

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ	8
ВСТУП.....	9
1 АНАЛІЗ ЗАДАЧІ ТА ХАРАКТЕРИСТИКА ПРЕДМЕТНОЇ ОБЛАСТІ	10
1.1 Система взаємодії користувачів з інформаційним ресурсом	10
1.2 Характеристика типового інформаційного ресурсу	10
1.3 Класифікація інформаційних ресурсів	14
1.4 Склад інформаційних ресурсів	16
1.5 Web-хостинг	17
1.6 WAMP комплекси та їх характеристика	18
1.6.1 Denwer	19
1.6.2 EasyPHP	20
1.6.3 OpenServer	21
1.6.4 WampServer	22
1.7 Постановка технічного завдання	23
1.8 Висновки	24
2 ВИБІР І ОБҐРУНТУВАННЯ ПРОГРАМНИХ ЗАСОБІВ ДЛЯ РОЗРОБКИ ІНФОРМАЦІЙНОГО РЕСУРСУ	26
2.1 HTML	26
2.2 PHP	27
2.3 jQuery	29
2.4 AJAX	30
2.5 Бази даних	32
2.5.1 MySQL	34
2.6 Зв'язок клієнтської й серверної частин інформаційного ресурсу	35
2.7 Проектування Баз даних для інформаційного ресурсу	36
2.8 Висновки	38
3 РОЗРОБКА ІНФОРМАЦІЙНОГО РЕСУРСУ	39
3.1 Створення бази даних MySQL для ІКС.....	39
3.2 Головна сторінка інформаційного ресурсу.....	40

3.2.1 Модуль "connection.php"	42
3.2.2 Модуль "header.php"	43
3.2.3 Модуль "content.php"	45
3.2.4 Модуль "leftblock.php"	46
3.2.5 Модуль "footer.php"	47
3.3 Сторінка "Команда"	49
3.4 Сторінка "Матчи"	52
3.5 Сторінка "Календарь туров чемпіоната"	54
3.6 Сторінка "Фотогалерея"	57
3.7 Висновки	59
4 ОХОРОНА ПРАЦІ	60
4.1 Загальні питання з охорони праці	60
4.1.1 Правові та організаційні основи охорони праці	61
4.1.2 Організаційно-технічні заходи з безпеки праці	62
4.2 Аналіз стану умов праці	63
4.2.1 Вимоги до приміщень	63
4.2.2 Вимоги до організації місця праці	64
4.2.3 Навантаження та напруженість процесу праці	65
4.3 Виробнича санітарія	67
4.3.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу.	67
4.3.2 Пожежна безпека	69
4.3.3 Електробезпека	71
4.4 Гігієнічні вимоги до параметрів виробничого середовища	71
4.4.1 Параметри мікроклімату	71
4.4.2 Освітлення	73
4.4.3 Шум та вібрація, електромагнітне випромінювання	75
4.4.4 Вентилювання	76
4.5 Заходи з організації виробничого середовища та попередження виникнення питань надзвичайних ситуацій	77
4.6 Висновки до розділу "Охорони праці"	81

ВИСНОВКИ	83
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	84
ДОДАТОК А. Лістинг веб-порталу	87
ДОДАТОК Б. Слайди презентації	98

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

ІКС - інформаційна комп'ютерна система.

ПЗ - програмне забезпечення.

БД - база даних.

ІР - інтернет ресурс.

WWW - World Wide Web.

HTML - HyperText Markup Language.

PHP - Personal Home Page.

WAMP - акронім від Windows, Apache, MySQL та PHP.

HTTP - HyperText Transfer Protocol - «протокол передачі гіпертексту».

GPL - GNU General Public License.

ВСТУП

Мережа Інтернет немислима для більшості з нас без поняття Web-додаток або більш модного, портал. Web-додаток – це локалізована наявність інформації в глобальній мережі, що розуміє під собою як зберігання даних, так і їх відображення користувачу за його запитом. Є три суті роботи з інформацією, а саме, зберігання даних (сховище даних), їх обробка та відображення (інтерфейс).

WWW - інтелектуальне досягнення людства в епоху високих технологій. На сьогоднішній день WWW містить мільйони ресурсів, на яких розміщена будь-яка інформація. Люди отримують доступ до цієї інформації за допомогою використання технології Internet. Для навігації в WWW використовуються спеціальні програми - web-браузери, які суттєво полегшують користування ресурсами WWW. Вся інформація в web-браузері відображається у вигляді Web-сторінок, які є основним елементом байтів WWW.

У дипломному проєкті розглядається створення ІКС футбольного клубу України "Шахтар Донецьк".

Для реалізації даної мети необхідно вирішити наступні основні задачі:

- ознайомитися з сучасними Інтернет технологіями;
- вивчення програмного інструментарію, що застосовується для розробки і створення Web-додатків;
- виявити і врахувати методи та способи подання на Web-сторінках різних видів інформації, що не перешкоджають їх доступності;
- визначитися зі структурою Web-сторінок.

Оволодіння знаннями в області Web-програмування дозволить легко вирішити проблему пов'язану зі створенням інформаційного ресурсу футбольного клубу, та реалізацією основних задач.

1 АНАЛІЗ ЗАДАЧІ ТА ХАРАКТЕРИСТИКА ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Система взаємодії користувачів з інформаційним ресурсом

Розвиток сучасних Інтернет технологій дає можливість створювати web-додатки та web-сервіси, які надають звичайним користувачам інтернету можливість отримувати потрібну їм інформацію у вільному доступі.

Однією із задач, яка буде розглядатися в даній дипломній роботі – задача взаємодії користувачів інтернету та інформаційного ресурсу футбольного клубу України "Шахтар Донецьк".

