml 1.5 Rus - редактор html, малий розмір, багато можливостей.

# CoffeeCup DHTML Menu Builder - програма для швидкого створення різних меню для Web-сторінок.

# AI HTML Utilities 1.6 - набір утиліт для розробки веб-сайтів: редактор HTML-текстів, CSS-файлів, утиліта для підбору кольорів, оптимізатор коду, збірка корисних кодів HTML і таблиця спеціальних символів.

# HTML Optimizer v 2.5 Rus - програма коректно видаляє непотрібні пропуски, лапки, коментарі і т.п. в HTML-коді, унаслідок чого об'єм сторіночки помітно зменшується (в середньому - на 25 %) і прискорюється її завантаження.

# Clear HTML v.1.54 - програма видаляє з коду сторінок непотрібні пропуски і упорядковує вміст.

++++++++++++++++++++++++++++++++

## 3.2 Верстання адаптивного інтерфейсу засобами веб розробки

+++++++++++++++++++++++++++++++++++++++++

## 3.3 Створення та програмування БД ресурсу

+++++++++++++++++++++++

## 3.4 Розробка функціональної частини проектованого продукту

++++++++++++++++

## 3.5 Опис функціональних можливостей та ілюстрація роботи створеного веб додатку

+++++++++++++++++++++++++++++++

# 4. ЕКОНОМІЧНА ДОЦІЛЬНІСТЬ РОЗРОБКИ ТА ВИКОРИСТАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## 4.1. Обґрунтування ринкової доцільності розробки веб ресурсу фотографів

Всі програмні продукти, в тому числі і веб ресурси, які розробляються на даний час, необхідно обґрунтувати з точки зору економічної [3] доцільності. Дане обґрунтування необхідне для того, щоб вчасно припинити (при втраті актуальності або надмірних витратах) розробку або здійснити необхідні інвестування в проект для забезпечення необхідними програмними або апаратними засобами розробників з метою одержання очікуваних результатів. Економічний ефект розробленого продукту визначається на основі економічних показників, які дають можливість прогнозувати результат від впровадження даної програми.

Враховуючи інтенсивний розвиток комп'ютерної техніки, на сьогодні такий аналіз є невід’ємною частиною попереднього аналізу аналогічних робіт, оскільки саме результат автоматизації виробничих процесів дає суттєве покращення в технології виробництва чи діагностування об'єктів, а кошти, що затрачаються на дану роботу, повинні бути еквівалентними тому ефекту, який принесе конкретне нововведення.

В даній роботі проводиться розрахунок економічних показників та аналіз всієї роботи по розробці алгоритмічного та програмного забезпечення веб-ресурсу формування рейтингу професійних фотографів та їх робіт. Дана розробка позитивно вплине на подальше збільшення клієнтів фотографів, оскільки потреба в таких ресурсах і Інтернеті зростає з кожним днем, і фотографам для того щоб отримувати прибутки, не обхідно захоплювати ринок фото індустрії. Таким чином, зі зростанням кількості фотографів (люди, які заробляють в галузі надання фото послуг) зростає пропозиція, і водночас виникає необхідність у ресурсах які допоможуть оптимально підібрати фотографа на своє свято, таким чином залишаючи конкурентів поза зоною досяжності.

Web-сайт - це своєрідний інтерфейс між тими хто надає певні послуги та тими хто зацікавлений в отриманні такого роду послуг. Тому створення сайту є одним з головних завдань виконання поставленої задачі під час виконання дипломного проекту.

Нині все більша кількість фотографів починає використовувати Internet для обміну інформацією, представлення своїх робіт широкому загалу та оптимізації роботи з клієнтами. Структурована інформація сайту дає змогу партнерам і клієнтам отримати повне уявлення про фотографа і його діяльність.

Споживач одержує з сайту багато корисної інформації про конкретного фотографа і його діяльність, також з веб-ресурсу можна дізнатися ціни, перелік замовлених дат та замовити фотографа вказавши всю необхідну інформацію в формі замовлення.

Такий спосіб представлення інформації потенційним клієнтам ефективніший за звичайні засоби комунікації: пошту, телефон, факс. За допомогою таких обов'язкових елементів, як інтерфейс або навігація, клієнт може самостійно вибрати необхідну для перегляду інформацію. Сайт є доповненням до рекламної кампанії, забезпечуючи можливість зворотного зв'язку й інтерактивної роботи з фотографом.

До аналогів розроблюваного ресурсу можна віднести наступні:

* best-wedding;
* girko.net;
* paramoloda.ua;
* та інші персональні сайти та спільноти в соціальних мережах.

Веб-ресурс «best-wedding» - це ресурс який спеціалізується на наданні різного роду послуг, спрямованих на організацію та проведення урочистих подій. Даний веб-ресур представляє собою веб сторінку, яка знаходиться за адресую: «http://www.best-wedding.com.ua/». На якій представлено у випадковому порядку інформацію про:

* весільні агенції;
* музикантів;
* фотографів;
* операторів.

Даний ресурс не дозволяє певною мірою отримати в’ю необхідну інформацію про фотографа так як він є більш глобальний ніж той ресурс який було розроблено під час виконання дипломного проекту. Він не дає можливості чітко зазначити всю необхідну інформацію про фотографа, а отже і не надає можливості здійснювати пошук фотографів за конкретними параметрами. На даному ресурсі реалізовано лише регіональний пошук, який є потрібний але не є цілком об’єктивний при виборі фотографа. На цьому сайті є можливість перегляду фотознімків представлених певним фотографом, але не має можливості робити відгуки та оцінювати кожний знімок окремо. В той час як в розроблюваному проекті є можливість оцінювати кожен знімок та залишати відгук за якими в подальшому буде проводитися відображення фотографів по рейтингу (відповідно до сумарної кількості оцінок всіх його фото робіт).

