

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК
СЕКЦІЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ПРОЕКТУВАННЯ

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему: «Web-додаток організації діяльності івент-агентства»

за спеціальністю 122 «Комп'ютерні науки»,
освітньо-професійна програма «Інформаційні технології проектування»

Виконавець роботи: студент групи ІТ-61-8 Андрусішин Іван Костянтинівич

Кваліфікаційна робота бакалавра
захищена на засіданні ЕК
з оцінкою _____ «___»
2020 р.

Науковий керівник

(підпис)

к.т.н., Антипенко В.П.

(науковий ступінь, вчене звання, прізвище та ініціали)

Голова комісії

(підпис)

Шифрін Д. М.

(науковий ступінь, вчене звання, прізвище та ініціали)

Засвідчую, що у цій дипломній роботі немає
запозичень з праць інших авторів
без відповідних посилань.

Студент _____
(підпис)

Суми-2020

Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра комп'ютерних наук
Секція інформаційних технологій проектування
Спеціальність 122 «Комп'ютерні науки»
Освітньо-професійна програма «Інформаційні технології проектування»

ЗАТВЕРДЖУЮ

Зав. секцією ІТП

_____ В. В. Шендрик
«__» _____ 2020 р.

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА СТУДЕНТУ

Андрусишин Іван Костянтинович

1 Тема роботи Web-додаток організації діяльності івент-агентства

керівник роботи Антипенко Вікторія Петрівна, к.т.н. _____,

затверджені наказом по університету від «14» травня 2020 р. № 0576-III

2 Строк подання студентом роботи «1» червня 2020 р.

3 Вхідні дані до роботи технічне завдання на розробку web-додатку організації діяльності івент-агентства

4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) аналіз предметної області, проектування web-додатку, розробка web-додатку

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) _____

6. Консультанти розділів роботи:

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання 01.10.2019

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Оформлення планування робіт	15.03.2020- 25.03.2020	
2	Оформлення технічного завдання	25.03.2020- 10.04.2020	
3	Проведення аналізу предметної області	10.04.2020- 20.04.2020	
4	Проведення проектування web-додатку	20.04.2020- 30.04.2020	
5	Розробка web-додатку	01.05.2020- 23.05.2020	
6	Тестування web-додатку	23.05.2020- 25.05.2020	
7	Завантаження web-додатку на хостинг	25.05.2020- 30.05.2020	
8	Оформлення пояснювальної записки	15.03.2020- 31.05.2020	

Студент

(підпис)

Андрусишин І.К.

Керівник роботи

(підпис)

к.т.н., Антипенко В.П.

РЕФЕРАТ

Тема кваліфікаційної роботи бакалавра «Web-додаток організації діяльності івент-агентства».

Пояснювальна записка складається зі вступу, 3 розділів, висновків, списку використаних джерел із 18 найменувань, 4 додатків.

Кваліфікаційну роботу бакалавра присвячено розробці web-додатку організації діяльності івент-агентства.

У першому розділі проведено огляд останніх досліджень за тематикою роботи та проаналізовано аналоги розроблюваного web-додатку, визначено їх переваги та недоліки. Також визначено мету та задачі проекту.

У другому розділі проведено структурно-функціональне моделювання, визначено варіанти використання web-додатку та спроектовано базу даних. У результаті були змодельовані такі діаграми як: контекстна діаграма, діаграма декомпозиції, діаграма варіантів використання та ER-діаграма.

У третьому розділі описано розробку web-додатку, яка є результатом проектування. Наведено архітектуру web-додатку. Також було описано його реалізацію з прикладами роботи.

Ключові слова: WEB-ДОДАТОК, БАЗА ДАНИХ, JAVASCRIPT, PHP, РОЗРОБКА.

ЗМІСТ

ВСТУП	6
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	8
1.1 Огляд останніх публікацій.....	8
1.2 Аналіз існуючих продуктів-аналогів.....	10
1.3 Постановка задачі.....	15
2 ПРОЕКТУВАННЯ WEB-ДОДАТКУ	17
2.1 Структурно-функціональне моделювання	17
2.2 Моделювання варіантів використання.....	19
2.3 Проектування бази даних	21
3 РОЗРОБКА WEB-ДОДАТКУ	25
3.1 Архітектура web-додатку	25
3.2 Програмна реалізація web-додатку організації діяльності івенту-агенства.....	26
ВИСНОВОК.....	34
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	35
Додаток А.....	37
Додаток Б	47
Додаток В.....	55
Додаток Г	56

ВСТУП

Свято з найдавніших часів є невід’ємним елементом культури. Воно завжди було значущою подією для суспільства та кожного індивіда як унікальна форма емоційно-символічного вираження (затвердження) їх ціннісно-світоглядних установок.

У сучасному суспільстві святкування різних подій відіграє важливу роль, оскільки на даний момент існує велика кількість додаткових послуг, які кожен бажає отримати на власний захід. Наприклад, це замовлення фотографа, ведучого, аніматора, декоратора тощо. Але сучасна людина є досить зайнятою сьогодні, і, на жаль, не має достатньо часу для організації таких подій власноруч. Також іноді має місце відсутність певних навичок. Це може бути недостатній досвід в організації масових заходів, страх відповідальності тощо. Тому, логічним у наш час стала поява івент-агентств, які вирішують вищеописані проблеми сучасного ініціатора (замовника).

Сьогодні, для кожного учаснику ринку мати власний web-сайт стало просто необхідно. В якості двох основних передумов, що призвели до такого стану справ, можна назвати бурхливий розвиток web-технологій і зростання добробуту суспільства. Звичайно бізнес не міг не відреагувати на ці зміни. Тому, зараз значна кількість підприємців перенесла частину власного «офлайнового» бізнесу в мережу Інтернет. Так, у відносно короткий термін, кількість різного роду комерційних web-сайтів значно зросла.

Метою даної роботи є розробка web-додатку організації діяльності івент-агентства. Об’єктом роботи є організація діяльності івент-агентства. Предметом дослідження є методи та технології організації діяльності івент-агентства.

Для досягнення мети необхідно виконати наступні задачі:

- провести аналіз предметної області, визначити актуальність продукту;

- ознайомитись з аналогами web-сайтів із даної тематики, виділити їх переваги та недоліки;
- обрати web-технології для розробки даного web-додатку;
- розробити модель та структуру web-ресурсу;
- реалізувати дану структуру у вигляді web-сайту;
- розробити функціонал для організації діяльності івент-агентства;
- провести тестування web-додатку.

Практичне значення розроблюваного додатку полягає в скороченні часу користувача під час оформлення замовлення на організацію власного івенту.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд останніх публікацій

Поняття «сайт» або «web-сайт» зараз стали досить поширеними в мережі Інтернет і щільно увійшли в лексикон сучасної людини.

Сайт або web-сайт [1] – це web-сторінка або група web-сторінок, яка об'єднується єдиною темою, автором, організацією; web-сторінки, логічно пов'язані між собою. Зазвичай сайт в мережі Інтернеті – це пов'язані дані, які мають власну унікальну адресу та сприймаються користувачем як єдине ціле. Web-сайти називаються так, тому що доступ до них відбувається по протоколу НТТР. Web-сайт може належати приватній особі або організації і бути доступним в мережі Інтернет під загальною ІР-адресою, доменним ім'ям або локально на одному комп'ютері. Сайт може містити одну або кілька сторінок. Наприклад, сайт компанії «Apple» містить кілька тисяч сторінок. На сайтах часто виділяють головну сторінку, по якій можна зрозуміти призначення сайту. Вона знаходиться першою на сайті. Усі сайти в сукупності складають Всесвітню павутину. Кожна сторінка може містити картинки, фотографії, текст, відео- або аудіо-файли. Також можна спостерігати так звані переходи на інші сторінки – посилання. Це особливі об'єкти, натиснувши на які, користувач автоматично переходить на іншу сторінку. Часто вони бувають виділені (підсвічені іншим кольором, підкреслені, можуть виступати в якості картинок). Зазвичай сторінки взаємопов'язані між собою. І відвідувач сайту сам вирішує на яке посилання йому переходити в залежності від потрібної йому інформації. У Всесвітній павутині перегляд сторінок сайтів здійснюється за допомогою браузерів.

Мережа Інтернет багато в чому розширює можливості користувача, цей процес буде постійно розвиватися. Організація, яка має намір поширення

інформації про власну діяльність та продаж власних товарів через мережу Інтернет, може створити власний web-сайт, який, як правило, складається з декількох переходів по посиланнях і представлена інформація про компанію й товари, які вона пропонує [2].

Споживач може переглянути зображення товару, його деталі, матеріали, розміри тощо. При створенні сайту потрібно визначити його призначення. Відповідь на це питання повинен дати не тільки розробник, а й користувач цього сайту. У першу чергу мета, яку переслідують всі організації – це отримання прибутку за рахунок розповсюдження реклами власних послуг. Для досягнення цих цілей компанії використовують різні бізнес інструменти.

У кожного сайту організації є ряд наступних завдань:

- сайт забезпечує імідж компанії; користувач розуміє, що вона використовує сучасні технології;
- на сайті представлена вичерпна інформація про організацію, її товари, послуги, ділові інтереси, клієнтів, що дає замовнику впевненість у професійності компанії;
- сайт – це найоптимальніший, дешевий і масовий спосіб реклами;
- сайт підвищує ймовірність того, що компанія стане більш відомою; це може допомогти в пошуку нових замовників і партнерів по бізнесу;
- якщо в організації використовуються сервісні або консультаційні послуги, то вони можуть допомогти користувачам визначитися з товаром і проконсультувати користувача по функціоналу сайту; також доступною є можливість виконувати послуги для територіально віддалених замовників.

Із вищеописаного зрозуміла актуальність використання сайтів для просування сучасного бізнесу. Тому створення web-додатку для організації діяльності івент-агентства не є виключенням. Реклама в мережі Інтернет приверне нових клієнтів, а розташована інформація на сайті надасть відповіді на запити користувачів. Також останні зможуть звернутися до івенту-

агенства та оформити замовлення на проведення власного заходу, використовуючи функціонал розробленого web-додатку.

1.2 Аналіз існуючих продуктів-аналогів

Розроблюваний web-сайт повинен мати попит на нинішньому ринку. Отже, він має бути конкурентно-спроможним і створюватися згідно сучасним тенденціям. Найкращим способом визначення ключових майбутніх переваг розробки є метод порівняння: визначення основних переваг та недоліків існуючих аналогів web-ресурсів івент-агентств [2].

В якості дослідження було розглянуто web-сайти таких івент-агентств, як Шаленій, Соль та VIVAT!.

На головній сторінці web-сайту івент-агентства «Шаленій» (рис. 1.1) розміщені загальний опис свят, портфолію з організованими агентством івентами, переваги над конкурентами, інформація про ключових робітників, перелік послуг та постійних клієнтів, декілька відгуків і інформація про агентство та його діяльність [3].

Більшість інформації, яка необхідна користувачу, знаходиться на головній сторінці. Окрім цього, якщо клієнт бажає дізнатися більше про організацію івенту, є окрема сторінка з данною інформацією (рис. 1.2).

Дизайн web-сайту можна охарактеризувати як сучасний з великою кількістю різних стилів та способів взаємодії з його відвідувачем. Багато святкових зображень та фотографій в поєднанні з яскравою гаммою направленні на підвищення настрою користувача, щоб йому було комфортніше знаходитись на сайті та у подальшому скористатися послугами цього івент-агентства.



Рисунок 1.1 – Головна сторінка web-сайту «Шаленій»

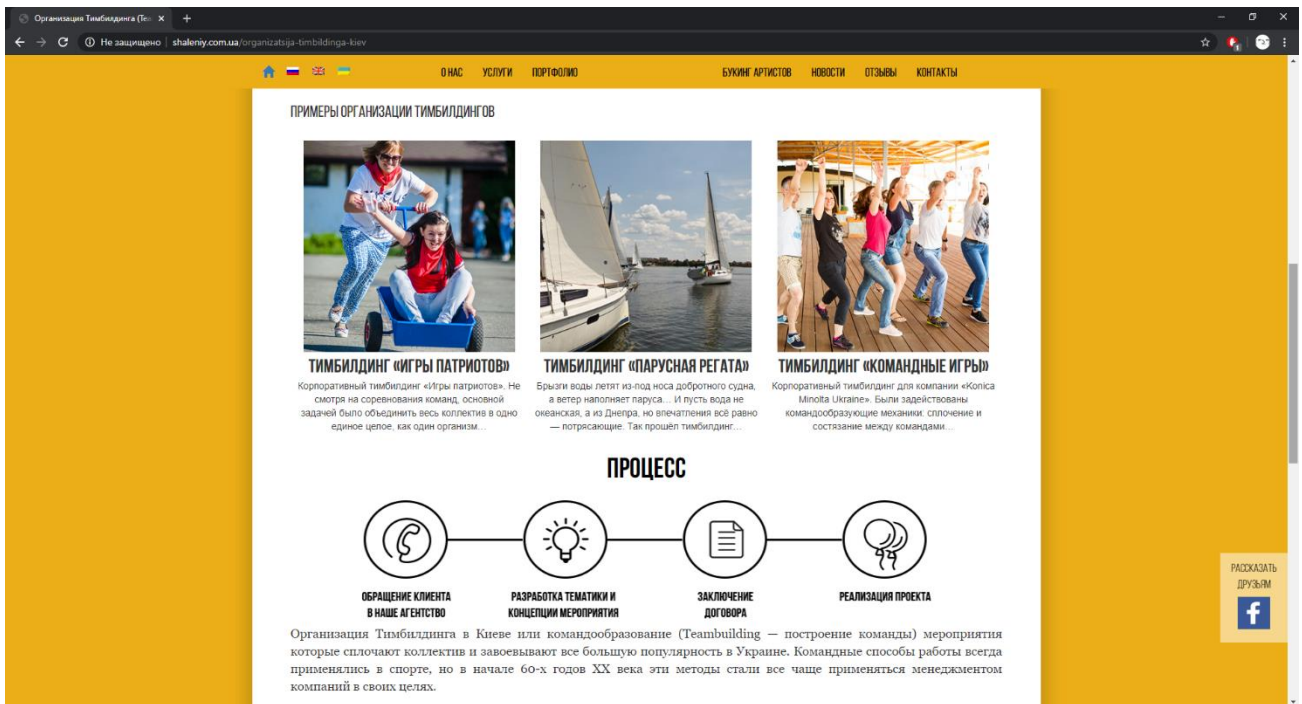


Рисунок 1.2 – Деталі організації івенту

Замовити організацію свята можна лише звернувшись за контактними даними, що знаходяться на головній сторінці чи на сторінці «Контакти»

(рис. 1.3). Окрім цього, користувач може написати відгук, який збережеться на сайті на сторінці «Отзывы».