Існуюча система взаємодії користувачів з ресурсом полягає в наступному.

Користувачі інтернету шукаючи потрібну їм інформацію переглядають різні ресурси, втрачаючи при цьому свій час.

Приймаючи до уваги вище викладене постановка завдання зводиться до наступного: потрібно створити інформаційний ресурс, який би надавав користувачам інтернету такі можливості:

- а) перегляд новин улюбленого футбольного клубу;
- б) ознайомлення зі складом футбольного клубу;
- в) перегляд результатів усіх матчів футбольного сезону;
- г) довідкова інформація про футбольний клуб;
- д) подача своїх пропозицій щодо вдосконалення web-ресурсу.

1.2 Характеристика типового інформаційного ресурсу

Перш ніж приступити до створення Web-сторінок, необхідно познайомитися з основними поняттями, термінами і допоміжними інструментальними засобами.

Web-додаток це набір сторінок в форматі HTML, які знаходяться на Web-сервері.

Web-сервер - це постійно підключений до Інтернету комп'ютер, який передає ці сторінки за запитом користувачів.

Web-вузол - це набагато більше, ніж просто сукупність Web-сторінок. Це і система гіперпосилань, що зв'язує вузол в єдине ціле, і єдине стилістичне оформлення вузла, і своєчасне оновлення інформації на вузлі, і швидкість завантаження окремих сторінок, і багато іншого.

Основна відмінність Web-документів від звичайних електронних документів складається в наявності гіперпосилань. Гіпертекстові посилання дозволяють, переглядаючи Web-документ, перейти до іншої його частини або іншого Web-документу, або звернутися до однієї зі служб Інтернету. Як правило, частина тексту Web-документа, відповідна гіперпосиланням, виділяється синім кольором або підкреслюється. При переміщенні на гіперпосилання курсор миші приймає форму розгорнутої долоні з вказівним пальцем. Якщо клацнути таким покажчиком за гіперпосиланням, то відбудеться перехід за адресою, яка вказана в гіперпосиланні. При поверненні з гіперпосилання на сторінку синій колір заміниться на вишневий. Заміна кольору дозволяє користувачеві відслідковувати «відпрацьовані» гіперпосилання.

Домашня сторінка - це web-сторінка, яка передається сервером по запиту користувача, який вказав у запиті адресу ресурсу. Файлу, що містить домашню сторінку ресурсу, присвоюється стандартне ім'я index.html. Домашня сторінка - це обличчя ресурсу, тому її проектуванню і оформленню приділяється особливе значення, так як основна її функція - надання користувачеві зручних способів переходу на інші сторінки і документи web-додатку.

Всі web-додатки в сукупності складають Всесвітню павутину, де комунікація (павутина) об'єднує сегменти інформації світового співтовариства в єдине ціле - базу даних і комунікації планетарного масштабу. Для прямого доступу клієнтів до web-додатків на серверах був спеціально розроблений протокол HTTP.

У більшості випадків в Інтернеті одному web-додатку відповідає одне доменне ім'я. Саме по доменних іменах ресурси ідентифікуються в глобальній мережі. Можливі інші варіанти: один ресурс на декількох доменах або кілька ресурсів під одним доменом. Звичайно декілька доменів використовують великі web-ресурси (веб-портали) щоб логічно відокремити різні види надаваних послуг (mail.google.com, news.google.com, maps.google.com). Нерідкі й випадки виділення окремих доменів для різних країн або мов. Наприклад, google.ru і google.fr логічно являються веб-порталом Google на різних мовах, але технічно це різні ресурси. Об'єднання декількох веб-порталів під одним доменом характерно для безкоштовних хостингів. Іноді для ідентифікації ресурсу в адресі після зазначення хосту стоїть тильда й ім'я ресурсу: example.com/~my-site-name/, а найчастіше використовується домен третього рівня: my-site-name.example.com.

Апаратні сервера для зберігання web-додатку називаються web-серверами. Сама послуга зберігання називається web-хостингом. Раніше кожний ресурс зберігався на своєму власному сервері, але із збільшенням Інтернету, технологічним поліпшенням серверів на одному комп'ютері стало можливо розміщення безлічі ресурсів (віртуальний хостинг). Зараз сервера для зберігання тільки одного ресурсу називаються виділеними (англ. dedicated).

Той самий ресурс може бути доступний по різних адресах і зберігається на різних серверах. Копія оригінального ресурсу в такому випадку називається дзеркалом. Існує також поняття оффлайнова версія додатку - це копія ресурсу, що може бути переглянута на будь-якому комп'ютері без підключення до комп'ютерної мережі й використання серверного програмного забезпечення (ПО). При розробці веб-порталу який тестують і налагоджують саме в оффлайновій версії, для того, щоб не демонструвати нісенітницю й помилки, прорахунки великого проекту. Саме для тестування в корпоративній мережі, або на початку в Інтернеті з обмеженим доступом під паролем запрошуються досвідчені «тестери». Це дозволяє прискорити виробництво великих проектів і налагодити їх для масового відвідувача (користувача).

Проекти великих web-додатків і порталів можуть зробити тільки обізнані й зацікавлені фахівці, на рівні перших осіб в організаційній структурі, являючись по суті партнером і авангардом менеджменту підприємства (організації). Активна комунікація на порталі часто виконує функцію директора направлення й офісу зі службою супроводу (переписка, комутатор прямого спілкування, оперативна довідка, і ін.) Багато порталів обновляють частіше ніж раз на день, а інтернет-магазини, - по факту руху товару (нові надходження, відсутності товару в наявності). Web-додатки з новинами реально виставляють інформацію, з точністю до хвилини, тому що журналісти мають пріоритети на цитування першоджерел згідно авторського права, пріоритету посилань, рейтингу, і ін.