Що до персональних сайтів, виникає деяка проблема з їх пошуком та переглядам їх вмісту. Адже кожен персональний сайт сильно відрізняється від будь якого іншого персонального сайту. В той час розроблюваний ресурс містить в собі особистий кабінет який і є окремим сайтом-візиткою фотографа.

Потенційними користувачами розроблюваного програмного продукту є фотографи, які розміщують свої роботи на даному ресурсі та іншу корисну інформацію про свою діяльність. Також цей сайт має іншу категорію користувачів – це відвідувачі сайту, які шукають фотографів на свою подію, бажають ознайомитися з роботами того чи іншого фотографа та обрати найкращого.

## 4.2. Розрахунок собівартості й ціни розробки (вказати назву розробки)

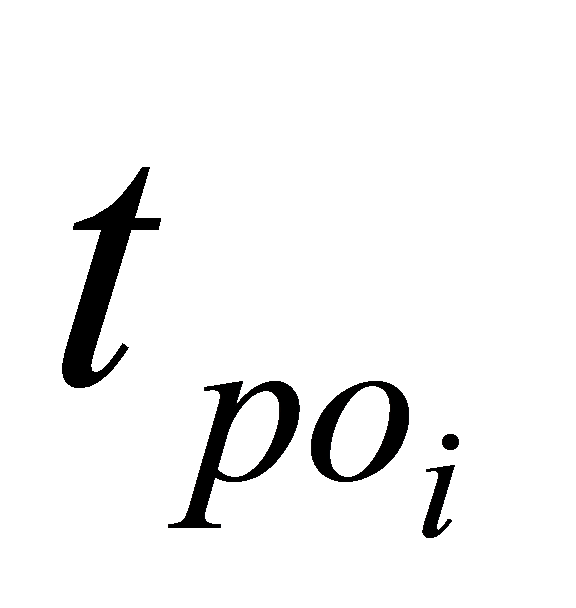
*В даному параграфі дипломної роботи:*

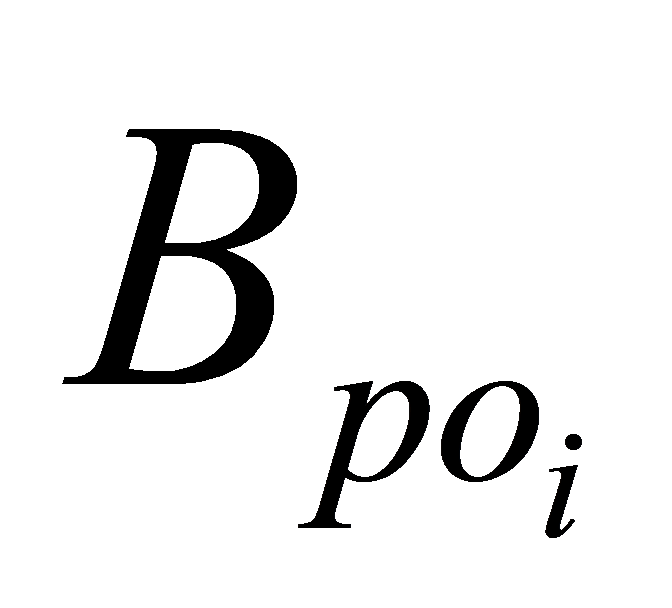
1. Дається визначення собівартості, ціни розробки, їх факторів та зазначається вплив на економічні результати діяльності суб’єкта господарської діяльності.

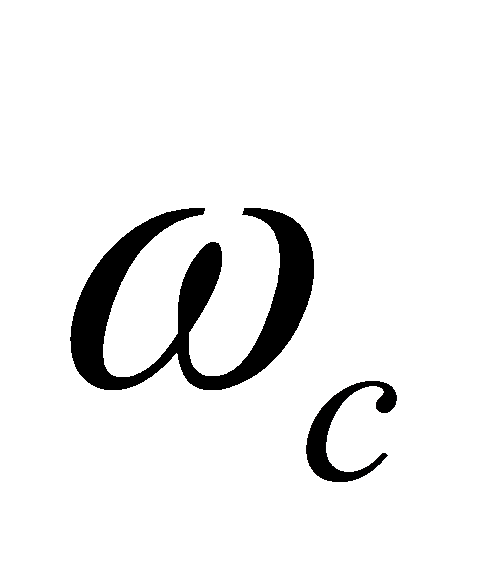
2. Вказується обрана методика та проводиться розрахунок собівартості та ціни розробки.

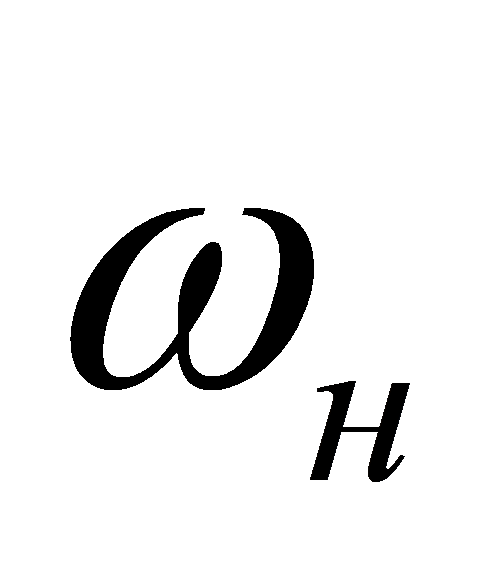
Наприклад. Оскільки собівартість відображає рівень витрат на [виробництво](http://uk.wikipedia.org/wiki/%D0%92%D0%B8%D1%80%D0%BE%D0%B1%D0%BD%D0%B8%D1%86%D1%82%D0%B2%D0%BE), то витрати на розробку () можна визначити за формулою:

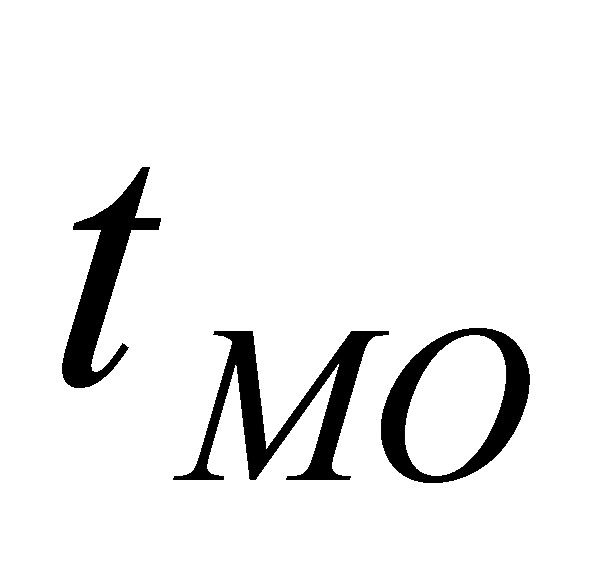
 (4.2.1)

де  *-* час, що витрачається на розробку працівником *і*-ої кваліфікації, люд.-міс;

- основна заробітна плата розробника *і*-ої кваліфікації, грн./міс.;

 - коефіцієнт, що враховує нарахування органам соціального захисту на заробітну плату, у відсотках від основної та додаткової заробітної плати;

 - коефіцієнт, що враховує накладні витрати установи, в якій виконується розробка, у відсотках до основної заробітної плати розробника;

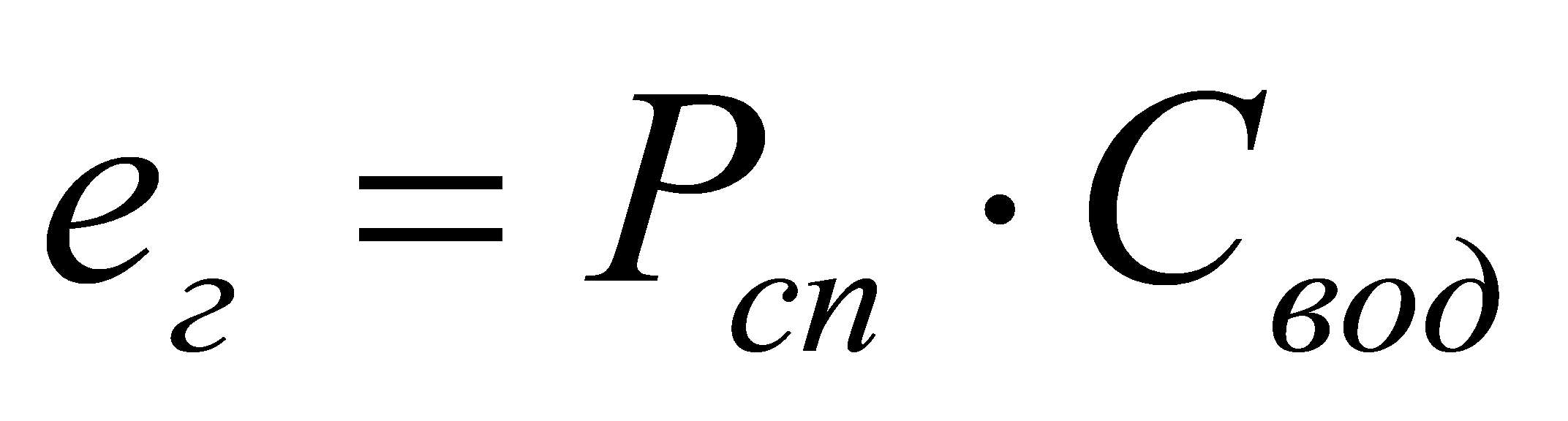
 *-* машинний час ЕОМ, необхідний для налагоджування розробки, машино-год.;

*ег -* експлуатаційні витрати, що припадають на 1 год. машинного часу, грн.;

- всі інші витрати підприємства (матеріальні витрати, витрати на придбання спец. устаткування, витрати на роботи, які виконують сторонні підприємства та організації, інші витрати та накладні витрати, грн./ розробку;

 - кількість видів витрат.

Експлуатаційні витрати, що припадають на 1 год. машинного часу визначаються за витратою електроенергії:

 (4.2.2)

де *Рсп -* споживана потужність ЕОМ;

*Сеод* - вартість 1 кВт/год електроенергії для підприємств;

Таблиця 4.2.1. - Кошторис витрат на розробку

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Найменування елементів витрат, в т.ч. інші витрати | Сума  витрат, грн. | |
| 1. |  |  | |
| 2. |  |  | |
| Всього | |  |

Ціну  розробки визначимо за формулою:

 (4.2.3)

 – ціна розробки, грн./розробку;

– норма рентабельності, що враховує прибуток установи, яка здійснює розробку, % від основи (собівартості);

– витрати на розробку, грн./розробку;

 – сума податків з врахуванням системи оподаткування для обраного суб’єкта господарювання, грн./розробку.

3. За результатами розрахунків формуються висновки.

## 4.3. Розрахунок можливого прибутку фірми-розробника та терміну окупності розробки.

*В даному параграфі дипломної роботи:*

1. Дається визначення прибутку, терміну окупності розробки з зазначенням їх впливу на економічні результати діяльності суб’єкта господарської діяльності.

2. Вказується обрана методика та проводиться розрахунок прибутку фірми розробника та терміну окупності розробки.