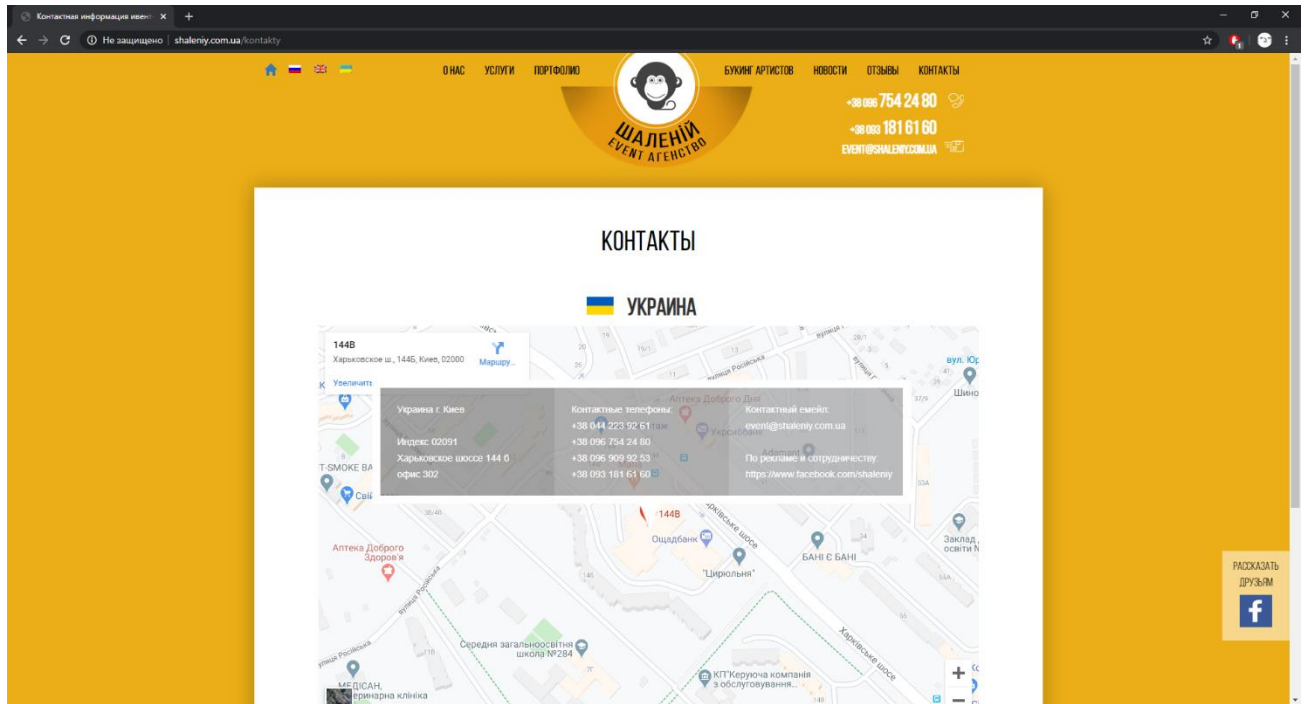


Рисунок 1.3 – Сторінка з контактами

Web-сайт івент-агентства «Соль» [4] (рис.1.4) є односторінковим. На ньому знаходиться інформація про послуги, які може надавати агентство, переваги над конкурентами, команду та виконані проекти компанії, партнерів, контактна інформація та інформація про агентство.

На web-сайті відображається досить мало текстової інформації, і користувачу не надається інформація про майбутнє свято та деталі його організації. (рис. 1.5).

Дизайн web-сайту є мінімалістичним із чорним текстом на білому фоні. Також на ньому знаходиться велика кількість анімованих ефектів, що надає ілюзію життя на web-сайті.



Рисунок 1.4 – Головна сторінка сайту «Соль»

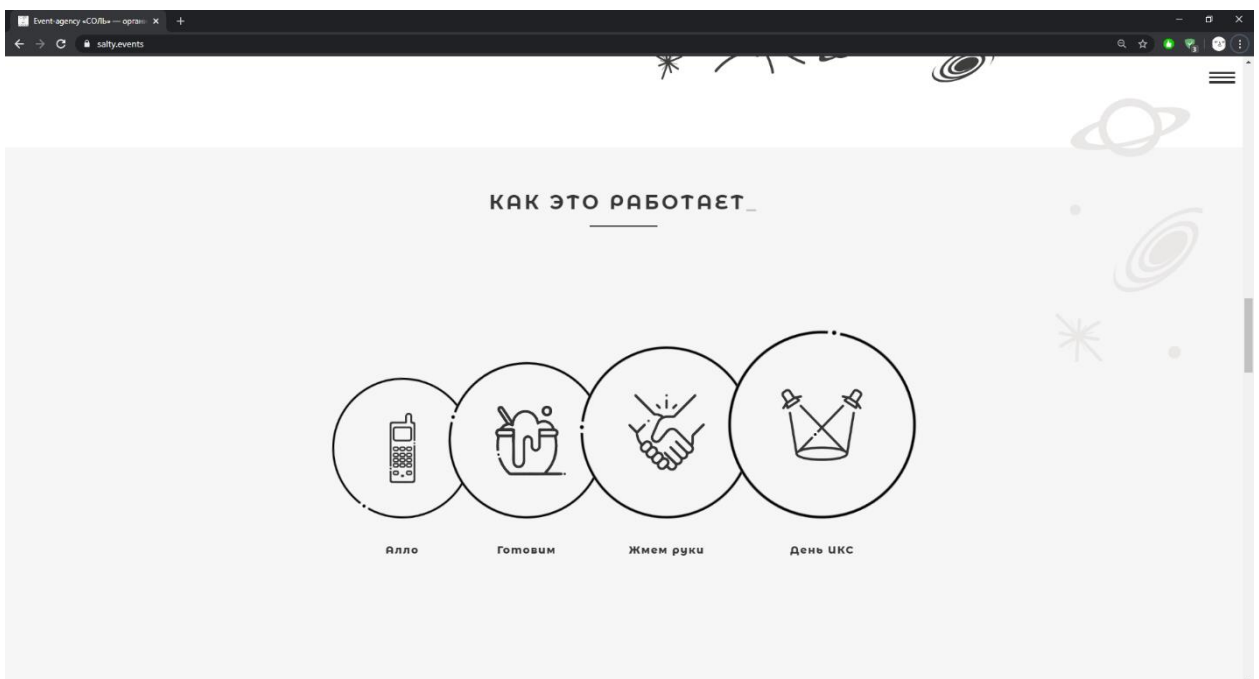


Рисунок 1.5 – Деталі організації івенту

Користувач має можливість замовити дзвінок чи написати листа за допомогою форми, яка відображається по натисканні на відповідну кнопку, що розташована внизу сторінки (рис. 1.6).

Рисунок 1.6 – Форма зворотнього зв'язку

Web-сайт івент-агентства «VIVAT!» (рис. 1.7) є регіональним представником у місті Суми, і, отже, є прямим конкурентом івент-агентства, для якого розробляється сайт. На його головній сторінці знаходиться досить стисла інформація про компанію та контактна інформація.

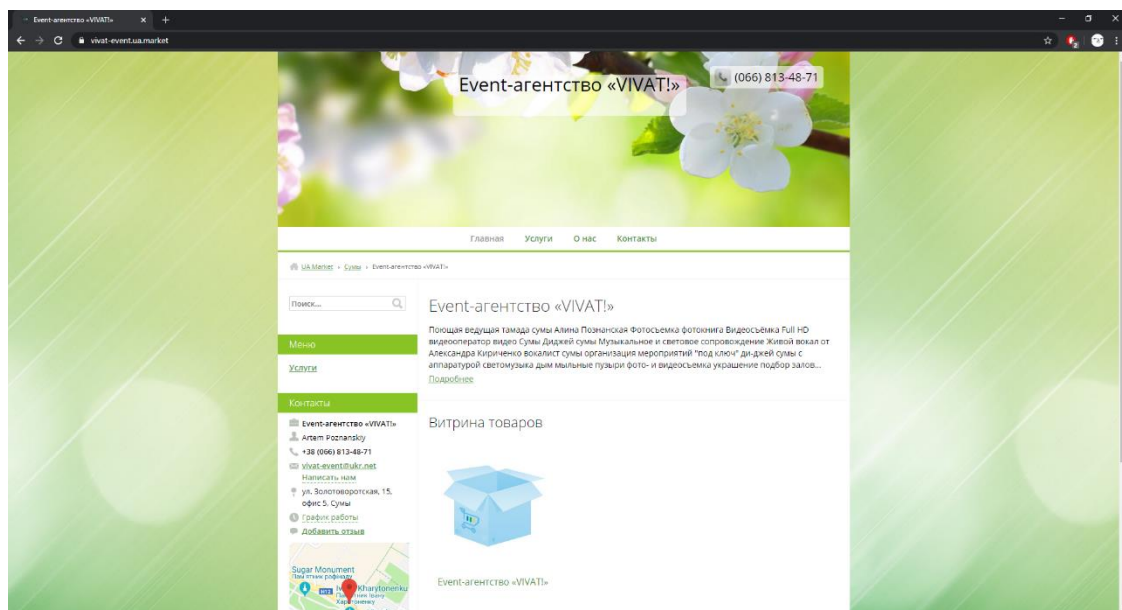


Рисунок 1.7 – Головна сторінка сайту «VIVAT!»

Інші сторінки сайту надають дещо повторювану інформацію.

Дизайн web-сайту можна назвати застарілим, так як він оформлений в одному простому стилі та на сайті немає ніяких анімованих елементів.

Користувач немає можливості взаємодіти із web-сайтом. Як висновок, відвідувач, перейшовши на нього, не знайде інформації про агентство, його діяльність. Побачивши оформлення web-сайту є вираження його швидкого закриття без ймовірності повернення. І таким чином, агентство може втратити потенційного клієнта.

Під час процесу порівняння аналогів була створена порівняльна таблиця (табл. 1.1), яка показує переваги, які можна використати під час розробки практичної частини даної роботи, та недоліки, які варто усунути.

Таблиця 1.1 – Порівняльна таблиця характеристик сайтів-аналогів

Характеристика	shaleny.com.ua	salty.events	vivat-event.ua.market
Зручний інтерфейс	+	+	-
Інтерактивність	+	+	-
Сучасний дизайн	-	+	-
Навігація	+	-	-
Наявність відгуків	+	-	-
Поле авторизації	-	-	-
Можливість замовлення онлайн	-	-	-

1.3 Постановка задачі

Метою даного проекту є розробка web-додатку для організації діяльності івент-агентства. Даний продукт буде використовуватися для

поширення інформації про агентство та його послуги в мережі Інтернет, а також для скорочення часу на оформлення замовлення щодо організації проведення івенту його робітником.

Розроблений web-додаток виконуватиме наступні функції:

- надавання інформації про агентство, його персонал та послуги з можливістю редагування контенту адміністратором;
- можливість оформлення замовлення на організацію івенту через даний web-додаток;
- збереження списку замовлень.

Для досягнення мети проекту необхідно виконати наступні задачі:

- провести аналіз предметної області, визначити актуальність продукту;
- ознайомитись з аналогами web-сайтів із даної тематики, виділити їх переваги та недоліки;
- обрати web-технології для розробки даного web-додатку;
- розробити модель та структуру web-ресурсу;
- реалізувати дану структуру у вигляді web-сайту;
- розробити функціонал для організації діяльності івент-агентства;
- провести тестування web-додатку.

Більш детальний опис задач описаний в технічному завданні (додаток А).

Web-додаток буде реалізовано, використовуючи такі web-технології, як: HTML5 для написання розмітки web-сторінок, CSS3 для опису зовнішнього вигляду web-сторінок, JavaScript для написання сценаріїв web-сторінок, щоб надати їм інтерактивності та PHP5 для написання скриптів, які виконуються на сервері. Для написання коду буде використано багатofункціональний текстовий редактор Notepad++.

2 ПРОЕКТУВАННЯ WEB-ДОДАТКУ

2.1 Структурно-функціональне моделювання

Для моделювання структури та функціоналу певного процесу потрібно використати методи на основі об'єктно-орієнтованого та структурного підходу до моделювання. Функціональне моделювання SADT (IDEF0) є основним методом моделювання цих процесів [5].

Мета структурно-функціонального моделювання – це за певними правилами та процедурами декомпонувати основний процес на більш менші частини.

IDEF0 – модель [6,7] показує специфікацію усіх дій, що відбуваються у процесі та їх взаємозв'язок між собою. Контекстну діаграму головного процесу web-додатку організації діяльності івент-агентства показано на рисунку 2.1.

Перелік головних даних даної діаграми наступний:

- вхідні дані: оформлення замовлення, нові дані;
- вихідні дані: програма організації, змінена інформація;
- управління: існуючі програми;
- механізми: користувач, адміністратор, редактор, база даних.