Web-додаток, який представляє інформацію про одну або декілька організацій, чи то футбольний клуб, чи велике підприємство, називається інформаційним ресурсом. До інформаційного ресурсу пред'являються вимоги в частині інформаційного вмісту, графічного дизайну, навігації, хостингу.

Офіційний ресурс зазвичай має наступні розділи:

- а) новини та актуальна інформація;
- б) детальна інформація про організацію;
- в) напрямки діяльності;
- г) структура організації;
- г) контактна інформація.

Ресурси футбольних клубів відносяться до групи спортивних порталів. Під спортивним web-додатком розуміється сукупність web-сторінок з повторюваним дизайном, що несуть в собі цілеспрямований процес інформування про діяльність організації (в даному випадку "футбольного клубу"), суспільства, об'єднаних за змістом, навігаційно і фізично знаходяться на одному сервері.

1.3 Класифікація інформаційних ресурсів

На даний час не існує єдиної класифікації типів web-додатків. Як правило, ресурси розрізняють за задачами, що вирішуються ним[1]. Один з варіантів класифікації додатків зображено на рисунку 1.1.

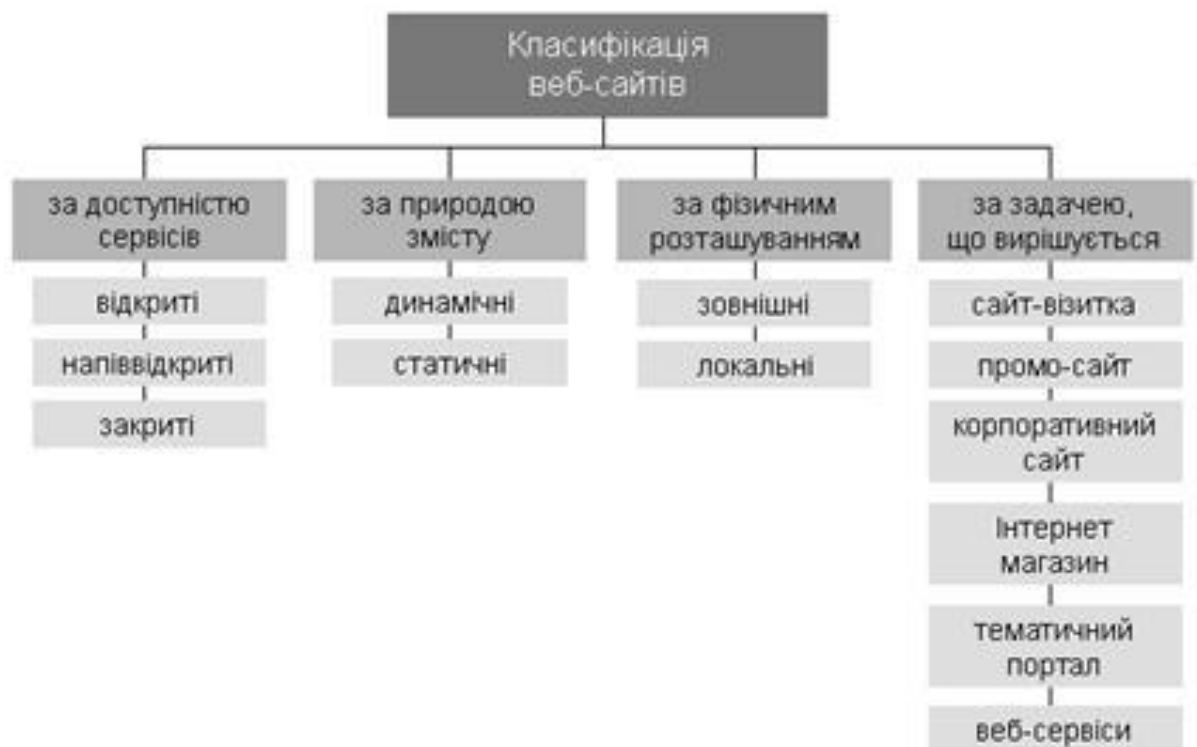


Рисунок 1.1 – Класифікація web-порталів

Розглянемо більш докладніше класифікацію web-додатків.

За доступністю сервісів:

а) відкриті - всі сервіси повністю доступні для будь-яких відвідувачів та користувачів;

б) напіввідкриті - для доступу необхідно зареєструватися;

в) закриті - повністю закриті портали організацій, особисті додатки приватних осіб.

За природою змісту:

а) статичні - весь вміст заздалегідь готується. Користувач видають файли в тому вигляді, в якому вони зберігаються на сервері;

б) динамічні - вміст генерується спеціальними скриптами (програмами) на основі других даних з будь - якого джерела.

За фізичним розташуванням:

а) зовнішні ресурси мережі Інтернет – забезпечують доступ будь-якому користувачу мережі;

б) локальні ресурси - доступні тільки в межах локальної мережі. Це можуть бути як корпоративні портали організацій, так і ресурси приватних осіб в локальній мережі провайдера.