*Наприклад. Величину річного чистого прибутку* П можна розрахувати за формулою:

 (4.3.1)

де Д - дохід, який підприємство отримує за рік, грн.;

Вт - витрати підприємства за рік, грн.;

Вр - виручка від реалізації продукції за рік, грн.;

 – сума податків з врахуванням системи оподаткування для обраного суб’єкта господарювання, грн.

- загальна ставка податку, %.

Дохід, який підприємство отримує за рік розраховуємо за формулою:

 (4.3.2)

Виручку від реалізації продукції Вр можна розрахувати за формулою:

 (4.3.3)

де - ціна реалізації виробу г-го найменування, грн.;

- кількість виробів і - го найменування, які були реалізовані за рік, шт.;

n - кількість найменувань видів виробів.

*Витрати підприємства * за рік можна розрахувати за формулою:

 (4.3.4)

Таблиця 4.3.1 – Аналіз доходу в залежності від кількості споживачів

|  |  |  |
| --- | --- | --- |
| № п/п  року (місяця) | Кількість споживачів | Дохід, грн |
|  |  |  |

*Якщо планується, що проект реалізовуватиметься як соціальний (неприбутковий), або вузькоспеціалізований, доцільно провести порівняльну характеристику затрат користувача до і після використання розробки і визначити його як величину прибутку.*

Таблиця 4.3.1 – Порівняння затрат користувача на виконання певних завдань (вказати яких) без використання розробки ( вказати якої) та після.

Наприклад. Порівняння затрат фірми користувача на виконання 7 проектів за 1 рік з сталою заробітною платою (120грн/день).

|  |  |  |
| --- | --- | --- |
| Назва порівняльного параметру | Виконання робіт вручну | Виконання робіт з використанням створеної програми |
| Тривалість розробки проекту | 40 днів | 25 днів |
| Час на виконання 7 проектів | 280 днів | 175 днів |
| Витрати на оплату праці | 33600 грн. | 21000 грн. |
| Експлуатаційні витрати | 956,48грн. | 597,8грн. |
| Загальна сума фінансових витрат | 34556,48 грн. | 21597,8грн. |

Очікуваний річний прибуток в результаті автоматизації ручної роботи визначимо за формулою:

*ПУ = (ЗР – ЗТ) × (1 –ПП),* грн. *(4.3.5)*  (4.3.1)

де  ЗР – річні експлуатаційні затрати при ручній обробці, грн.

ЗТ – річні поточні затрати, пов’язані з експлуатацією програми, грн..

ПП – податок на прибуток, %

Пу = (34556,48 -21597,8)∙(1-0,19)= 10496,53грн.

Економія часу на виконання 7 проектів – 105днів.

За результатами вище проведених розрахунків проведемо розрахунок терміну окупності розробки за формулою:

То = , років (4.3.6)

де  То – термін окупності розробки;

П – значення прибутку, отриманого в результаті реалізації розробки, грн.;

І – розмір інвестицій (витрат), необхідних для реалізації розробки, грн.

4. За результатами розрахунків формуються висновки.

## 4.4 . Аналіз конкурентоспроможності розробки (вказати розробку)

*В даному параграфі дипломної роботи:*

1. Провести порівняльну характеристику програмних продуктів.

Наприклад. Серед найбільш поширених розробок, що забезпечують схожий принцип роботи розробленого продукту можна віднести (вказати розробки, проаналізовані у параграфі 4.1)

Порівняльна характеристика за основними показниками даного програмного продукту і аналогів (продуктів) наводиться в таблиці 4.4.1 (Приклад)

Таблиця 4.4.1 - Порівняльна характеристика програмних продуктів

|  |  |  |  |
| --- | --- | --- | --- |
| Назва показника | Коефіцієнт вагомості, | Кількість балів, | |
| Вказати назву аналога (прототипу) | Розроблена система |
| 1. Технічні показники:  1. Якість товару |  |  |  |
| 1.1. Надійність |  |  |  |
| 1.2. Зручність |  |  |  |
| 1.3. Інтерфейс  ....... |  |  |  |
| 2. Економічні показники:  2.1. Ціна, грн..  2.2. Витрати, пов’язані з використанням, грн.  2.3. ..... |  |  |  |

2. За результатами порівняльних характеристик провести аналіз конкурентоспроможності розробки із використаннямкоефіцієнта конкурентоспроможності за обраною методикою.

3.Підвести підсумки по параграфу

Висновки по розділу вцілому.

Таблиця - Зведені техніко-економічні показники розробки та реалізації (вказати назву розробки)

|  |  |  |
| --- | --- | --- |
| **Показник** | **Розмірність** | **Значення** |
| Витрати на:   * розробку програмного забезпечення * реалізацію розробки * ….   Податок...  ....  Дохід  Прибуток  Термін окупності  Індекс конкурентоспроможності |  |  |

# 5 ОХОРОНА ПРАЦІ

## 5.1 Значення охорони праці для користувачів ПК

Науково-технічний процес призводить до корінної зміни характеру і засобів праці. Цей процес створює для людини велику кількість благ:

* зменшує фізичне навантаження;
* робить працю інтелектуальною, цікавою, різноманітною;
* розвиває творчі здібності людини.

Широке впровадження комп'ютерної техніки істотно підвищує продуктивність праці.

Збереження здоров'я користувачів, персональних ПК, всіх працюючих, підтримання ефективності та надійності їх праці на належному рівні є одним із аспектів застосування дисципліни - охорона праці.

Широкомасштабні заходи, спрямовані на поліпшення здоров'я людей, повинні здійснюватись на кожному підприємстві у встановленому законодавчому порядку. Суворе додержання умов гігієни та фізіології праці є не тільки особистою справою людини, але й колективу, оскільки порушення принципів гігієни позначається не тільки на здоров'ї порушника, але й інших членів колективу.