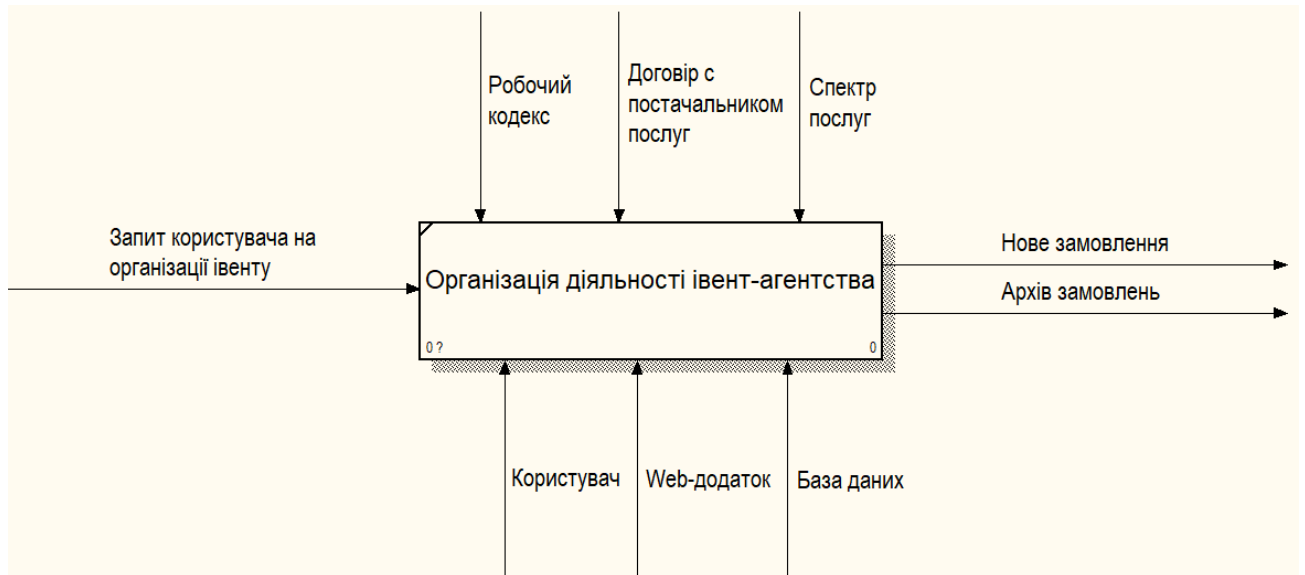


Рисунок 2.1 – Контекстна діаграма

Для відображення внутрішніх процесів контекстної діаграми, вона декомпозується на діаграми нижчого рівня. Усі дуги та зв'язки з батьківської діаграми переносяться на діаграму декомпозиції [8] (рис. 2.2). Головна батьківська діаграма декомпозується на наступні блоки:

- перехід між сторінками;
- авторизація;
- зміна інформації;
- редагування замовлення;
- замовлення свята.

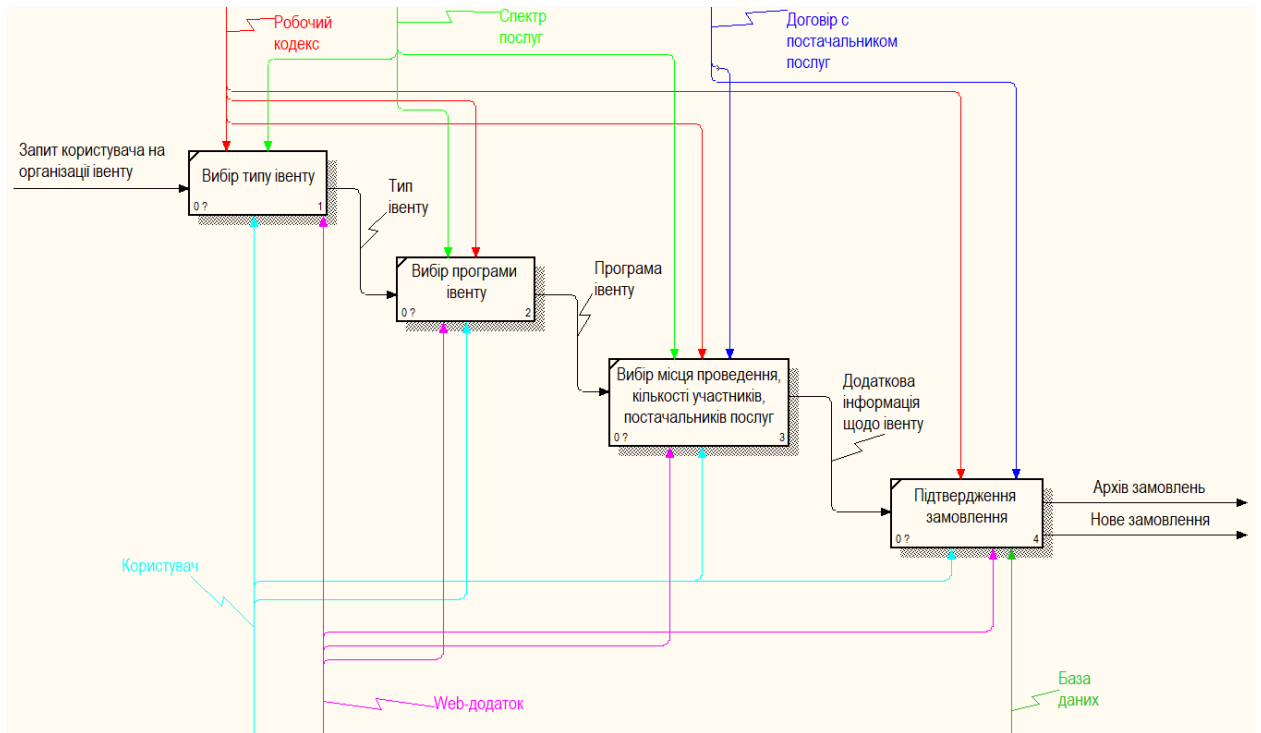


Рисунок 2.2 – Діаграма декомпозиції

Вхідними даними до діяльності «перехід між сторінками» є дані для проведення івенту, а для «авторизації» та «редагування замовлення» – нові дані. Стрілками управління для діяльностей «редагування замовлення» та «замовлення свята» є «існуючі програми». Стрілками механізмів для діяльності «перехід між сторінками» є «користувач», «редактор» та «адміністратор», для «авторизації» та «редагування замовлення» – «редактор», «адміністратор» та «база даних»; для «зміни інформації» – «адміністратор» та «база даних», а для «замовлення свята» – «користувач», «редактор», «адміністратор» та «база даних».

2.2 Моделювання варіантів використання

Для опису дій, які може виконувати інформаційна система була створена діаграма варіантів використання. Діаграма варіантів використання [9] є готовим концептуальним поданням системи під час її

проектування та розробки. Дана діаграма складається з акторів, варіантів використання і відносин між ними. При побудові діаграми можуть використовуватися також загальні елементи нотації: примітки і механізми розширення.

Суть даної діаграми складається в тому, що за допомогою варіантів використання проєктована система представляється у вигляді безлічі акторів, які взаємодіють з системою [10]. При цьому актором називається будь-який об'єкт, суб'єкт або система, що взаємодіє з моделюється системою ззовні. А варіантом використання називають специфікацію сервісів (функцій), які актор отримує від системи. Інакше кажучи, кожен варіант використання визначає деякий набір дій, що здійснюються системою при взаємодії з актором. При цьому в моделі ніяк не відображається спосіб реалізації цього набору дій [11].

Діаграма варіантів використання web-додатку організації діяльності івент-агентства представлена на рисунку 2.3.

Варіанти використання web-додатку є наступними:

- перехід між сторінками;
- замовлення свята;
- редагування замовлення;
- зміна інформації;
- авторизація.

Акторами на даній діаграмі виступають користувач, редактор та адміністратор. Елементом зовнішнього зв'язку, до якого додаток звертається при потребі актора в діаграмі виступає «База даних».

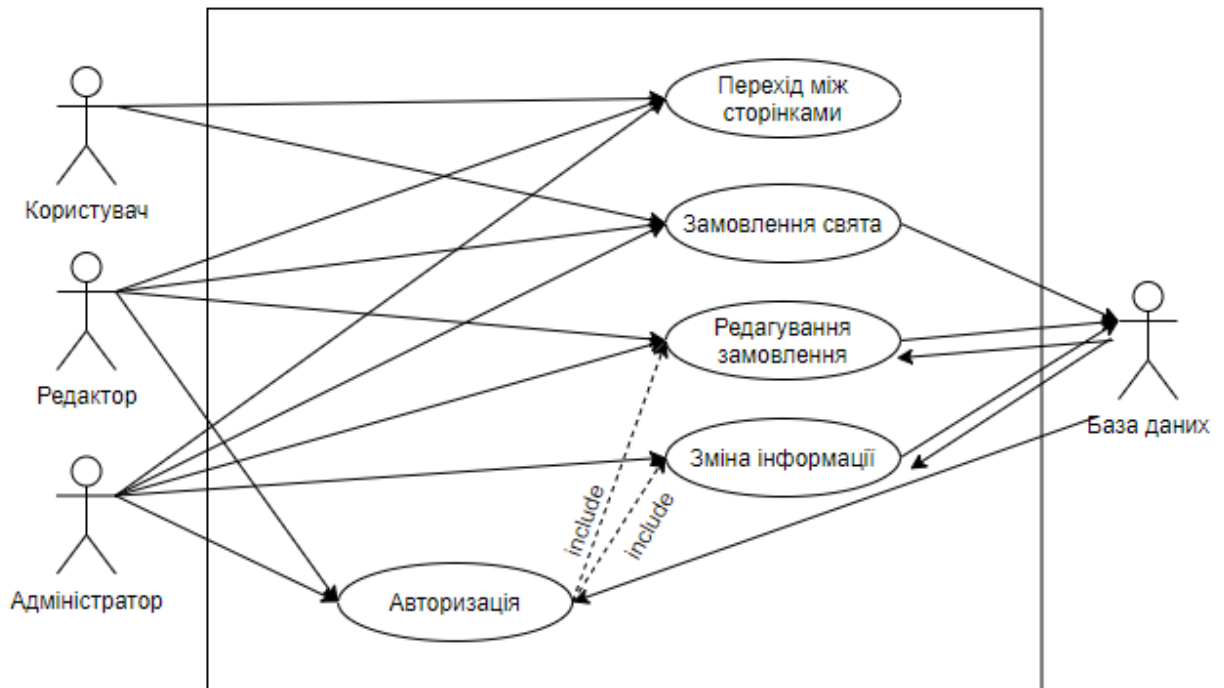


Рисунок 2.3 – Діаграма варіантів використання

2.3 Проектування бази даних

На сьогоднішній день жодна складна система не розробляється напряму. В першу чергу, вона проектується. База даних, що є однією з найважливіших частин інформаційної системи, звичайно ж, є складним об'єктом, який також підлягає проектуванню [12].

База даних являє собою подобу електронної картотеки, тобто сховище або контейнер для деякого набору файлів даних, занесених в комп'ютер. Таким чином, в інформаційній системі база даних відповідає за процес зберігання інформації, при цьому пошук цієї інформації, відповідно, система здійснює в цій базі даних. Мета проектування бази даних – це правильне відображення обраної для автоматизації предметної області [13].

Під час проектування бази даних відбувається виділення сутностей. Сутність [14] являє тип об'єктів, які повинні зберігатися в базі даних. Кожна таблиця в базі даних повинна представляти одну сутність. Як правило, сутності відповідають об'єктам з реального світу. У кожній сутності визначають набір атрибутів. Атрибут представляє властивість, яке описує деяку характеристику об'єкта. Кожен стовпець повинен зберігати один атрибут сутності. А кожен рядок представляє окремий об'єкт або екземпляр сутності [15].

У процесі проектування бази даних було виділено наступні сутності:

- робітники (workers);
- замовлення (orders);
- архів (arhive);
- інформація (info);

На рисунку 2.4 приведені представлення моделі бази даних додатку з атрибутами сутностей та зв'язками між таблицями.

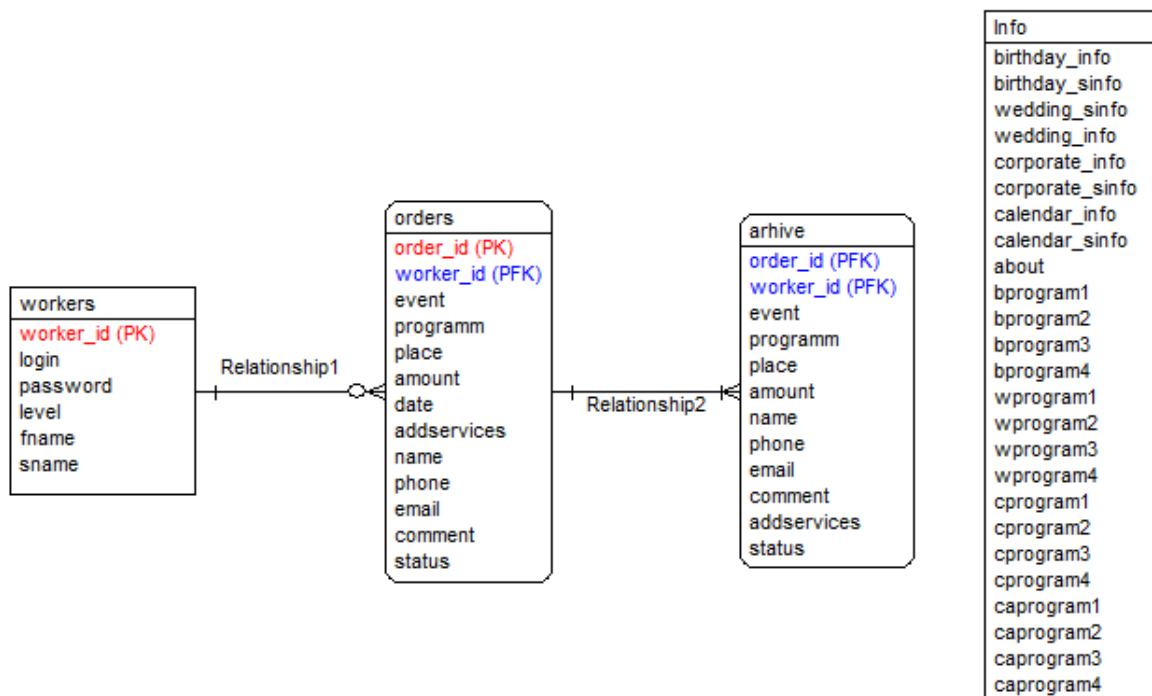


Рисунок 2.4 – Представлення моделі бази даних додатку

Наступним етапом після виділення сутностей є реалізація структури бази даних. Результат реалізації наведений в таблиці 2.1.