За задачею, що вирішується ресурсом:

а) додаток-візитка - містить загальні дані про власника ресурсу, вид діяльності, історію, прайс-листи, контактні дані, реквізити, схему проїзду;

б) корпоративний додаток - містить повну інформацію про компанію-власника, її послуги або продукцію, події в житті компанії. Відрізняється від додатку-візитки повнотою наданої інформації, часто містить різні функціональні інструменти для роботи з контентом (пошук і фільтри, календарі подій, фотогалереї, корпоративні блоги, форуми). Може бути інтегрований з внутрішніми інформаційними системами компанії власника. Може містити закриті розділи для тих чи інших груп користувачів - співробітників, дилерів, контрагентів тощо;

в) інтернет-магазин - web-додаток з каталогом продукції, за допомогою якого клієнт може замовити потрібні йому товари;

г) промо-додаток - ресурс про конкретну торгову марку або продукт, на таких ресурсах розміщується вичерпна інформація про бренд, різні рекламні акції (конкурси, вікторини, ігри і тощо);

д) тематичний портал - це дуже великий web-ресурс, який надає вичерпну інформацію з певної тематики. Портали містять засоби взаємодії з користувачами і дозволяють користувачам спілкуватися в рамках порталу (форуми, чати);

ж) Web-сервіс – послуга, що створена для виконання будь-яких завдань в рамках мережі Інтернет. Прикладами таких сервісів можуть бути наступні: дошка оголошень, каталоги додатків, пошукові сервіси, поштові сервіси, форуми, блоги, портали обміну файлами, портали соціальних мереж тощо.

1.4 Склад інформаційного ресурсу

Структура web-додатку - це його каркас, який визначає порядок навігації[2].

Базові структури web-додатків можна розділити на три групи: лінійна, деревоподібна, решітчаста.

Лінійна структура являє собою послідовність web-сторінок, доступ до яких можливий тільки з попередньої і наступної. Така структура може бути застосована для іміджевих ресурсів, додатків-презентацій, онлайн-навчальних посібників.

Деревоподібна структура є ієрархічною, де головна сторінка є першим, самим верхнім рівнем, сторінки категорій - нижчим, другим рівнем, а кінцеві сторінки, відповідно - третім. У ряді випадків рівнів може бути більше, якщо категорії містять під категорії, але велика кількість рівнів негативно позначається на індексації сторінок ресурсу. Дана структура підходить практично для будь-якого ресурсу. Це може бути тематичний портал, інтернет-магазин.

Решітчаста структура ресурсу дозволяє здійснювати переходи, як по вертикалі, так і по горизонталі між гілками в різних рівнях. Ця структура застосовується, як правило в додатках-каталогах.

Web-додатки різних структур в основному включають в себе чотири типи сторінок [3]:

- а) головна сторінка;
- б) сторінка категорій;
- в) кінцева сторінки;
- г) службові сторінки.

1.5 Web-хостинг

Web-хостинг - це спосіб розміщення додатку в мережі інтернет. Як тільки ви розмістили свій ресурс на сервері - хто завгодно може отримати доступ до нього, набравши доменне ім'я в рядку браузера. Доступ до нього можливий 24 години на добу, 7 днів на тиждень, 365 днів на рік.

Слово "хостинг", має кілька значень: господар, основний пристрій, головний комп'ютер, утримувач заїжджого двору. Значення "утримувач заїжджого двору" найближче до правильного тлумачення терміну "хостинг" в Інтернеті, тому що мається на увазі, що ваша web-сторінка розміщена на комп'ютері хостингу провайдера Інтернету поряд з безліччю web-сторінок інших людей

Для того, щоб розмістити свій web-додаток в мережі, необхідно:

а) перш за все, мати власний web-додаток. Потрібно мати копію ресурсу на локальному комп'ютері (в html-файлах), або ж готові матеріали + скрипт, який дозволяв би створити web-додаток безпосередньо на сервері;

б) доменне ім'я. Потрібно знайти і придбати доменне ім'я ресурсу. Бажано, щоб ім'я говорило про тематику ресурсу, і легко запам'ятовувалося. Доменне ім'я може бути в будь-якій, на ваш вибір, доменній зоні, наприклад: .com .net .org .ru і т.д. При виборі доменної зони керуйтеся даними про тематиках доменів. Домени .com, наприклад, призначені для порталів комерційної спрямованості, .org - державних установ і громадських організацій. Регіональні домени говорять про приналежність ресурсу якому-небудь регіону. Це необов'язкове правило, але варто про це пам'ятати.

в) замовити один з хостинг-планів в компанії, що здійснює послуги web-хостингу.

Сервер - це комп'ютер, розміщений у провайдера інтернету і підключений до всесвітньої мережі, на якому зберігається копія web-додатку. Як правило, один сервер вміщує чимало web-додатків, за рахунок великого дискового простору.

Але в цілому, це не має значення. Важливий не обсяг сервера, а його обчислювальні потужності і пропускна здатність інтернет-каналів провайдера.

За допомогою будь-якого web-браузера (програми для перегляду web-додатків в інтернеті) можна отримати доступ до свого ресурсу, набравши в рядку його адресу.

Web-хостинг можна порівняти з орендою нерухомості. Безліч компаній готові запропонувати хостинг за помірну плату, однак мало хто може забезпечити хостингом належної якості. Говорячи про вартість хостингу, потрібно відзначити, що для більшості web-додатків вартість плати на місяць може не перевищувати 100 гривень. В межах цієї вартості можна знайти хостинг-компанію, яка надає близько 1000 мегабайт дискового простору на сервері і всілякі функції, такі як: установка і запуск скриптів, підтримка баз даних, адміністративна панель (тобто ваша панель управління) web-додатку, де робляться настройки не тільки ресурсу, але і електронної пошти.

Плануючи запуснути свій ресурс в мережі, потрібно не тільки вирішити, якого рівня буде web-додаток, але і правильно визначитися з вибором провайдера хостингу. Статистично, більше 70% власників порталів переходили з одного web-хостингу на інший хоча б один раз. Перехід до іншого провайдера був викликаний незадоволеністю якістю послуг, що надаються web-хостингом.

Щоб не зробити помилок у виборі web-хостингу, необхідно відвідати форуми і рейтингові ресурси, де обговорюються хостери. Уважно почитати думки клієнтів про різних хостинг-провайдерів. Час, витрачений на вивчення форумів і читання відгуків, окупиться стабільністю роботи ресурсу.