Комплекс організаційних заходів і технічних засобів захисту, накопичений досвід роботи показує, що є реальна можливість добитися успіхів щодо усунення впливу небезпечних і шкідливих факторів на працюючих з комп'ютерами. Проте, особи, допущені до роботи з комп'ютерною технікою, все ще зазнають впливу небезпечних і шкідливих виробничих факторів, таких як монотонність, підвищена температура зовнішнього середовища, відсутність або недостатня освітленість робочої зони, електричний струм, статистична електрика та інших.

Праця багатьох працівників пов'язана з впливом психофізичних факторів, таких як розумове перевантаження, перенапруження зорових і слухових аналізаторів, емоційні перевантаження.

Нещасні випадки під час роботи за комп’ютерами можуть статися внаслідок дій фізично-небезпечних виробничих факторів при підключенні периферійних пристроїв зі знятим кожухом корпусу системного блока, при ураженні електричним струмом, виконанні працівниками невластивих для них робіт. Електричні установки становлять потенційну небезпеку для людини як у процесі експлуатації, так і під час проведення профілактичних робіт.

Дослідження, проведені фахівцями Всесвітньої організації охорони здоров'я (ВООЗ) показали, що у професійних операторів та канцелярських службовців, які у своїй діяльності використовують монітор, частіше зустрічаються порушення органів зору, опорно-рухового апарату центральної нервової, серцево-судинної, імунної та статевої систем, захворювання шкіри.

Дослідження медиків-гігієністів, психологів, світлотехніків та фахівців з охорони праці та ергономіки показали, що сучасна професія користувача монітор належить до розумової праці, яка характеризується: високою напруженістю зорових функцій; одноманітною позою; великою кількістю стереотипних високо координованих рухів, що виконуються лише м'язами кистей рух на фоні малої загальної рухової активності; значним нервово-емоційним компонентом, особливо в умовах дефіциту часу; роботою з великими масивами інформації, що викликає активізацію уваги та інших вищих психічних функцій. Крім того, при роботі з дисплеями на електронно-променевих трубках виникає вплив на користувача цілої низки факторів фізичної природи - електростатичні поля, радіочастотне та рентгенівське випромінювання тощо.

Встановлено, що стан організму користувача значно залежить від типу роботи з монітор та умов її виконання. В загальному усі користувачі комп'ютерів поділяються на професіоналів та непрофесіоналів.

Для кожної категорії користувачів комп'ютерів характерні свої особливості впливу комплексу несприятливих факторів трудового процесу та умов праці.

Основним потенційним джерелом небезпеки для людини в ОЦ є електричні установки, до яких відноситься практично все обладнання ПК, так як в процесі експлуатації або проведенні профілактичних робіт людина може торкатись частин установок, що знаходяться під напругою.

Специфічна небезпека електроустановок в ОЦ: провідники під струмом, корпуси настільної ПК та іншого обладнання, що знаходиться під напругою в результат пошкодження (пробою) ізоляції не подають ніяких сигналів, що попереджують людину про небезпеку. Реакція людини на електричний струм виникає лише при протіканні його через тіло людини.

Небезпека електричного струму на відміну від інших небезпек полягає в тому, що людина не в стані без спеціальних приладів виявити напругу дистанційно, як, наприклад, рухомі предмети чи механізми, розпечені об'єкти, відкриті люки, необгороджені краї майданчиків, які знаходяться на висоті, і т.п. Небезпека виявляється занадто пізно - коли людина вже уражена.

Електротравматизм людини можливий, коли при експлуатації електроустановок не виконуються необхідні технічні умови, коли електроустановка експлуатується на низькому технічному рівні, коли не виконуються вимога ПУЕ і ПТБ (Правила улаштування електроустановок і Правила техніки безпеки). Електротравматизм - це результат порушення вимог правил техніки безпеки та інструктажів, відсутності технагляду та аварійного режиму роботи електроустановок. Це наслідок незадовільної ізоляції струмопровідних частин, перехід напруги на корпус електроустановок, обрив проводів (і як наслідок - крокова напруга, напруга дотику). Причиною електротравматизму в багатьох випадках є непродумані, помилкові дії оперативного обслуговуючого персоналу.

Одним з найсуттєвіших факторів впливу на людину є випромінювання моніторів комп'ютерів. Види електромагнітних випромінювань електронно- променевих трубок (ЕПТ), що використовуються в моніторах ПК, середні дані замірів [7], нормовані значення та їх діапазон наведені в таблиці 5.1.

Таблиця 5.1 - Види електромагнітних випромінювань ЕПТ

|  |  |  |  |
| --- | --- | --- | --- |
| Види випромінювань | Діапазон | Фактичні (середні) дані замірів | Нормовані значення |
| Ультрафіолетове | 220-280нм | 0 0-0.01 Вт/м2 | 0.01 Вт/м2 |
| ІЧ-випромінювання | 320-400нм | 3,5-6,0 Вт/м2 | 10 Вт/м2 |
| Електростатичне поле | 400-700нм | 0.05-4.5 Вт/м2 | 100 Вт/м2 |
| Електричний струм | 50 Гц | 15 кВ/м | 20-60 кВ/м |
| Яскравість | 700 нм-1мм | U=220В; І=2А | U=220В; І=0.1А |

Інфрачервоне випромінювання шкідливо впливає на зір, втомлюючи очі, при тривалій дії порушує нормальне сприйняття кольору оком людини.