Таблиця 2.1 – Структура бази даних розроблюваного web-додатку івент-агенства

Таблиця	Поле	Зміст	Тип	Ключі	Обмеження
workers	worker_id	Номер робітника	integer	PK	not null
	login	Логін	varchar(15)		not null
	password	Пароль	varchar(15)		not null
	level	Рівень доступу	int		not null
	fname	Ім'я робітника	varchar(20)		not null
	lname	Прізвище робітника	varchar(25)		not null
orders	order_id	Номер замовлення	integer	PK	not null
	worker_id	Номер робітника	integer	PFK	not null
	event	Івент	varchar(25)		not null
	program	Програма івенту	varchar(30)		not null
	place	Місце проведення	varchar(25)		not null
	amount	Кількість учасників	varchar(5)		not null

Продовження таблиці 2.1 – Структура бази даних розроблюваного web-додатку івент-агенства

orders	date	Дата	date		not null
	addservices	Додаткові послуги	varchar(50)		
	name	Ім'я	varchar(40)		not null
	phone	Номер телефону	varchar(15)		not null
	email	Електронна пошта	varchar(25)		not null
	comment	Коментар	text		
	status	Статус	varchar(20)		not null

3 РОЗРОБКА WEB-ДОДАТКУ

3.1 Архітектура web-додатку

Перед початком розробки додатку, спочатку потрібно спроектувати його архітектуру.

Архітектура додатку [16] (рис. 3.1) представляє програмну частину або структуру його сторінок.

Архітектура web-додатку організації діяльності івент-агентства містить в собі базу даних, в якій зберігається інформація, що відображена на сайті, а також список замовлень, клієнтську частину, через яку користувач взаємодіє з додатком використовуючи графічний інтерфейс браузера, надсилає форму та отримує результати, серверну частину, яка відповідальна за зв'язок клієнтської частини з базою даних.

На рисунку 3.1 зображена схема архітектури додатку.

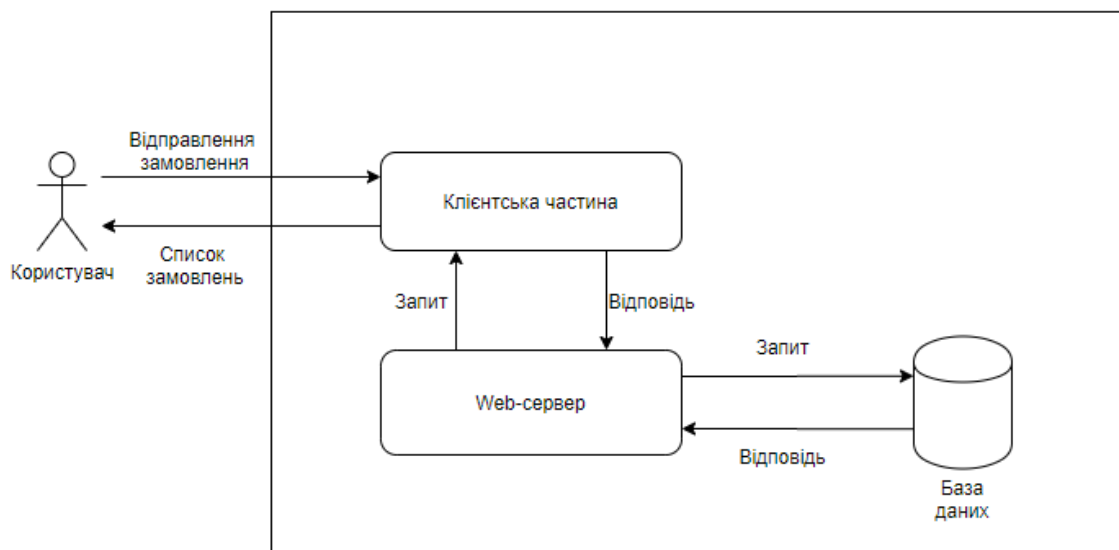


Рисунок 3.1 – Архітектура розроблюваного web-додатку

3.2 Програмна реалізація web-додатку організації діяльності івенту-агенства

Згідно з шаблону головної сторінки з технічного завдання була створена головна сторінка сайту (рис. 3.2).

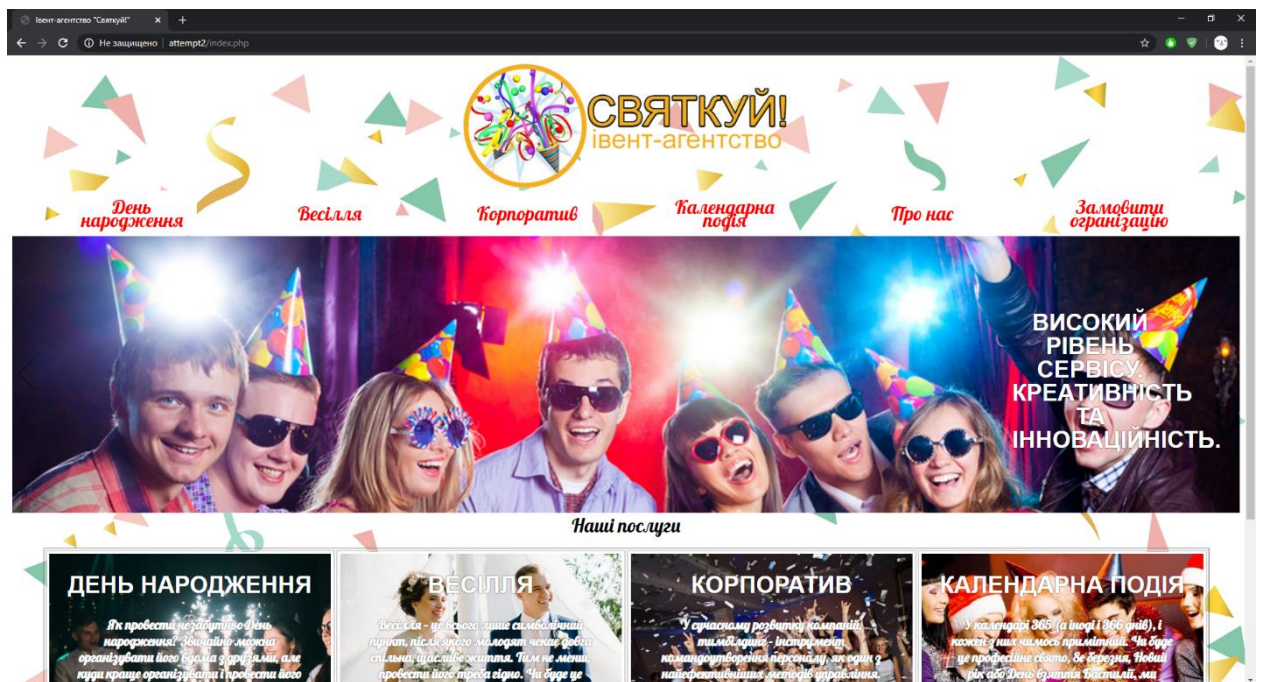


Рисунок 3.2 – Головна сторінка сайту

На сторінці знаходяться логотип івент-агентства, по натисканню на який відбувається перехід на головну сторінку, меню для переходу між сторінками, слайдер, який змінює фонову картинку та текстову інформацію на ній про переваги агентства, та таблиця з інформацією щодо послуг, що надає компанія. Лістинг знаходиться в додатку Г.

Сторінка «Весілля», що зображена на рисунку 3.3, надає коротку інформацію про саму подію. Для того, щоб дізнатися опис однієї з програм організації, користувач натискає на відповідну картинку, після чого з'являється текст з відповідною інформацією. Таким способом оформлені всі сторінки з інформацією про івенти.



Рисунок 3.3 – Сторінка «Весілля»

На сторінці замовлення (рис. 3.4) користувач поступово обирає параметри замовлення - тип івенту, програму його проведення, приблизну кількість учасників, бажане місце проведення, додаткові послуги та дату проведення івенту. Після цього користувач вводить свої контактні дані та підтверджує замовлення (рис. 3.5).

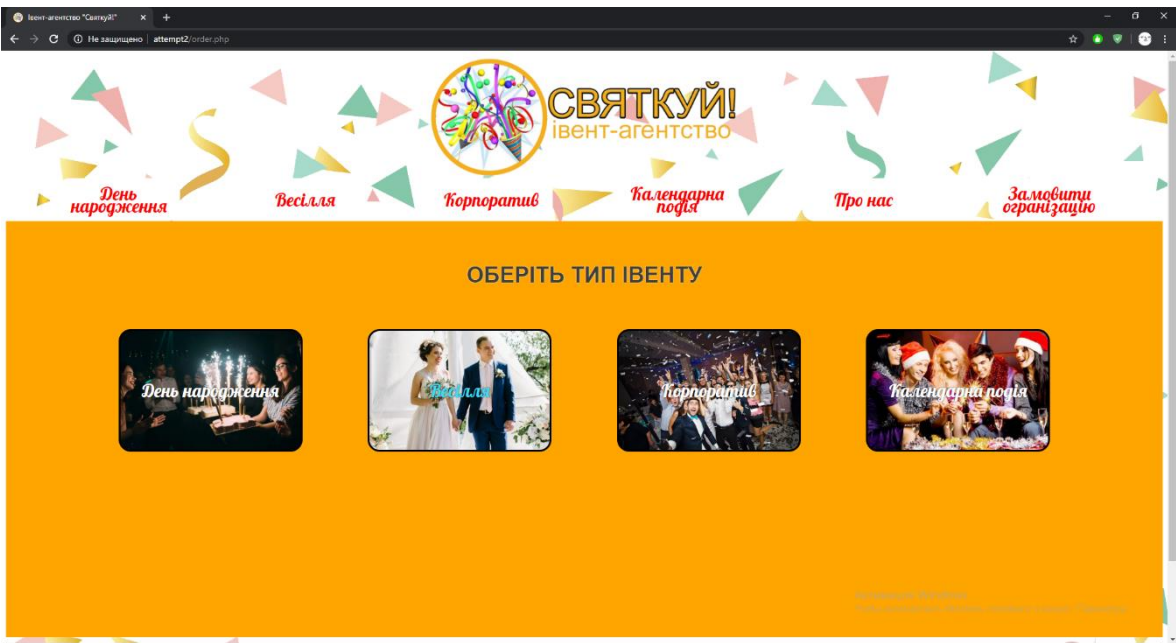


Рисунок 3.4 – Сторінка замовлення

←Повернутися

Оберіть дату івенту
30.06.2020

Оберіть час
17:00

Тип івенту:
День народження

Програма проведення:
Ювілей

Місце проведення:
Ресторан Ювілейний

Кількість учасників
≤50

Додаткові послуги
Фотограф, Ведучий

Введіть ваше ім'я

Введіть номер телефону

Введіть електронну пошту

Введіть коментар до замовлення(за бажанням)

Підтвердити

Рисунок 3.5 – Підтвердження замовлення

Для робітників агентства доступна авторизація (рис. 3.6), форма якої знаходиться внизу сторінки. Після успішної авторизації робітника перенаправить на сторінку в залежності з його рівнем доступу: редактор може редагувати нові замовлення та переглядати архів замовлень (рис. 3.7), а адміністратор може виконувати ті ж дії, що і редактор але й має можливість додавати користувачів із правами Редактора та редагувати матеріали розділів (рис.3.8).

Логін:

Пароль:

Вхід

Рисунок 3.6 – Форма авторизації

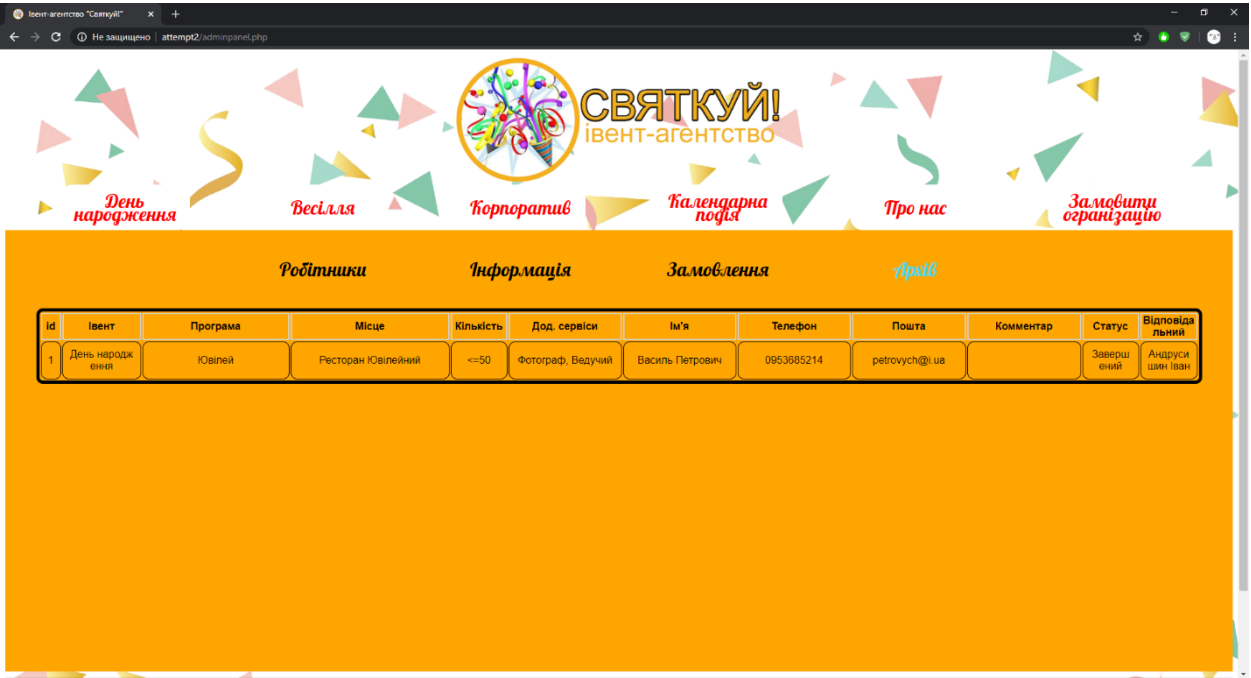


Рисунок 3.7 – Архів замовлень

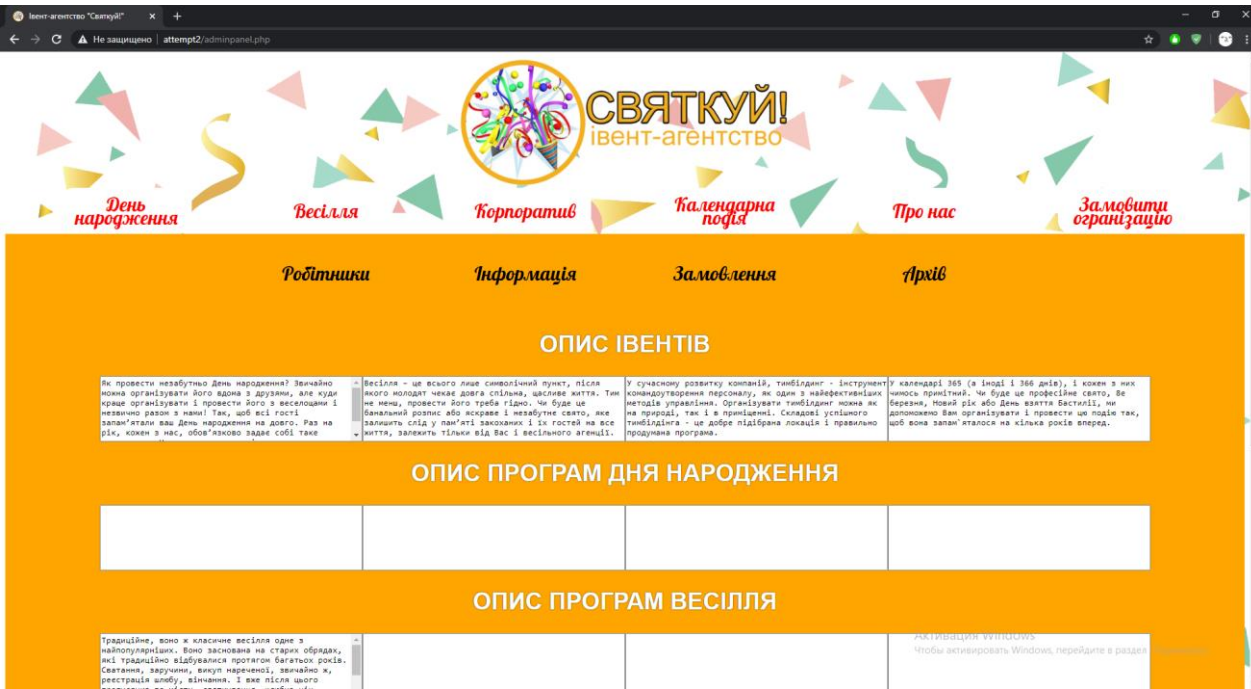


Рисунок 3.8 – Редагування інформації на сайті

Сторінки web-додатку івенту-агенства створювались за допомогою блочної верстки сайту. Кожен блок має власні властивості (рис. 3.7). Останні для кожного блоку були описані за допомогою каскадної таблиці стилів

(CSS), яка також надає можливість визначити кольори, шрифти, верстку та інші аспекти вигляду сторінки.

```
#howeare, #what{
    font-family: 'Lobster', cursive;
    font-size:30px;
    border-radius:20px;
    float: center;
    width: 250px;
    height: 70px;
    position: relative;
    vertical-align: middle;
    text-align: center;
    margin:0 auto;
}
```

Рисунок 3.7 – Властивості блоків

У прикладі (рис. 3.7) описується позиціонування, розміри блоку, шрифт тексту та його властивості.

Окрім цього, за допомогою CSS [17] можливо написати інтерактивні елементи, тобто, властивості будуть змінюватись під час або після певної взаємодії з тим чи іншим об'єктом. Наприклад, при наведенні на кнопку вона буде змінювати власний колір (рис. 3.8).



Рисунок 3.8 – Властивості посилань

Тим самим сторінка з динамічними ефектами не є статичною і привертає увагу користувачів, не даючи засумувати.

Також одним із прикладів роботи таблиць стилів є слайдер (рис. 3.9).



Рисунок 3.9 – Слайдер зображень

Суть слайдеру заключається в блоці, що розташований на сторінці, який по чергово змінюється зображення з текстом. Зміна відбувається шляхом поступового розмиття одного зображення та збільшенням чіткості наступного через певний проміжок часу. Приклад коду наведений на рисунку 3.10.

```
.camera_wrap .camera_pag .camera_pag_ul li {
    border: 1px solid #cfcfcf;
    border-radius: 500px;
    width: 14px;
    height: 14px;
    line-height: 0;
    transition: 0.5s ease;
    -o-transition: 0.5s ease;
    -webkit-transition: 0.5s ease;
    margin-left: 4px;
    float: left;
    box-shadow: inset 0 1px 1px #909090;
    cursor: pointer;
}
```

Рисунок 3.10 – Приклад роботи слайдеру

JavaScript – це мова програмування, яка дозволяє створювати скрипти, які вбудовуються в HTML-сторінки та виконуються в браузері відвідувача сторінки [18]. JavaScript надає наступні можливості:

- вионання математичних операцій;
- обробка даних без підключення до мережі Інтернет;
- взаємодія з користувачем;
- події;
- додавання анімації та багато іншого.

Із допомогою JavaScript було створено форму замовлення. Суть її роботи полягає в тому, що користувач поступово обирає параметри замовлення, які послідовно пов'язані між собою. Реалізовано форму у вигляді блоків, які сховані за замовчуванням, та які відкриваються після натискання на необхідний блок. Після вибору параметрів замовлення відкривається сторінка підтвердження замовлення, яка згруповує всі попередні рішення та дає можливість підтвердити замовлення шляхом введення дати проведення та контактних даних. Приклад роботи скрипту наведений на рисунку 3.11.

```
<script type="text/javascript">
document.getElementById("menu2_1").style.display="none";
document.getElementById("menu2_2").style.display="none";
document.getElementById("menu2_3").style.display="none";
document.getElementById("menu2_4").style.display="none";

class Order {
  constructor(elem) {
    this._elem = elem;
    elem.onclick = this.onClick.bind(this); // (*)
  }

  birthday() {
    etype = 'День народження';
    document.getElementById("menu1").style.display="none";
    document.getElementById("menu2_1").style.display="block";
document.getElementById("back").style.display="block";
  }
}
```

Рисунок 3.11 – Скрипт роботи форми

Для роботи з даними необхідно підключитися до бази даних. Вона потрібна для збереження замовлення, авторизації, відображення чи редагування даних. Приклад підключення до бази даних зображено на рисунку 3.12.


```

$hostname = 'localhost';
$username = 'user';
$passwordname = 'pass';
$basename = 'event_agency';
$conn = mysqli_connect($hostname, $username, $passwordname, $basename) or die ('Error');
mysqli_set_charset($conn, 'utf8');

```

Рисунок 3.12 – Підключення до бази даних

Після того, як база даних підключена, потрібно написати код для взаємодії бази з формою, що розміщена на сайті. Взаємодія полягає в тому, що значення, які будуть введені в певні поля форми, заповнять сповпець в самій базі даних або будуть використані для перевірки даних. Приклад цього наведений нижче (рис. 3.13).

```

if(isset($_POST['change']))
{
    $bsi = $_POST['birthday_sinfo'];
    $wsi = $_POST['wedding_sinfo'];
    $csi = $_POST['corporate_sinfo'];
    $casi = $_POST['calendar_sinfo'];
    $bp1 = $_POST['bprogram1'];
    $bp2 = $_POST['bprogram2'];
    $bp3 = $_POST['bprogram3'];
    $bp4 = $_POST['bprogram4'];
    $wp1 = $_POST['wprogram1'];
    $wp2 = $_POST['wprogram2'];
    $wp3 = $_POST['wprogram3'];
    $wp4 = $_POST['wprogram4'];
    $cp1 = $_POST['cprogram1'];
    $cp2 = $_POST['cprogram2'];
    $cp3 = $_POST['cprogram3'];
    $cp4 = $_POST['cprogram4'];
    $cap1 = $_POST['caprogram1'];
    $cap2 = $_POST['caprogram2'];
    $cap3 = $_POST['caprogram3'];
    $cap4 = $_POST['caprogram4'];

    $result = "UPDATE info SET birthday_sinfo=$bsi, wedding_sinfo='$wsi', wprogram1='$wp1', corporate_sinfo="

```

Рисунок 3.13 – Обробка даних через форму

ВИСНОВОК

У ході виконання дипломного проекту було розроблено web-додаток для організації діяльності івент-агенства.

Після проведення огляду останніх публікацій за темою роботи було визначено, що розробка та використання web-сайтів організаціями є досить актуальними сьогодні. Також було проаналізовано аналоги сайтів івент-агентств та виділено їх недоліки. У результаті була поставлена задача створити новий web-додаток організації діяльності івент-агентства. В аналітичній частині дипломної роботи була сформована мета проекту, визначені задачі для виконання та описані вимоги до продукту. Останні описані в технічному завданні (додаток А). Також було визначено технології, за допомогою яких буде створюватись web-додаток івент-агенства.

У проектній частині роботи проведено структурно-функціональне моделювання для визначення майбутнього функціоналу та визначено варіанти використання, які показують дії взаємодії системи з акторами.

У практичній частині роботи представлено архітектуру web-додатку організації діяльності івент-агентства та виконано його програмну реалізацію за допомогою визначених раніше технологій. Також було наведено приклади його роботи.

Використання розробленого web-додатку організації діяльності івент-агентства дозволяє скоротити час користувача під час оформлення замовлення на організацію власного івенту.

Результати даної дипломної роботи були апробовані на науково-практичній конференції ІМА 2020 в Сумському державному університеті (додаток В).

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Що таке сайт? [Електронний ресурс] – Режим доступу до ресурсу: <https://2ip.ua/ua/blog/website> (дата звернення 05.03.2020)
2. Что такое event-агентство? [Електронний ресурс] – Режим доступу до ресурсу: <https://eventum-premo.ru/эффективное-event-агентство> (дата звернення 10.03.2020)
3. Event агентство #SHALENIY [Електронний ресурс] – Режим доступу до ресурсу: <http://shaleniy.com.ua/>. (дата звернення 05.04.2020)
4. Event-agency "СОЛЬ" [Електронний ресурс] – Режим доступу до ресурсу: <https://salty.events/>. (дата звернення 05.04.2020)
5. Петров, В.И. Информационные системы / В.Н. Петров. СПб. : Питер, 2002. – 688 с. (дата звернення 07.04.2020)
6. Орлов, С.А. Технологии разработки программного обеспечения : учеб. / С.А. Орлов. – СПб. : Питер, 2002. – 464 с. (дата звернення 05.12.2019)
7. Анісімов А.В. Інформаційні системи та бази даних: Навчальний посібник для студентів факультету комп'ютерних наук та кібернетики. / Анісімов А.В., Кулябко П.П. – Київ. – 2017. – 110 с. (дата звернення 24.04.2020)
8. Табунщик Г. В. Проективання, моделювання та аналіз інформаційних систем: Навчальний посібник / Г.В. Табунщик, Р.К. Кудерметов, А. В. Притула. – Запоріжжя : ЗНТУ, 2011. – 292 с. (дата звернення 25.04.2020)
9. Калянов, Г.Н. CASE. Структурный системный анализ (автоматизация и применение) / Г.Н. Калянов. – М. : Лори, 1996. – с. (дата звернення 15.04.2020)
10. Леоненков, А.В. Самоучитель UML 2 / А.В. Леоненков. – СПб.: БХВ - Петербург, 2007. – 576с. (дата звернення 13.04.2020)

11. Ларман, К. Применение UML и шаблонов проектирования: Уч. Пос / К. Ларман. - М.: Издательский дом «Вильямс», 2001. - 496 с. (дата звернення 20.04.2020)
12. Иванов, К. К. Проектування бази даних. Роль процесу в створенні інформаційної системи / К. К. Иванов, А. А. Єфремов, І. А. Ващенко. - Текст: безпосередній // Молодий вчений. - 2016. - № 18 (122). - С. 40-42. - URL: <https://moluch.ru/archive/122/33704/> (дата звернення: 05.05.2020).
13. Анализ методов проектирования БД. // Shpargalum.ru. Электронный ресурс. URL: <http://shpargalum.ru/shpora-gos-povtas/proektirovanie-avtomatizirovannyix-sistem-na-osnove-bd/analiz-metodov-proektirovaniya-bd.html> (дата звернення: 06.05.2020).
14. Основы проектирования баз данных [Электронный ресурс] – Режим доступа до ресурсу: <https://metanit.com/sql/tutorial/1.1.php>. (дата звернення 07.05.2020)
15. MySQL 4 - Строковые функции [Электронный ресурс] Режим доступа: <http://www.codenet.ru/db/mysql/mystring4> (дата звернення 10.05.2020)
16. Структура веб-сайтів, різновиди веб-сайтів [Электронный ресурс] Режим доступа: <http://urokinformatiki.in.ua/urok-26-struktura-veb-sajtiv-riznovidiv-veb-sajtiv/> (дата звернення 02.05.2020)
17. CSS Tutorial [Электронный ресурс] – Режим доступа до ресурсу: <https://www.w3schools.com/css/>. (дата звернення 05.05.2020)
18. Dane, Cameron HTML5, JavaScript, and jQuery 24??“Hour Trainer / Dane Cameron. - Москва: Наука, 2015. - 150 с. (дата звернення 09.05.2020)

Додаток А

ТЕХНІЧНЕ ЗАВДАННЯ на розробку web-додатку організації діяльності івент-агентства

Суми 2020

1 Призначення й мета створення web-додатку

1.1 Призначення web-додатку

Web-додаток повинен представляти діяльність івент-агентства.

1.2 Мета створення web-додатку –

Забезпечення належної організації діяльності івент-агентства та скорочення часу при оформленні замовлення.