1.6 WAMP комплекси та їх характеристика

WAMP - акронім від «Windows, Apache, MySQL та PHP»[4].

Даним скороченням описується комплектація пакетів програм, під відповідну ОС, що позначається першою літерою. Походить від LAMP, де перша

буква означає Linux. Існує також MAMP, під Mac OS. Дані пакети використовуються для web-розробки.

Системи WAMP, MAMP і LAMP поставляються в формі пакетів, що зв'язують упаковані програми таким чином, щоб їх не потрібно було встановлювати і налаштовувати окремо. Це означає, що потрібно просто завантажити і встановити одну програму, і слідувати простим підказкам, щоб підготувати сервер для розробки, і запустити його в найкоротші терміни та мінімальними зусиллями.

1.6.1 Denwer

Денвер - набір дистрибутивів (локальний сервер WAMP) і програмна оболонка, призначені для створення і налагодження web-додатків, іншого динамічного вмісту інтернет-сторінок на локальному ПК (без необхідності підключення до мережі Інтернет) під керуванням ОС Windows [5].

Відразу після установки доступний повністю працюючий web-сервер Apache, що працює на локальному комп'ютері, на якому може працювати необмежену кількість порталів, що дуже ефективно для розробки і налагодження сценаріїв PHP без завантаження його файлів на віддалений сервер. Для запуску практично всіх утиліт «Денвера» використовується додаток Run в підкаталозі /denwer кореневого каталогу установки «Денвера». При запуску створюється віртуальний диск (за замовчуванням Z :), де зберігаються всі файли проектів.

Особливістю, що відрізняє Denwer від інших WAMP-дистрибутивів, є автоматична правка системного файлу hosts, що є локальним аналогом DNS-сервера, що дозволяє звертатися до локальних додатків, які працюють під управлінням Денвера, по іменах, що збігається з імям папки, розташованої в каталозі /home Денвера.

До базового пакету входять:

а) Web-сервер Apache з підтримкою SSI, SSL, mod_rewrite, mod_php;

- б) Інтерпретатор PHP з підтримкою GD, MySQL, SQLite;
- в) СУБД MySQL з підтримкою транзакцій (mysqld-max);
- г) Система управління віртуальними хостами, заснована на шаблонах;
- д) Система управління запуском і завершенням;
- ж) Панель phpMyAdmin для адміністрування СУБД;
- и) Ядро інтерпретатора Perl без стандартних бібліотек (поставляються окремо);
- к) Емулятор sendmail і сервера SMTP з підтримкою роботи спільно з PHP, Perl, Parser і ін.;

л) Установник.

До пакетів розширень входять:

- а) Повна версія ActivePerl 5.8;
- б) Інтерпретатор PHP версії 5 з повним набором модулів;
- в) Інтерпретатор Python;
- г) СУБД MS SQL, PostgreSQL або InterBase / FireBird версій 1 і 2;
- д) Інтерпретатор PHP версії 3 або 4;
- ж) Інтерпретатор Parser;
- и) Виправлення в CONFIGURATION;

1.6.2 EasyPHP

EasyPHP - готова до використання збірка для PHP-розробників.

Для цього програмного продукту можлива установка користувачем Portable версії на USB-накопичувач. Підтримка модулів-розширень таких як WordPress (Блог), SPIP (англ.), Drupal, Joomla (CMS-системи), Prestashop (інтернет-магазин), phpBB, Phorum (Веб-форуми, в процесі розробки) [5].

EasyPHP 16.1.1 включає в себе:

- а) PHP 7.x / 5.6.x / 5.5.x / 5.4.x;
- б) PhpMyAdmin 4.6.1;

- в) Apache 2.4.20;
- г) Nginx 1.9
- д) MySQL 5.7.11;
- ж) Xdebug 2.4.0.

У версії 14.1 відсутня підтримка російської мови.

Досить проста і нічим не примітна збірка. Нам пропонують звичайне трей-меню без надмірностей, нехитрі налаштування і заплутану стартову сторінку без російської мови (версія 14.1).

1.6.3 Open Server

Open Server Panel - це портативна серверна платформа і програмне середовище, створена спеціально для web-розробників з урахуванням їх рекомендацій і побажань.

Програмний комплекс має багатий набір серверного програмного забезпечення, зручний, багатофункціональний продуманий інтерфейс, має потужні можливості з адміністрування та налаштування компонентів. Платформа широко використовується з метою розробки, налагодження і тестування web-проектів, а так само для надання web-сервісів в локальних мережах.

Хоча спочатку програмні продукти, що входять до складу комплексу, не розроблялись спеціально для роботи один з одним, така зв'язка стала дуже популярною серед користувачів Windows, в першу чергу через те, що вони отримували безкоштовний комплекс програм з надійністю на рівні Linux серверів.

Зручність і простота управління безумовно не залишать вас байдужими, за час свого існування Open Server зарекомендував себе як першокласний і надійний інструмент необхідний кожному web-майстру [5].

До програмного комплексу входить:

- а) OSPanel 5.2.6;
- б) Apache 2.2.31 / 2.4.25;

- в) Bind 9.11;
- г) Nginx 1.6.3 / 1.8.1 / 1.10.2 / 1.11.7;
- д) MySQL 5.1.73 / 5.5.53 / 5.6.34 / 5.7.16;
- ж) MariaDB 5.5.53 / 10.0.28 / 10.1.19;
- и) MongoDB 2.4.14 / 2.6.12 / 3.0.14 / 3.2.11 / 3.4.0;
- к) Memcached 1.2.6 / 1.4.5;
- л) FTP FileZilla 0.9.59;
- м) PHP 5.2.17/5.3.29/5.4.45/5.5.38/5.6.29/7.0.14/7.1.00;
- н) ImageMagick 6.8.9-9-Q16;
- о) Sendmail 32;
- п) PHPMyAdmin 4.6.5.2;
- р) PHPMemcachedAdmin 1.2.2;

1.6.4 WampServer

WampServer це платформа для web-розробки під Windows для динамічних web-додатків за допомогою сервера Apache2, інтерпретатора скриптів PHP і бази даних MySQL. У нього також входить web-додаток PHPMyAdmin для найбільш простої обробки баз даних.