Наступним за ступенем впливу на людський організм є фактор освітлення. Недостатнє або надто сильне освітлення впливає на органи зору, призводить до втоми очей, їх напруження що, в свою чергу, викликає загальну втому працівників. Можна стверджувати, що цей фактор є основним при аналізі зорової роботи, до якої відноситься робота за комп'ютером. Велику роль також відіграє розміщення робочих місць по відношенню до джерел природного освітлення - вікон. Приміщення з ПК слід розміщувати з вікнами на північ або захід, а самі відео термінали так, щоби не створювати відблисків на моніторах, які впливають на ступінь розрізнення текстово-графічної інформації. Отже, наведені вище фактори є основними при роботі в ОЦ (обчислювальний центр), і саме їх впливом можна пояснити причини виникнення скарг користувачів ПК (таблиця 5.2) і професійних захворювань.

Таблиця 5.2 - Характеристика скарг користувачів ПК, що працюють з ЕПТ, порівняно з контрольною групою

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Симптоми втоми | Відсотки від кількості досліджуваних | | | |
| Користувачі ППК | | | Мікромонтажниці |
|  | Інженери - програмісти | оператори | учні |
| Загальна втома | 29.4 | 22.2 | 38 | 97 |
| Млявість | 17.6 | 18.5 | 7.7 | 43 |
| Сонливість | 11.8 | 7.4 | 11.5 | 0 |
| Головний біль | 17.6 | 18.5 | 15.4 | 54 |
| Важкість у голові | 11.8 | 11.1 | 3.8 | 51 |
| Втома м'язів рук | 5.8 | 14.8 | 15.4 | 36 |
| Різь в очах | 58.8 | 37 | 7.7 | 67 |
| Розпливчатість меж | 35.3 | 55.5 | 19.2 | 4 |
| Сумарно-очні симптоми | 94.1 | 92.6 | 26.9 | 71 |

Третім фактором є виробничий шум. В умовах офісних приміщень при відсутності роботи машин і механізмів, що є джерелами шуму, зазвичай рівень шуму і вібрацій є в нормі.

## 5.2 Забезпечення нормальних умов праці та Розрахунок освітленості робочого місця

Однією з основних умов праці є освітленість робочого приміщення. Від освітленості у виробничому приміщенні залежить комфортність умов праці. Правильно спроектоване і виконане виробниче освітлення покращує умови зорової роботи, запобігає втомлюваності оператора, сприяє продуктивності працюючого персоналу, підвищує продуктивність праці та зменшує можливість отримання професійного захворювання рисунок 5.1.

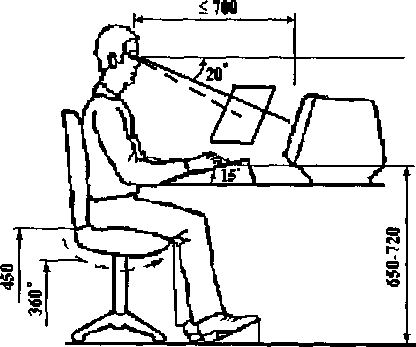


Рисунок 5.1 – Положення оператора підчас роботи

Для умов роботи програміста характерний ІІІ (високої точності) розряд зорової роботи, що характеризується значенням освітленості, наведеної в таблиці 5.3.

Таблиця 5.3 - Характеристика штучної освітленості робочих місць

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Назва примі-  щення | Розряд зорової роботи | Освітленість, лк. | | | | Тип світиль-  ника |
| Загальне освітлення | Комбіноване освітлення | Авар.Осві. для роботи | Авар.Осві. для евак. |
| Офіс | ІІІ | 300 | 1000 | 15 | 50 | ЛБ-40 |

Наведемо розрахунок природного освітлення. Для проведення розрахунку природного освітлення потрібно метричні параметри приміщення. Виходячи з ДСанПІН 3.3.2.007-98 [8], а також, беручи до уваги характер робіт, відповідно до яких, площа приміщення на одного працівника в приміщенні становить 6м2. Визначаємо необхідну площу для роботи за формулою:

(5.1)

де - площа приміщення, що відводиться на одного працівника;

п - кількість працівників.

Оскільки в приміщенні працює одна людина, тоді необхідна площа для роботи має становити не менше:

; (5.2)

Реальні розміри приміщення становлять: довжина - 2,5 м; ширина - 3 м; висота - 3.2 м, висота робочої поверхні . Тобто площа приміщення становить:

; (5.2)

що відповідає вимогам санітарних норм рисунок 5.2.

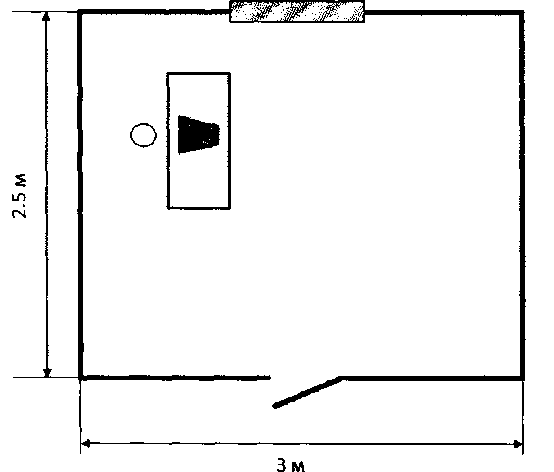


Рисунок 5.2 – Схема організації робочого місця у приміщенні

Далі знаходимо необхідну площу вікон:

(5.3)

де Ін - нормоване значення КПО ( для зорової роботи середньої точності і при бічному освітленні); - площа підлоги;

- значення світлової характеристики вікон (визначається нижче);

- загальний коефіцієнт світло пропускання вікон;

- коефіцієнт запасу (при концентрації пилу < 1 мг/м3 і природному освітленні під кутом).

= 1,8- коефіцієнт, який враховує відбивання світла від поверхні

- коефіцієнт, який враховує затемнення вікон будинками, які стоять навпроти.