1.3 Цільова аудиторія

Цільову аудиторію web-додатку можна розділити на наступні групи:

1. Замовники.
2. Співробітники.
3. Інші зацікавлені відвідувачі.

2 Вимоги до web-додатку

2.1 Вимоги до web-додатку в цілому

2.1.1 Вимоги до структури й функціонування web-додатку

Web-додаток повинен бути реалізований у вигляді сайту, доступного в мережі Інтернет. Сайт повинен складатися із взаємозалежних розділів із чітко розділеними функціями.

2.1.2 Вимоги до персоналу

Персонал не повинен володіти спеціальними технічними навичками та знаннями технологій для підтримання сайту й експлуатації web-інтерфейсу через систему керування сайтом (CMS). Винятком є загальні навички роботи з персональним комп'ютером і стандартним web-браузером.

2.1.3 Вимоги до збереженні інформації

У системі керування сайтом повинен бути передбачений механізм резервного копіювання структури й вмісту бази даних. Процедура резервного копіювання повинна проводитися автоматично не рідше 1 рази в тиждень. Резервне копіювання графічного вмісту повинне здійснюватися вручну.

2.1.4 Вимоги до розмежування доступу

Інформація, розташовувана на сайті, є загальнодоступною.

Користувачів сайту можна розділити на 3 групи відповідно до прав доступу:

1. Відвідувачі.
2. Редактор.
3. Адміністратор.

Відвідувачі мають доступ тільки до загальнодоступної частини сайту. Доступ до адміністративної частини мають користувачі із правами редактора й адміністратора.

Редактор може виконувати всі ті ж дії, що й Відвідувач, але й має можливість отримувати інформацію про створені замовлення та редагувати їх.

Адміністратор може виконувати всі ті ж дії, що й Редактор, але й має можливість додавати користувачів із правами Редактора та редагувати матеріали розділів.

Доступ до адміністративної частини повинен здійснюватися з використанням унікального логіна й пароля.

2.2 Вимоги до функцій, виконуваних сайтом

2.2.1 Основні вимоги

2.2.1.1 Структура сайту

Сайт повинен складатися з наступних розділів:

Головна – виводиться інформація про івент-агентство та його послуги;

Організація свят – описується івент та його програма проведення.

Замовлення – оформлення замовлення для його подальшої організації агентством;

Інформація про компанію – показується інформація щодо агентств, її діяльність та персонал.

2.2.1.2 Навігація

Користувацький інтерфейс сайту повинен забезпечувати наочне, інтуїтивно зрозуміле представлення структури розміщеної на ньому інформації, швидкий і логічний перехід до розділів і сторінок. Навігаційні елементи повинні забезпечувати однозначне розуміння користувачем їх змісту: посилання на сторінки повинні бути мати заголовок, умовні позначки відповідати загальноприйнятим. Графічні елементи навігації повинні бути мати альтернативний підпис.

Система повинна забезпечувати навігацію по всіх доступних користувачу ресурсам і відображати відповідну інформацію. Для навігації

повинна використовуватися система контент-меню. Меню повинне бути представлено у вигляді текстового блоку (список гіперпосилань) у верхній частині сторінки (залежно від затвердженого дизайну).

2.2.1.3 Наповнення сайту (контент)

Сторінки всіх розділів сайту повинні формуватися програмним шляхом на підставі інформації з бази даних на сервері.

Модифікація вмісту розділів повинна здійснюватися за допомогою адміністраторського web-інтерфейсу (системи керування сайтом), який без застосування спеціальних навичок програмування (без використання програмування й спеціального кодування або форматування) повинен передбачати можливість редагування інформаційного вмісту сторінок сайту. Наповнення інформацією повинне проводитися з використанням шаблонів сторінок сайту.

2.2.1.4 Система навігації (карта сайту)

Взаємозв'язок між розділами й підрозділами сайту (карта сайту) представлено на рисунку 1.



Рисунок 1– Карта сайту

2.2.2 Вимоги до функціональних можливостей

Система керування контентом (адміністративна частина сайту) повинна надавати можливість додавання, редагування й видалення вмісту статичних і динамічних сторінок. Також повинна бути передбачена можливість додавання інформації без відображення на сайті.

2.2.2.1 Функціональні можливості розділів

На головній сторінці будуть представлені наступні елементи:

- авторизація користувачів та зміна контентного наповнення авторизованими користувачами;
- перегляд короткої інформації про агентство;
- перегляд короткої інформації про послуги;
- можливість замовити організацію свята через кнопку замовлення;
- можливість переходу між сторінками використовуючи меню зверху.

На сторінці про івент будуть представлені наступні елементи:

- авторизація користувачів та зміна контентного наповнення авторизованими користувачами;
- перегляд повної інформації про свята, які організує агентство та їх програму;
- можливість замовити організацію свята через кнопку замовлення;
- можливість переходу між сторінками використовуючи меню зверху.

На сторінці про компанію будуть представлені наступні елементи:

- авторизація користувачів та зміна контентного наповнення авторизованими користувачами;
- перегляд повної інформації про компанію та її персонал;
- можливість замовити організацію свята через кнопку замовлення;
- можливість переходу між сторінками використовуючи меню зверху.

2.2.2.2 Загальні вимоги

Стиль сайту можна описати як сучасний, діловий.

Оформлення не повинне защемляти інформативність: хоча на сайті не повинно бути досить багато графіки, він повинен бути зручний користувачам у плані навігації й цікавий для багаторазового відвідування.

Розташування елементів на головній сторінці сайту схематично показано на рисунку 2.

		Логотип								
День народження	Весілля	Корпоратив	Календарна подія	Про компанію	ЗАМОВИТИ СВЯТО					
		Хто ми такі								
Що ми організуємо										
День народження	Весілля	Корпоратив	Календарна подія							
<table border="0"> <tr> <td style="border: 1px solid black; padding: 5px;">Логотип</td> <td style="padding: 5px;"> День народження Весілля Корпоратив Календарна подія </td> <td style="padding: 5px;"> Мова: <input type="text" value="Українська"/> </td> <td style="padding: 5px;"> Логін <input type="text"/> Пароль <input type="text"/> </td> <td style="border: 1px solid black; padding: 5px;">Вхід</td> </tr> </table>						Логотип	День народження Весілля Корпоратив Календарна подія	Мова: <input type="text" value="Українська"/>	Логін <input type="text"/> Пароль <input type="text"/>	Вхід
Логотип	День народження Весілля Корпоратив Календарна подія	Мова: <input type="text" value="Українська"/>	Логін <input type="text"/> Пароль <input type="text"/>	Вхід						

Рисунок 2 – Головна сторінка

2.2.2.3 Типові навігаційні й інформаційні елементи

шапка сайту;
основне меню;
основне поле контенту.

2.2.2.4 Шапка сайту

Шапка сайту повинна містити логотип і назву сайту. Логотип є посиланням на головну сторінку сайту.

2.2.2.5 Основне меню

Основне меню повинне розташовуватися у верхній частині вікна (під логотипом).

2.2.2.6 Основне поле контенту

Основне поле контенту повинне розташовуватися в центрі сторінки. У цьому полі відображається основний зміст обраного розділу. Стильове оформлення матеріалів і їх елементів (посилань, заголовків, основного тексту, зображень, форм, таблиць і т.п.) повинне бути єдиним для всього web-сайту.

2.3 Вимоги до видів забезпечення

2.3.1 Вимоги до інформаційного забезпечення

Реалізація сайту відбувається з використанням:

- Apache 2.4;
- PHP 5.6;
- MySQL 5.6;
- JavaScript;
- MySQLWorkbench 6.3.5.

2.3.2 Вимоги до лінгвістичного забезпечення

Сайт повинен бути виконаний українською мовою.

2.3.3 Вимоги до програмного забезпечення

Програмне забезпечення клієнтської частини повинне задовольняти наступним вимогам:

web-браузер: Internet Explorer 7.0 і вище, або Firefox 3.5 і вище, або Opera 9.5 і вище, або Safari 3.2.1 і вище, або Chrome 2 і вище;

включена підтримка javascript, Flash і cookies.

2.3.4 Вимоги до апаратного забезпечення

Апаратне забезпечення серверної частини повинне задовольняти наступним вимогам:

web-сервер Apache з модулем MOD_REWRITE;

не менш 500 МБ вільного місця на диску.

Апаратне забезпечення клієнтської частини повинне забезпечувати підтримку програмного забезпечення клієнтської частини, зазначеного в п. 2.2.3

3 Склад і зміст робіт зі створення сайту

Докладний опис етапів роботи зі створення сайту наведено в таблиці 1.

Таблиця 1 – Етапи створення сайту

№	Склад і зміст робіт	Строк розробки (у робочих днях)
1	Розробка каркасу: Проектування розмітки та наповнення сайту	2 дні
2	Авторизація: Розроблення та реалізація блоку Авторизації на web-сайті	1 день

Продовження таблиці 1 – Етапи створення сайту

3	Замовлення: Створення модуля замовлення івенту	1 день
4	Адміністративна частина: Розробка модуля для редагування даних та перегляду списку замовлень.	2 день
5	Головний вивід (основна контентна частина): Реалізація виводу даних у центральній частині сайту	2 дні
6	Завершення роботи: Проведення стилістичних виправлень web-сайту, перевірка (тестування) реалізованого функціоналу	1 день
	Загальна тривалість робіт (з урахуванням резервного строку на налагодження й виправлення помилок) і строк закінчення проекту	12

4 Вимоги до складу й змісту робіт із введення сайту в експлуатацію

Для створення умов функціонування, при яких гарантується відповідність створюваного сайту вимогам сьогодення ТЗ і можливість його ефективної роботи, в організації Замовника повинен бути проведений певний комплекс заходів.

Для переносу сайту на хостинг необхідно, щоб параметри хостинга відповідали вимогам, зазначеним у ТЗ. На хостинг переноситься програма (сайт), зверстаний шаблон дизайну й структура й наповнення бази даних з подальшою їх доробкою.

Додаток Б

Планування робіт

У даний час щодня святкується велика кількість урочистих подій, і не завжди для їх організації власноруч є час. Для вирішення даної проблеми були створені івент-агентства. Вони спеціалізуються на організації свят чи подій шляхом вибору програми проведення того чи іншого заходу з існуючих або написанням власної. Таким чином, організація події проводиться професіоналами, не гаючи час замовника.

Повсюдне поширення та доступність мережі Інтернет наряду з комп'ютерною технікою, досить скоро привели до того, що багато людей стали значну частину часу проводити онлайн – у середньому, близько трьох годин на добу. Для великої кількості людей мережа Інтернет стала основним джерелом інформації, яке поступово витіснило з їхнього життя друковані ЗМІ та телебачення, внаслідок чого, роль телевізійної та друкованої реклами стала падати.

Оцінюючи те, що відбувається, бізнес не міг не відреагувати на зміни. Значна кількість підприємців почала активно переносити частину власного «офлайнового» бізнесу в мережу Інтернет. Через деякий час, розуміння необхідності володіння web-ресурсом прийшло й до решти. Так, у відносно короткий термін, кількість різного роду комерційних web-сайтів значно зросла.

Деталізація мети проекту методом SMART. Метою дипломного проекту є розробка web-додатку, який буде забезпечувати належну організацію діяльності івент-агентства за рахунок скорочення часу при оформленні замовлення за допомогою розробленого продукту.

Результати деталізації методом SMART розміщені у таблиці Б.1.

Таблиця Б.1 – Деталізація мети проекту методом SMART

Specific (конкретна)	Розробити WEB-додаток для організації діяльності івент-агентства
Measurable (вимірювання)	Оскільки даний проект не є комерційним, то результатом його роботи є оцінка замовника.
Achievable (досяжна, узгоджена)	Ціль даного проекту вважається досяжною, так як розробника володіє усіма необхідними знаннями з HTML, CSS, мови програмування JavaScript, PHP, MySQL, які використовуються для розробки даного Web-додатку.
Relevant (реалістична)	У наявності є всі необхідні технічні та програмні засоби. Розробники достатньо кваліфіковані для виконання поставлених задач.
Time-framed (обмежена в часі)	Ціль має часове обмеження, оскільки web-додатку розроблюється з обмеженням у часі на основі сформованого календарного плану.

Планування змісту робіт. Для планування змісту структури робіт використовують WBS (Work Break Structure). Це графічне представлення згрупованих елементів проекту у вигляді пакета робіт, які ієрархічно пов'язані з продуктом проекту. На першому рівні WBS фіксується продукт проекту. Він повинен відповідати продукту проекту. Наступний II рівень відповідає діям або основним заходам для досягнення продукту проекту.

Потім триває розбивка цих дій доти, поки не відбувається виконання дій елементарних робіт. Елементарні роботи – це роботи, які мають один чіткий результат, який використовується при прийнятті цієї роботи; на які призначений один конкретний відповідальний; на неї можна обчислити витрати праці і тривалість виконання. Зазвичай декомпозиція завершується тоді, коли для

розкриття змісту потрібні вузькі фахівці, що знають технологічні особливості їх виконання. Діаграма WBS представлена на рисунку Б.1.