Має автоматичний інсталятор. Для управління сервером і його налаштуваннями WampServer створює іконку в треї. Дозволяє встановити будь-яку версію Apache, MySQL та PHP в якості доповнення [5].

Компоненти:

- а) Apache HTTP Server;
- б) PHP;
- в) MySQL;
- г) PhpMyAdmin;

Можливості WampServer:

- а) керувати налаштуваннями Apache і MySQL;

- б) змінювати статус "в мережі", "не в мережі" (доступно кожному користувачу або ж лімітовано на локальному хості);
- в) встановлювати або змінювати версію Apache, MySQL та PHP;
- г) керувати параметрами конфігурацій серверів;
- д) отримувати доступ до логів;
- ж) отримувати доступ до файлів конфігурацій;

1.7 Постановка технічного завдання

В даній дипломній роботі необхідно розробити ІКС футбольного клубу.

Метою даної ІКС буде спрощення отримання вболівальниками потрібної їм інформації про їх улюблений футбольний клуб. Проект може називатися комерційним, прибуток очікується від продажу рекламних місць на ресурсі.

Функціональні вимоги до ресурсу:

а) головна сторінка повинна відображати новини футбольного клубу Шахтар Донецьк, у скороченому вигляді. При потребі користувача, він повинен отримати більш детальну інформацію за кожною новиною;

б) бокова панель ресурсу повинна відображати таблиці зі статистичними показниками Чемпіона України з футболу, а саме: таблицю найкращих бомбардирів ліги, таблицю найкращих асистентів ліги;

в) хедер ресурсу має містити меню всього ресурсу, логотип ресурсу та заголовок;

г) футер ресурсу має містити посилання на сторінку зворотнього зв'язку користувача з адміністрацією ресурсу. Посилання на сторінку, що відображає інформацію про розробника ресурсу, та гіперпосилання на сторінку розробника;

д) сторінка "команда" має містити таблицю з усіма гравцями команди Шахтар Донецьк, таблиця повинна мати функцію сортування, для того, щоб користувач міг отримати інформацію про найдорожчого гравця команди,

найкращого бомбардира, гравця котрий зіграв найбільше матчів в поточному сезоні;

ж) сторінка "матчи" повинна містити інформацію про всі матчі команди в поточному футбольному сезоні, також вона повинна мати функцію адаптивної фільтрації, завдяки якій користувач зможе ввести назву команди супротивника, або дату матчу який його цікавить, та таблиця має від фільтруватися за цими даними;

ж) сторінка "календарь туров" повинна надавати користувачу детальну інформацію про тур чемпіонату з футболу України. Номер туру повинен вводити користувач;

и) сторінка "фотоальбом" має містити галерею з фотографіями футболістів та тренера команди Шахтар Донецьк. Фотогалерея повинна відображати мініатюри фотографій, які при натисканні будуть розвертатися у повний розмір.

к) повинна бути створена База даних для ІКС.

ІКС повинна бути розроблена з використанням таких засобів web-програмування, які дозволять їй коректно відображатися та функціонувати на різних платформах.

1.8 Висновки

В якості WAMP пакету програм для розробки Web-додатку був обраний Denwer. Ми бачимо мізерний набір не надто свіжих компонентів, чорне вікно і ніякого GUI. До плюсів цього пакета можна віднести його гранично малий розмір і хорошу «базу знань» на форумі, накопичену за багато років. Фактично управління сервером здійснюється через виконання ".bat" скриптів по запуску і зупинці модулів, як такої керуючої програми немає взагалі.

Denwer гранично простий і зрозумілий, а тому має найбільшу армію шанувальників серед «новачків». Денвер виконує основні завдання.

Розвиток Інтернет технологій надає нові можливості по взаємодії футбольного клубу як організації з вболівальниками та користувачами інтернету.

Розроблюваний ресурс повинен бути орієнтований на поліпшення отримання інформації про футбольний клуб користувачами інтернету та вболівальниками.

Після входу на інформаційний ресурс, користувач повинен отримати доступ до широкого спектру інформації:

- а) ознайомлення з новинами організації представленої на порталі;
- б) перегляд статистичних показників чемпіонату України з футболу;
- в) перегляд інформації про склад футбольної команди "ФК Шахтар Донецьк";
- г) перегляд таблиці всіх матчів футбольного сезону команди "ФК Шахтар Донецьк";
- д) форма зворотнього зв'язку з адміністрацією ресурсу.

ІКС не повинна забезпечувати збереження конфіденційності інформації - доступ до виключно інформаційного ресурсу надано всім користувачам інтернету.

Таким чином, ресурс буде забезпечувати користувачів інтернету потрібною їм інформацією у вільному доступі.

2 ВИБІР І ОБҐРУНТУВАННЯ ПРОГРАМНИХ ЗАСОБІВ ДЛЯ РОЗРОБКИ ІНФОРМАЦІЙНОГО РЕСУРСУ

2.1 HTML

HTML - мова розмітки (маркування) гіпертексту. Гіпертекст своїм розвитком зобов'язаний інтернету, хоч і створювався він зовсім не для того. HTML дає можливість виробляти перехід від однієї частини тексту до іншої, і, що чудово, ці частини можуть зберігатися на абсолютно різних комп'ютерах [6].