Визначимо спочатку необхідні для розрахунку значення. Нормоване значення КПО знайдемо, скориставшись табл.:

(5.4)

де І = 1,5 - значення коефіцієнту природного освітлення ( для зорової роботи 3-го розряду),

М - коефіцієнт світлового клімату, М = 0,9;

С - коефіцієнт сонячності клімату, С = 0,85.

(5.5)

Приймаємо коефіцієнт запасу Кз = 1,8.

Значення світлової характеристики вікон визначається відношеннями:

(5.6)

(5.7)

З таблиці знаходимо = 14. Площа підлоги рівна .

Оскільки вікна не мають світлозахисних пристроїв і виготовлені з подвійних дерев'яних рам, в яких уставлене віконне скло, то за знайденими значеннями в таблицях визначаємо загальний коефіцієнт світло пропускання вікон:

(5.8)

- загальний коефіцієнт світло проникання

; - коефіцієнти світло пропускання матеріалу вікна, виду вікна та його конструкції.

- коефіцієнт, який враховує витрати світла в сонцезахисних конструкціях;

- коефіцієнт, який враховує витрати світла в захисній сітці, яка встановлюється під лампою.

Всі значення коефіцієнтів: ,Ег, ьу, Еа беруться із відповідних таблиць. Отже:

(5.9)

Отримуємо необхідну площу світлових отворів:

(5.8)

Кількість вікон визначаємо за формулою:

; (5.10)

де - стандартна площа вікна.

Відповідно: вікно.

Для освітлення приміщення, коли природного освітлення недостатньо, або немає, використовується штучне освітлення. Воно поділяється на загальне, місцеве (концентрує світловий потік безпосередньо на робочому місці) і комбіноване (поєднання загального і місцевого).

Для освітлення приміщення будемо використовувати люмінесцентні лампи. Оскільки спектральний склад їх світла найбільш близький до природного, вони в 2 - 3 рази економічніші, мають більший коефіцієнт корисної дії, не втрачають енергії на нагрівання (максимальна температура досягає 40 - 50°С), світлова віддача в 2 - 4 рази більша ніж у ламп розжарювання. До їх недоліків слід віднести явище стробоскопічного ефекту (миготіння), якого можна уникнути застосувавши схему дволампового включення або включення трьох ламп у різні фази трьохфазної мережі.

Розрахунок проводимо за методом коефіцієнта використання світлового потоку з урахуванням нормованої потужності освітлення.

Метод коефіцієнта використання світлового потоку визначає світловий потік ламп, необхідний для досягнення заданої освітленості з урахуванням світла, відбитого від стін, стелі і робочої поверхні, тобто сумарний світловий потік Ф£,лм:

(5.11)

де - нормована освітленість (для люмінесцентних ламп =300 лм);

S- площа приміщення (8=7.5 м );

К - коефіцієнт запасу (К = 1,4 );

- коефіцієнт мінімальної освітленості ( = 1,2);

- коефіцієнт використання світлового потоку (відношення потоку, що падає на розрахункову поверхню до сумарного потоку всіх ламп; визначається за показниками приміщення, типом світильника та коефіцієнтами відбиття світла від стін pстін, стелі рстел та підлоги рпід). Показник приміщення визначається за формулою:

(5.12)

де А, В - розміри приміщення;

Н - висота підвісу світильника над робочою поверхнею.

Для прийнятого приміщення А=2.5м, В=3м, а Н=2.5м.

(5.13)

Для освітлення вибираємо люмінесцентні лампи денного світла ЛБ-40-1 з наступними параметрами:

Напруга живлення, В - 220;

Потужність споживання, Вт - 80;

Світловий потік, лм - 4320;

Світлова віддача, лк/Вт - 48,3

Для усунення стробоскопічного ефекту лампи у світильнику використовуються попарно. Вибираємо світильник НОГЛ - 2x80 з габаритами 1655 х 230 х 380 мм. Для світильників цього типу при заданому значенні і=0,625; рпід =60%; pстін,=35%; рстел=15% знаходимо з таблиці значення = 15%. Світловий потік становитиме:

(5.14)

Кількість ламп:

N = Ф / Ф л, (5.15)

де Ф. = 4300 лм - освітленість однією лампою:

(5.16)

Оскільки у світильнику ОДО застосовується по дві лампи, то необхідно використовувати 3 світильники.

## 5.3 Забезпечення електробезпеки

При регламентованому технічному обслуговуванні, обов'язковій перевірці з періодичністю не рідше 1 разу на рік, підлягають захист пристроїв ПК від перевантажень по струму і виконаний на базі реле максимального струму, захист від коротких замикань. При заміні елементів захисту не допускається застосовувати реле з само поверненням.

Електричний опір і міцність ізоляції в пристроях ПК перевіряється відповідно до ЕД з врахуванням наявності подвійної, підсиленої і додаткової ізоляції.

При технічному обслуговуванні пристроїв ПК підлягає обов'язковій перевірці справність зовнішнього підключення ПК до мережі і підключених пристроїв. Проводи і кабелі не повинні мати пошкоджень ізоляції і захисної оболонки, обривів жил у місцях приєднання. В місцях введення у вхідні пристрої проводи і кабелі повинні бути закріплені, щоб не створювати натягу струмопровідних жил. З'єднувальні пристрої, зокрема вбудовані в ПК, повинні мати справні контакти, в з'єднувальних пристроях релейно-контактного типу контактний зазор у відключеному стані повинен бути не меншим 3 мм.

Для безпечної експлуатації ПК в приміщенні, де вона встановлена, повинні забезпечуватися кліматичні умови, встановлені ЕД.