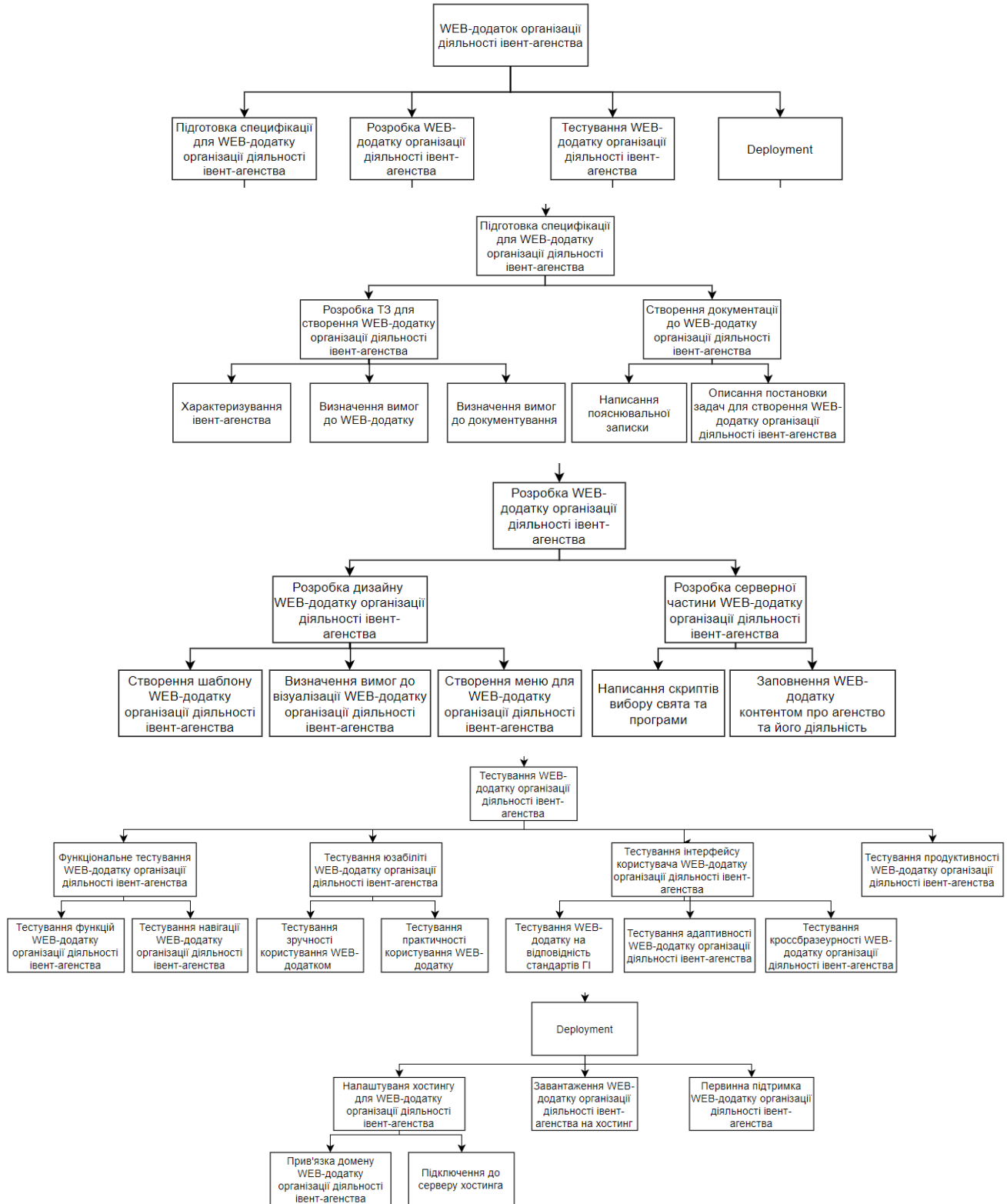


Рисунок Б.1 – WBS. Структура робіт проекту

Планування структури виконавців. Після побудови WBS розробляють організаційну структуру виконавців.

OBS – організаційна структура виконавців (організацій) проекту. Визначається за переліком пакетів робіт нижнього рівня кожної гілки WBS-структури. Представляється відповідальними (відповідальні – це не обов’язково керівники організацій(відділів), а ті люди які безпосередньо організують виконання робіт) за виконання пакетів робіт.

Організаційна структура представляє собою графічне відображення учасників проекту та їх відповідальних осіб, які задіяні в реалізації проекту. На верхньому рівні OBS розташована команда проекту. На наступному рівні фіксуються виконавці: організації, відділи тощо. Потім, рівнем нижче, для кожного виконавця вказують прізвища конкретних осіб, які будуть відповідати за виконання елементарних робіт WBS. Потрібно пам’ятати, що відповідальні – це не обов’язково керівники, а ті співробітники, які безпосередньо організують і відповідають у виконавця за виконання елементарної роботи, зазначеної у WBS. Для них ця елементарна робота також є проектом (у порівнянні з загальним проектом). Для себе вони також можуть побудувати WBS- структуру й застосовувати інші інструменти планування. Саме на цьому рівні закладається певна якість майбутнього продукту проекту.

Діаграма OBS представлена на рисунку Б.2. Список виконавців, що функціонують в проекті знаходиться в табл. Б.2.

Таблиця Б.2 – Список виконавців проекту

Ім’я	Роль	Проектна роль
Андрусишин І.К.	Виконавець	Виконує розробку web-додатку.
Антипенко В.П.	Керівник проекту	Формує завдання на розробку проекту.

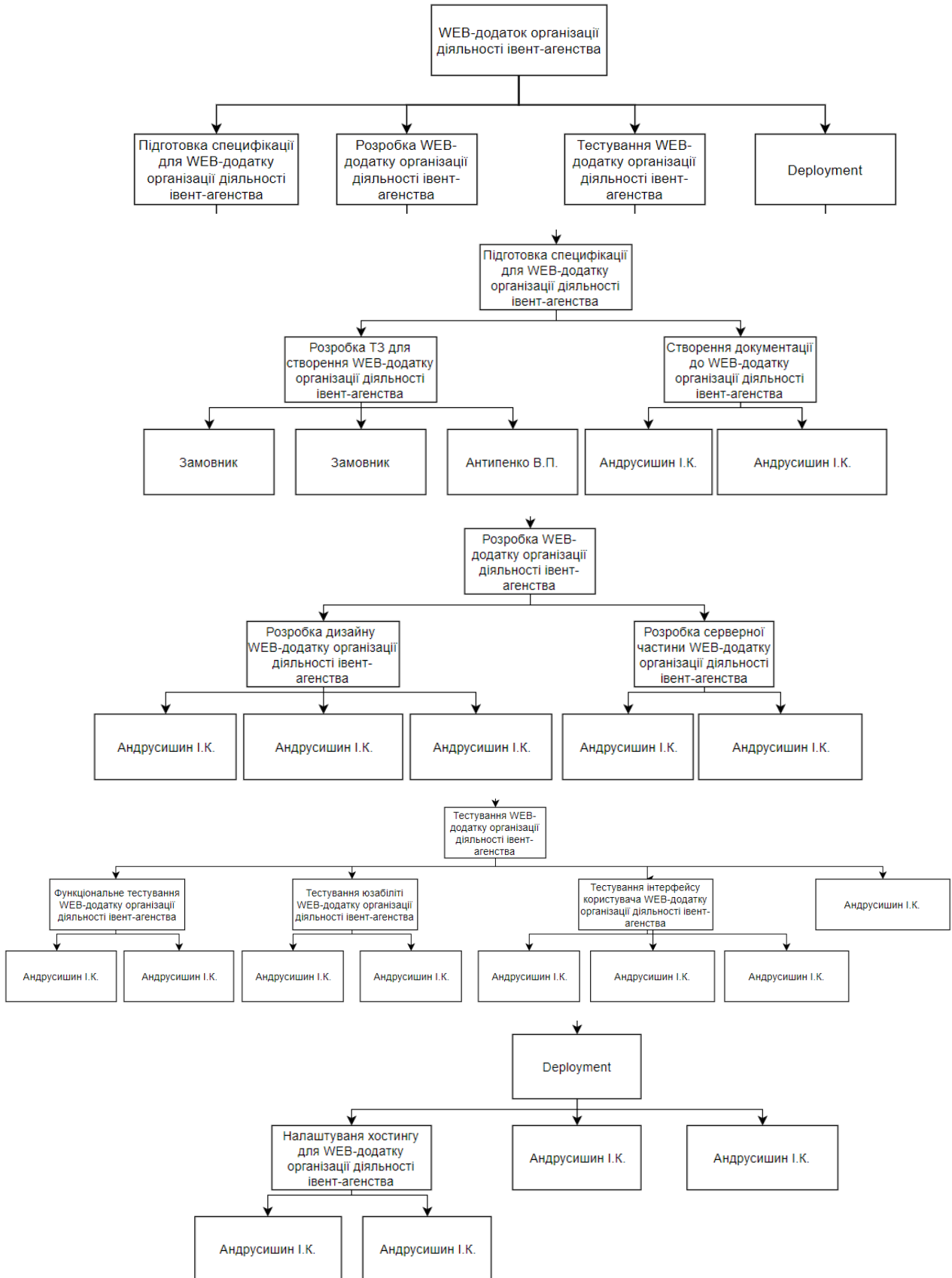


Рисунок Б.2 – OBS структура

Діаграма Ганта. Діаграма Ганта – це інструмент управління проектами, що допомагає планувати проекти всіх розмірів. Терміни та завдання управління проектами перетворюються на горизонтальну смужкову діаграму із зазначенням дати початку та закінчення, а також залежностей, планування та строків, у тому числі, скільки завдань виконано на етапі та хто є власником завдання. Це корисно для відстеження завдань, коли існує велика команда та декілька зацікавлених сторін, коли область застосування змінюється.

Діаграма Ганта представлена на рисунку Б.3.

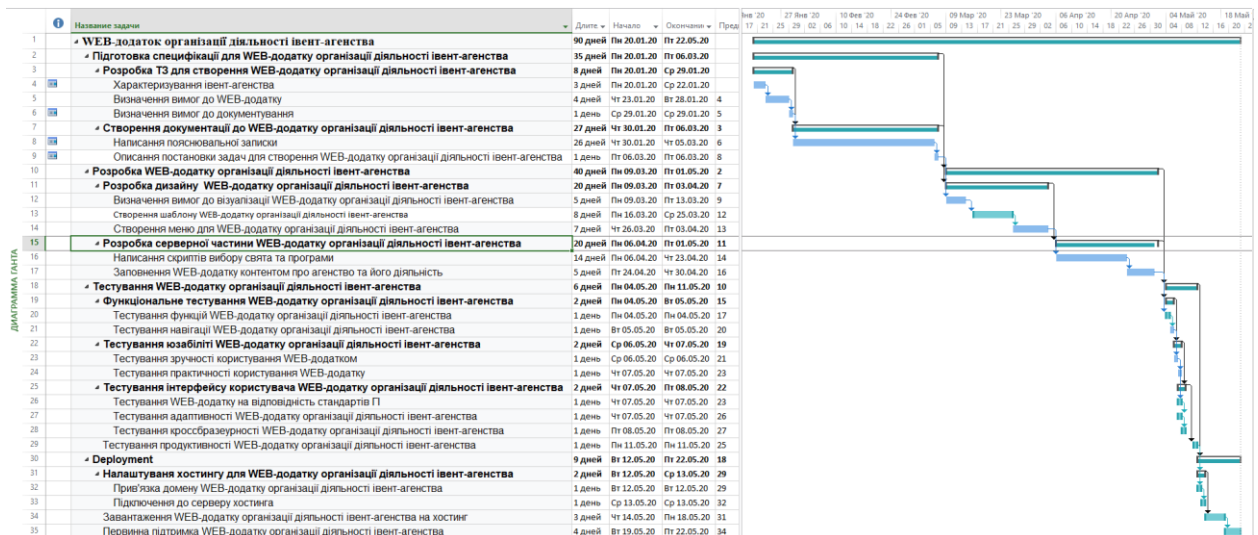


Рисунок Б.3 – Діаграма Ганта

Управління ризиками проекту. Управління ризиками проекту включає процеси, пов'язані із здійсненням планування управління ризиками, виявленням, аналізом, реагуванням, а також моніторингом і контролем ризиків в проекті.

Процес управління ризиками включає наступні етапи:

- 1) планування управління ризиками;
- 2) ідентифікація ризиків;
- 3) проведення якісного аналізу ризиків;
- 4) проведення кількісного аналізу ризиків;

- 5) планування реагування на відомі ризики;
- б) моніторинг та контроль ризиків.

Ризики проекту представлені у таблиці Б.2

Таблиця Б.2 – Ризики проекту

№	Назва	Опис	План вирішення ризиків	Вірогідність виникнення	Ступінь впливу	RV	Mitigation
1	Низька кваліфікація	Використання нових технологій, які не були вивчені розробником.	Необхідність закладати час на вивчення нової технології розробнику.	3	5	100	T and M
2	Важкість інтеграції	Ризик виникнення різних проблем в процесі інтеграції модулів.	Використання модулів з легкою інтеграцією.	2	2	200	Material
3	Низька продуктивність	Продуктивність - річ динамічна, нелінійна, і її досить важко оцінити. Найбільше значення має наслідок закону Паркінсона: розробники активізуються тільки до кінця терміну задачі проекту, а в решту часу працюють в півсили.	Використання гнучкої методології розробки.	5	4	0	Time
4	Завантаження на хостинг	Ризик раптового припинення діяльності хостингу.	Зміна хостингу на більш надійний.	1	3	50	Material
5	Бюджет	Ризик перевищення виділеного на проект бюджету.	Планування методом «набігаючої хвилі».	1	1	0	Time

Рівні критичності:

- 1) Negligible;

- 2) Low;
- 3) Medium;
- 4) High;
- 5) Critical.

Таблица Б.3 – Probability / Impact Matrix

5				3	
4					
3					1
2		2			
1	5		4		
Probability / Impact	1	2	3	4	5

Додаток В

АПРОБАЦІЯ

IMA :: 2020

СЕКЦІЯ 2: Інформаційні
технології проектування

Web application for organization the activity of event-agency

Andrusyshyn I.K., *Student* Antypenko V.P., *Senior Lecturer*
Sumy State University, Sumy, Ukraine

Nowadays celebrating various events plays an important role, as currently there are many additional services which everyone wants to get for their own event.

For instance, it can be an order the photographer, the presenter, the animator, the decorator etc. However, a modern person is quite busy today, and, unfortunately, does not have enough time to organize such events by his own.

Moreover, even a lack of certain skills can take place. These may be insufficient experience and/or fantasy in organizing mass events, fear of responsibility etc.

Therefore, today the appearance of event agencies has become logical in order to solve the issues of the modern initiator (customer), which are described above.

At the present time, the Internet takes an important place in the life of both each person and any organization. As two main prerequisites that have led to this state of affairs the rapid progress of web technologies and the growth of the society's well-being can be considered.

Furthermore, as now websites of various organizations are an important tool for developing their business, having their own representation in the global network for each market participant has become a necessary condition.