HTML не варто плутати з мовами програмування, він створений спеціально для розмітки web-сторінок. Саме мова розмітки дає браузеру необхідні інструкції про те, як відображати тексти та інші елементи сторінки на моніторі. Важливо зауважити, що не тільки різні браузери, але і різні їх версії можуть по-різному сприймати і відображати на екрані код. Отже, деякі елементи коректно виглядають в браузері Opera можуть виглядати інакше в Internet Explorer і інших браузерах.

Код розмітки в HTML складається з так званих «тегів». Теги надають інформацію браузерам про форматування і розмітки сторінки. Назва тега полягає в кутові дужки «<» і «>». Всі теги відкриваються і закриваються, наприклад, тег ``, який робить текст жирним.

Ці теги впливають на зміст всередині нього. Дані, що знаходяться між такими символами, називаються вмістом тега. Іноді зустрічаються теги, які одночасно і закриваються і відкриваються, наприклад `<hr>` - лінія.

Редактори - HomeSite і DreamWeaver. Вони допоможуть створити як додаток-візитку, так і справжній шедевр сучасного мистецтва. До мінусів цього ПЗ можна віднести: довгу обробку масивного коду і невелике засмічення коду зайвими тегами. Потрібне знання HTML в цьому випадку на порядок-два вище, ніж при роботі з FrontPage.

Для роботи з html сторінками у дипломному проекті використано програму AkelPad, також непоганою програмою - редактором є Notepad ++.

2.2 PHP

PHP - мова створення сценаріїв, яка давно переросла свою назву. Справа в тому, що PHP - це аббревіатура від слів Personal Home Page. Перша версія PHP була створена Расмусом Лердорфом в 1994 році і представляла собою набір інструментів для відстеження відвідувачів Web-сторінки. Згодом PHP з набору інструментів перетворився в повноцінну мову програмування, а його назва була змінена як рекурсивне освіту PHP HyperText Preprocessor (препроцесор гіпертексту PHP) [9].

PHP - це серверна мова створення сценаріїв. Конструкції PHP, вставлені в HTML-текст, виконуються сервером при кожному відвідуванні сторінки. Результат їх обробки разом зі звичайним HTML-текстом передається браузеру. В даний час основною версією PHP є п'ята.

Існують два основні конкуренти PHP: Active Server Pages (ASP) компанії Microsoft і ColdFusion компанії Allaire. У порівнянні з ними PHP має низку переваг, серед яких:

- а) висока продуктивність;
- б) функціональність. Розробку PHP-програми можна відокремити від власне розробки Web-сторінки, що спростить життя і програмісту, і дизайнеру;
- в) PHP є абсолютно безкоштовним;
- г) простота у використанні;
- д) переносимість. Один і той же PHP-код можна використовувати як в середовищі NT, так і на платформах UNIX.

PHP - мова програмування, спеціально розроблена для написання web-додатків (скриптів, сценаріїв), що виконуються на Web-Сервері. Синтаксис мови багато в чому ґрунтується на синтаксисі C, Java і Perl. Він дуже схожий на C и на

Perl, тому для професійного програміста не важко буде його вивчити. З іншого боку, мова PHP простіше, чим C, і його може освоїти web-майстер, що не знає поки інших мов програмування.

Величезним плюсом PHP є те, що PHP-скрипти виконуються на стороні сервера. PHP не залежить від швидкості комп'ютера користувача або його браузера, він повністю працює на сервері. Користувач навіть може не знати, чи отримує він звичайний Html-файл або результат виконання скрипта.

Сценарії мовою PHP можуть виконуватися на сервері у вигляді окремих файлів, а можуть інтегруватися в html сторінки.

PHP здатний генерувати й перетворювати не тільки HTML документи, але й зображення різних форматів - JPEG, GIF, PNG, файли PDF і FLASH. PHP здатний формувати дані в будь-якому текстовому форматі, включаючи XHTML і XML.

PHP - кросплатформенна технологія. Дистрибутив PHP доступний для більшості операційних систем, включаючи Linux, багато модифікацій Unix, Microsoft Windows, Mac OS і багатьох інших. PHP підтримується на більшості web-серверів, таких, як Apache, Microsoft Internet Information Server (IIS), Microsoft PersonalWeb Server і інших.

Для більшості серверів PHP поставляється в 2-х варіантах - у якості модуля й у якості CGI препроцесора.

PHP підтримує роботу з ODBC і велику кількість баз даних: Mysql, MSQL, Oracle, Postgresql, Sqlite і ін.

Мова програмування PHP, особливо у зв'язці з найпопулярнішою базою даних MySQL - найоптимальніший варіант для створення інформаційного ресурсу різної складності.

За допомогою мови PHP буде реалізовано частину функцій інформаційного ресурсу.

2.3 jQuery

jQuery - бібліотека JavaScript, що фокусується на взаємодії JavaScript і HTML. Бібліотека jQuery допомагає легко отримувати доступ до будь-якого елемента DOM, звертатися до атрибутів і вмісту елементів DOM, маніпулювати ними. Також бібліотека jQuery надає зручний API для роботи з AJAX [10].

Можливості jQuery:

а) движок кросбраузерних CSS-селекторів Sizzle, що виділився в окремий проект;

б) перехід по дереву DOM, включаючи підтримку XPath як плагіна;

в) події (с рус. пер. - "События");

г) візуальні ефекти;

д) AJAX доповнення;

ж) JavaScript плагіни.

jQuery відокремлює поведінку від структури HTML. Наприклад, замість прямої вказівки на обробник події натискання кнопки, управління передається jQuery, яка ідентифікує кнопки і потім перетворює його в обробник події кліка. Такий поділ поведінки і структури також називається принципом ненав'язливого JavaScript.