Враховуючи велику щільність монтажу в пристроях ПК, при їх технічному обслуговуванні повинні забезпечуватися шляхи витоку, повітряні зазори і відстані по ізоляції в ланцюгах, пошкодження ізоляції яких може призвести до ураження електричним струмом. При кожному регламентованому технічному обслуговуванні шляху витоку в ланцюгах напругою вище 42 В повинні очищатися від пилу шляхом протирання спиртом або іншим нейтральним розчинником, а пошкоджені місця ізоляції повинні покриватися ізоляційним лаком. При заміні елементів в цих ланцюгах повинні витримуватися повітряні зазори між струмоведучими частинами і не допускатися гострі виступи припою і виводів елементів.

Вимоги електричної і механічної безпеки для ПК і систем обробки даних встановлені ГОСТ 25861 - 83. Додаткові або особливі заходи безпеки, яких необхідно дотримуватися при експлуатації і технічному обслуговуванні.

Категорично забороняється на пристроях ПК, що знаходяться під напругою:

* зняття і установка вентиляторів, блоків і вузлів;
* від'єднання і приєднання зовнішніх і внутрішніх роз'ємів;
* електромонтажні роботи по заміні електрорадіоелементів;
* заміна мережевих запобіжників.

У пристроях введення і виведення інформації ПК, а також в пристроях відображення інформації з високовольтними телевізійними трубками при їх роботі можуть створюватися і накопичуватися заряди статичної електрики, тому вони повинні експлуатуватися із застосуванням засобів захисту від статичної електрики, вказаних в ЕД. Ці засоби підлягають перевірці при регламентованому технічному обслуговуванні пристроїв.

Захисні огорожі (кожухи, сітки, бар'єри), що перегороджують доступ до рухомих частин, ланцюгів високої напруги, газорозрядних трубок високого тиску і т. д., повинні бути в справному стані, а дверці, що є на них, повинні надійно утримуватися замками в закритому стані. При необхідності, якщо це передбачено конструкцією пристрою, захисна огорожа повинна бути заземлена.

Особи, що допускаються до експлуатації і технічного обслуговування ПК, повинні пройти цільове навчання з вивчення правил роботи і вимог безпеки при роботі з ПК, а також ЕД на конкретні види ПК, до роботи з якими вони одержують допуск. До експлуатації ПК допускаються особи, що мають групу по електробезпеці не нижче II, до технічного обслуговування - групу III.

## 5.4 Пожежна безпека

Пожежа на підприємстві становить велику небезпеку для життя працюючих і може заподіяти величезний матеріальний збиток. За визначенням, пожежа – не контрольоване горіння поза спеціальним вогнищем, що наносить збиток.

Пожежна безпека може бути забезпечена мірами пожежної профілактики й активного пожежного захисту . Поняття пожежної профілактики включає комплекс заходів, необхідних для попередження виникнення пожежі чи зменшення його наслідків. Під активним пожежним захистом розуміються міри, що забезпечують успішну боротьбу з виникаючими чи пожежами вибухонебезпечною ситуацією.

Для боротьби з пожежами застосовують наступні способи:

* + ізолюють вогнище горіння від повітря чи подають до нього незапальні гази в такій кількості, щоб відносний зміст кисню виявився недостатнім для процесу горіння;
* прохолоджують вогнище горіння до температури нижче точок загоряння матеріалів, що знаходяться в небезпечній зоні;
* гальмують швидкість хімічної реакції в полум'ї;
* механічно зривають полум'я, впливаючи на нього сильними струменями газу, води чи порошку;
* створюють умови, при яких полум'я може поширюватися тільки через вузькі канали, зменшують силу полум'я і площу вогнища пожежі.

Приміщення, у якому розташована серверна, відноситься по пожежній безпеці до категорії В по НАПББ.ОЗ.002-2007 [9]і ступеню вогнестійкості II. Категорія В - Приміщення віднесено до категорії В, тому, що елементи комп’ютерної техніки при контакті з водою можуть вибухнути або загорітись.

II ступінь вогнестійкості - будинки з несучими та огороджувальними конструкціями з природних матеріалів або штучного каменю, бетону або залізобетону з застосуванням листових і плиткових негорючих матеріалів. У покритті будівлі застосовані незахищені стальні конструкції.

У даному приміщені категорії В не існує великого ризику виникнення пожежі. Також приміщення оснащено автоматичною систему пожежогасіння та порошковими вогнегасниками типу ОП-5Б. Отже, фактично приміщення відповідає установленим нормам пожежної безпеки.

# ВИСНОВКИ

Під час роботи над дипломним проектом було проведено розробку програмного забезпечення у вигляді веб-ресурсу формування рейтингу професійних фотографів та їх робіт засобами веб розробки.

Підчас розробки проекту було проведено аналіз вже існуючих аналогічних програмних рішень. Розроблено прототип та зовнішнього вигляду, функціоналу та структуру БД. Підібрано найоптимальніші засоби та інструменти веб-розробки за допомогою яких і розроблявся проект. Було проведено налаштування робочого місця для роботи над проектом: проведено налаштування всіх програмних засобів і серверів, без яких неможлива б була подальша робота над проектом. Основною частиною підчас розробки була частина з написання програмного коду та програмування функціоналу веб-ресурсу. Також по завершенню написання коду було проведено тестування програмного засобу та кінцеве виправлення виникнувши помилок. По завершенню написання програмного продукту було оформлено звіт до дипломного проекту в якому додатково було обраховано економічна доцільність розробки та використання розробленого програмного забезпечення. Також звіт диплому містить розділ з охорони праці в якому наведені дані про оптимальні робочі умови програміста та пораховано штучне освітлення для приміщення в якому проводилася розробка програмного продукту.

Отже, завдання дипломного проектування виконано в повному обсязі, результатом чого є готовий веб сайт, на якому відображаються фотороботи конкретного фотографа згідно його рейтингового місця серед інших фото робіт.