Thus, the purpose of this project is to develop the web application for organization the activity of event agency. The topicality of this work is to advertise an event agency for attracting the new customers and to simplify the process of event ordering.

The web application has been implemented using web technologies such as HTML5 for creating the markup of web page, CSS3 for describing their frontend, JavaScript for the web pages scripting to give them interactivity and PHP5 for scripting, which are run on the server. Sublime Text3 multi-function text editor has been used to write the web code.

The result of this work is a developed web application for organization the activity of event agency, which allows to reduce the user time during creation the order for the organization of customer's event.

Додаток Г

ЛІСТИНГ ПРОГРАМНОГО КОДУ

index.php

```

<?php
require_once ('system/header.php');
?>
<html>
  <head>
    <title>
      Івент-агентство "Святкуй!"
    </title>
    <link rel="shortcut icon" href="images/siteico.ico" type="image/x-
icon">
    <link rel="stylesheet" type="text/css" href="css/main.css">
    <link
href="https://fonts.googleapis.com/css?family=Lobster&display=swap"
rel="stylesheet">

    <link rel="stylesheet" media="screen" href="css/style.css">
    <link rel="stylesheet" href="css/camera.css">
    <link rel="stylesheet" href="css/carousel.css">
    <link rel="stylesheet" href="css/font-awesome.css">
    <script src="js/jquery.js"></script>
    <script src="js/script.js"></script>
    <script src="js/superfish.js"></script>
    <script src="js/camera.js"></script>
    <script type="text/javascript" src="script.js"></script>

    <script>
$(document).ready(function() {
  jQuery('#camera_wrap').camera({
    loader: false,
    pagination: false,
    thumbnails: false,
    height: '45%',
    caption: false,
    navigation: true,
    fx: 'simpleFade'
  });
});

$(window).load (
  function(){$('#.carousel1').carouFredSel({auto: false, prev:
'.prev1',next: '.next1', width: 1940, items: {
  visible : {min: 1,
  max: 1
},

```



```

height: 'auto',
width: 1940,

    }, responsive: true,

scroll: 1,

mousewheel: false,

swipe: {onMouse: true, onTouch: true}}});

} );
</script>
</head>
<body>
    <div class="header">
        <a href="index.php"></a>
    </div>
    <div class="menu">
<div class="link">
    <a href="birthday.php">День народження</a> <!-- </a> -->
</div>
<div class="link">
    <a href="wedding.php">Весілля</a><!-- </a>-->
</div>
<div class="link">
    <a href="corporate.php">Корпоратив</a><!-- </a>-->
</div>
<div class="link">
    <a href="calendar.php">Календарна подія</a><!-- </a>-->
</div>
    <div class="link">
    <a href="aboutus.php">Про нас</a><!-- </a>-->
</div>
    <div class="link">
    <a href="order.php">Замовити організацію</a><!-- </a>-->
</div>
</div>

<div id="shortinfo">
    <div class="slider_wrapper">
        <div id="camera_wrap" class="">
            <div data-src="images/xd1.jpg">
                <div class="caption fadeIn">
                    <h2>ВИСОКИЙ РІВЕНЬ СЕРВІСУ. КРЕАТИВНІСТЬ ТА
ІННОВАЦІЙНІСТЬ.</h2>
                </div>
            </div>
        </div>
        <div data-src="images/xd2.jpg">

```

```

        <div class="caption fadeIn">
            <h2>ВІДПОВІДАЛЬНІСТЬ, ЯКІСТЬ ТА СРОКИ.</h2>
        </div>
    </div>
    <div data-src="images/xd3.jpg">
        <div class="caption fadeIn">
            <h2>ЕФФЕКТИВНИЙ РОЗПОЛІД БЮДЖЕТУ. </h2>
        </div>
    </div>
</div>
</div>
<div id="what">Наші послуги
</div>
<div id="shortinfo1">
<table id="shorttable" border="1" width="100%" height="100%">
<tr>
    <th><div id="place1"><h2>День народження</h2>
        <div id="xd">
<?php
$hostname = 'localhost';
$username = 'root';
$passwordname = '';
$basename = 'event_agency';
$conn = new mysqli($hostname, $username, $passwordname, $basename);

mysqli_set_charset($conn, 'utf8');
$sql = "SELECT * FROM info";
$run = $conn->query($sql);
$row = $run->fetch_assoc();
echo $row['birthday_sinfo'];?>
        </div>
    </div>
</th>
    <th><div id="place2"><h2>Весілля</h2>
        <div id="xd">
<?php
echo $row['wedding_sinfo'];?>
        </div></div>
    </th>
    <th><div id="place3"><h2>Корпоратив</h2>
        <div id="xd">
<?php
echo $row['corporate_sinfo'];?>
        </div></div>
    </th>
    <th><div id="place4"><h2>Календарна подія</h2>
        <div id="xd">
<?php
echo $row['calendar_sinfo'];?>
        </div></div>
    </th>
</tr>
</table>
</div>

```

```

<footer>
<div id="footerimg">
</div>
<div id="fastlink">
<table border="1">
<tr>
<th><a href="news.html">День народження</a>
</th></tr>
<tr>
<th><a href="news.html">Весілля</a>
</th></tr>
<tr>
<th><a href="news.html">Корпоратив</a>
</th></tr>
<tr>
<th><a href="news.html">Календарна подія</a>
</th>
</tr>
<tr>
<th><a href="news.html">Про нас</a>
</th>
</tr>
</table>
</div>
<div id="footerinfo">xdxdxd</div>
<div id="logg">
<?php

if ($userData){
    echo "<script language='JavaScript'
ENGINE='text/javascript'>location='/'</script>";
} elseif ($_POST['login'] && $_POST['pass']) {
    $login = htmlspecialchars($_POST['login']);
    $pass = htmlspecialchars($_POST['pass']);
    $loginRequest = mysqli_query($conn, "SELECT * FROM `workers` WHERE
`login` = '$login' AND `pass` = '$pass'");
    if (mysqli_num_rows($loginRequest)) {
        $userData = mysqli_fetch_assoc($loginRequest);
        $_SESSION["login"] = $login;
        $_SESSION["pass"] = $pass;
    } else {
        echo "Wrong login or password!</br>";
    }
}

if ($userData){
    echo "" . $userData[name] . "| <a href = '/orderslist.php'>Admin
panel</a> | <a href = '/exit.php'>Вихід</a></br>";
} else {
    echo "
<form action='auth.php' method='POST'>
<div id='logininfo'>Логін:<br>
    Пароль:</div>
    <div id='logininput'><input type='text' name='login' size='20'><br><input
type='password' name='pass' size='20'></div>

```

```

    <div id='loggbutton'><input type='submit' value='Вхід'
height='20px'></div>
</form> ";
}
?>

</div>
</footer>
</body>
</html>

```

wedding.php

```

<html>
  <head>
    <title>
      Івент-агентство "Святкуй!"
    </title>
    <link rel="shortcut icon" href="images/siteico.ico" type="image/x-
icon">
    <link rel="stylesheet" type="text/css" href="css/main.css">
    <link
href="https://fonts.googleapis.com/css?family=Lobster&display=swap"
rel="stylesheet">

    <link rel="stylesheet" media="screen" href="css/style.css">
    <link rel="stylesheet" href="css/camera.css">
    <link rel="stylesheet" href="css/carousel.css">
    <link rel="stylesheet" href="css/font-awesome.css">
    <script src="js/jquery.js"></script>
    <script src="js/script.js"></script>
    <script src="js/superfish.js"></script>
    <script src="js/camera.js"></script>
    <script type="text/javascript" src="script.js"></script>

  </head>
  <body>
    <div class="header">
      <a href="index.php"></a>
    </div>
    <div class="menu">
    <div class="link">
      <a href="birthday.php">День народження</a> <!-- </a> -->
    </div>
    <div class="link">
      <a href="wedding.php">Весілля</a><!-- </a>-->
    </div>
    <div class="link">
      <a href="corporate.php">Корпоратив</a><!-- </a>-->
    </div>
    <div class="link">

```

```

<a href="calendar.php">Календарна подія</a><!-- </a>-->
</div>
  <div class="link">
    <a href="aboutus.php">Про нас</a><!-- </a>-->
  </div>
  <div class="link">
    <a href="order.php">Замовити організацію</a><!-- </a>-->
  </div>
</div>

<div id="info">
<div id="menuinfo1" style="line-height: 30px;">
<h1>Весілля</h1>
<div style="width:80%; vertical-align:middle; text-align:center;display:
inline-block;  ">
  <?php
$hostname = 'localhost';
$username = 'root';
$passwordname = '';
$basename = 'event_agency';
$conn = new mysqli($hostname, $username, $passwordname, $basename);

mysqli_set_charset($conn, 'utf8');
    $sql = "SELECT * FROM info";
    $run = $conn->query($sql);
    $row = $run->fetch_assoc();
    echo $row['wedding_sinfo'];?>
</div>
  <div id="what" style="color:white;">Наші програми організації</div>
<div id="wprogram1" class="choice"><a data-action="program1">Класичне
весілля</a></div>
  <div id="wprogram2" class="choice"><a data-action="program2">Креативне
тематичне весілля</a></div>
  <div id="wprogram3" class="choice"><a data-action="program3">Національне
весілля</a></div>
  <div id="wprogram4" class="choice"><a data-
action="program4">Екстримальне весілля</a></div></div>
  <div id="menuinfo2">
  <div id="wprogram1text" style="width:80%; vertical-align:middle; text-
align:center;display:inline-block;">

</div>
<div id="wprogram2text"><?php
    $sql = "SELECT * FROM info";
    $run = $conn->query($sql);
    $row = $run->fetch_assoc();
    echo $row['wprogram1'];?></div>
<div id="wprogram3text"><?php
    $sql = "SELECT * FROM info";
    $run = $conn->query($sql);
    $row = $run->fetch_assoc();
    echo $row['wprogram1'];?></div>

```

```

<div id="wprogram4text"><?php
    $sql = "SELECT * FROM info";
    $run = $conn->query($sql);
    $row = $run->fetch_assoc();
    echo $row['wprogram1'];?></div>
</div>
</div>

<script>
document.getElementById("wprogram1text").style.display="none";
document.getElementById("wprogram2text").style.display="none";
document.getElementById("wprogram3text").style.display="none";
document.getElementById("wprogram4text").style.display="none";

class Info {
    constructor(elem) {
        this._elem = elem;
        elem.onclick = this.onClick.bind(this);
    }

    program1() {
        document.getElementById("wprogram1text").style.display="block";
        document.getElementById("wprogram1").style.border="5px solid
white";
        document.getElementById("wprogram2text").style.display="none";
        document.getElementById("wprogram3text").style.display="none";
        document.getElementById("wprogram4text").style.display="none";
        document.getElementById("wprogram2").style.border="none";
        document.getElementById("wprogram3").style.border="none";
        document.getElementById("wprogram4").style.border="none";
    }

    program2() {
        document.getElementById("wprogram2text").style.display="block";
        document.getElementById("wprogram2").style.border="5px solid white";
        document.getElementById("wprogram1text").style.display="none";
        document.getElementById("wprogram3text").style.display="none";
        document.getElementById("wprogram4text").style.display="none";
        document.getElementById("wprogram1").style.border="none";
        document.getElementById("wprogram3").style.border="none";
        document.getElementById("wprogram4").style.border="none";
    }

    program3() {
        document.getElementById("wprogram3text").style.display="block";
        document.getElementById("wprogram3").style.border="5px solid white";
        document.getElementById("wprogram2text").style.display="none";
        document.getElementById("wprogram1text").style.display="none";
        document.getElementById("wprogram4text").style.display="none";
        document.getElementById("wprogram2").style.border="none";
        document.getElementById("wprogram1").style.border="none";
        document.getElementById("wprogram4").style.border="none";
    }

    program4() {

```

```

document.getElementById("wprogram4text").style.display="block";
document.getElementById("wprogram4").style.border="5px solid white";
document.getElementById("wprogram2text").style.display="none";
document.getElementById("wprogram3text").style.display="none";
document.getElementById("wprogram1text").style.display="none";
document.getElementById("wprogram2").style.border="none";
document.getElementById("wprogram3").style.border="none";
document.getElementById("wprogram1").style.border="none";
}

onClick(event) {
    let action = event.target.dataset.action;
    if (action) {
        this[action]();
    }
}
}

new Info(info);
</script>

<footer>
<div id="footering">
</div>
<div id="fastlink">
<table border="1">
<tr>
<th><a href="news.html">День народження</a>
</th></tr>
<tr>
<th><a href="news.html">Весілля</a>
</th></tr>
<tr>
<th><a href="news.html">Корпоратив</a>
</th></tr>
<tr>
<th><a href="news.html">Календарна подія</a>
</th>
</tr>
<tr>
<th><a href="news.html">Про нас</a>
</th>
</tr>
</table>
</div>
<div id="footerinfo">xdxdxd</div>
<div id="logg">
<?php
//require_once ('system/header.php');

if ($userData){
    echo "<script language='JavaScript'
ENGINE='text/javascript'>location='/</script>";
} elseif ($_POST['login'] && $_POST['pass']) {

```

```

    $login = htmlspecialchars($_POST['login']);
    $pass = htmlspecialchars($_POST['pass']);
    $loginRequest = mysqli_query($conn, "SELECT * FROM `workers` WHERE
`login` = '$login' AND `pass` = '$pass'");
    if (mysqli_num_rows($loginRequest)) {
        $userData = mysqli_fetch_assoc($loginRequest);
        $_SESSION["login"] = $login;
        $_SESSION["pass"] = $pass;
    } else {
        echo "Wrong login or password!</br>";
    }
}

if ($userData){
    echo "" . $userData[name] . " | <a href = '/orderslist.php'>Admin
panel</a> | <a href = '/exit.php'>Вихід</a></br>";
} else {
    echo "
<form action='auth.php' method='POST'>
<div id='logginfo'>Логін:<br>
    Пароль:</div>
    <div id='logininput'><input type='text' name='login' size='20'><br><input
type='password' name='pass' size='20'></div>
    <div id='loggbutton'><input type='submit' value='Вхід'
height='20px'></div>
</form> ";
}
?>

</div>

</footer>
</body>
</html>

```