Бібліотека jQuery містить функціональність, корисну для максимально широкого кола завдань. Проте, розробниками бібліотеки не ставилося завдання суміщення в jQuery функцій, які підійшли б усюди, оскільки це призвело б до великого коду, велика частина якого не затребувана. Тому була реалізована архітектура компактного універсального ядра бібліотеки та плагінів. Це дозволяє зібрати для інформаційного ресурсу саме ту JavaScript-функціональність, яка на ньому була б затребувана.

jQuery, як правило, включається в web-сторінку як один зовнішній JavaScript-файл:

```
<Head>
```

```
<Script src = "jquery-2.2.2.min.js">
</ Script>
</ Head>
```

За допомогою jQuery у дипломному проєкті будуть реалізовані важливі функції:

- а) адаптивна фільтрація сторінки "матчі";
- б) сортування таблиці "команда".

2.4 AJAX

За допомогою JavaScript Ви можете створювати асинхронні запити і відправляти їх на сервер.

Використання асинхронних запитів дозволяє значно прискорити завантаження сторінок, так як в цьому випадку оновлюватися буде тільки та частина сторінки, яка містить нові дані, а не сторінка цілком.

Техніка використання асинхронних запитів називається AJAX - Asynchronous JavaScript And XML (Асинхронний JavaScript і XML) [11].

Створення AJAX запитів на "чистому" JavaScript має кілька недоліків:

- а) код навіть самого простого AJAX запиту виходить досить громіздким і складним для розуміння без спеціального ознайомлення;
- б) необхідно додавати додатковий код для підтримки старих версій браузерів.

AJAX запит створений за допомогою jQuery займає всього один рядок коду, і вже оптимізований для використання і з новими, і старими версіями браузерів.

Синтаксис:

```
$("селектор").load(url,дані,функція)
```

Селектор - вибирає елемент, в якому буде відображений результат запиту.

URL - адресу файлу, який буде запитаний у сервера за допомогою AJAX.

Дані - які будуть передані запитом файлу. Якщо ви хочете запросити файл, при цьому не передавати ніяких даних, залиште цей параметр порожнім.

Функція - ім'я функції, яка буде викликана після виконання запиту.

Приклад:

```
$ (Document) .ready (function () {  
    $ ("# But1"). Click (function () {  
        $ ("# Par1"). Load ("testfile.txt");  
    })  
});
```

Супроводжуючі функції дозволяють виконувати різний код в залежності від стану виконання AJAX запиту:

Метод `ajaxSend ()` виконує переданий в нього код під час відправлення AJAX запиту.

Метод `ajaxComplete` виконує переданий в нього код після завершення виконання AJAX запиту (при цьому неважливо сталася помилка чи ні).

Метод `ajaxSuccess` виконує переданий в нього код, якщо виконання AJAX запиту завершується успішно.

Метод `ajaxError` виконує переданий в нього код, якщо виконання AJAX запиту завершується з помилкою.

Методи `ajaxStart` і `ajaxStop` використовуються для супроводу групи AJAX запитів.

Метод `ajaxStart` виконує переданий в нього код при відправленні першого запиту з групи.

Метод `ajaxStop` виконує переданий в нього код при завершенні останнього запиту з групи.

Високо-рівневі методи створення AJAX запитів підходять для створення більшої частини запитів, але іноді їх функціональності стає недостатньо.

Низько-рівневі AJAX запити надають ширшу функціональність, але більш складні у використанні.

2.5 Бази даних

Програмування web-додатків виконується з метою створення деякого корисного функціоналу. Робота з базами даних - це одна з найважливіших складових програмування додатків динамічного типу. Будь то формування сторінок «на льоту» або ж реагування на дії відвідувачів ресурсу - взаємодія з базами даних потрібна завжди [8].

Бази даних для ІКС (БД) використовуються з метою зберігання різної інформації і, спрощено, представляють собою деякий набір взаємозв'язаних таблиць. Розміри таблиць в БД різні, їх кількість довільна. Саме в базах даних зберігається на сервері необхідна для роботи ресурсу інформація, наприклад, інформація про клієнтів, каталог товарів, статистичні дані і т. д.

Програмування додатків динамічного типу виконується за допомогою різних скриптів, поділених зазвичай на серверні і клієнтські. Програмування додатків за допомогою серверних скриптів дозволяє обробляти дані, введені відвідувачами ресурсу в web-форми, генерувати динамічні сторінки, відсилати і приймати cookies. Для отримання інформації, необхідної при виконанні подібних дій, серверні скрипти звертаються до баз даних. Звернення скрипта до БД називається запитом.

Для побудови запитів до баз даних широко застосовується SQL (Structured Query Language) - «мова структурованих запитів». За допомогою SQL може здійснюватися додавання, видалення, редагування записів в таблицях баз даних, вибірка даних відповідно до різних умов, сортування даних і багато іншого.

У програмуванні додатків управління БД здійснюється за допомогою клієнт-серверних систем управління базами даних (СКБД), таких як Oracle, MS SQL Server, PostgreSQL, MySQL та ін. Клієнт-серверні СУБД обробляють запити централізовано, до їх переваг відносять забезпечення високої надійності баз даних, високої доступності та високої безпеки.

СУБД MySQL - вільна система управління базами даних, одна з найбільш часто вживаних в програмуванні інформаційних ресурсів. СУБД MySQL підтримує велику кількість існуючих типів таблиць (InnoDB, MyISAM і т.д.), а завдяки відкритій архітектурі і GPL-ліцензуванню, в СУБД MySQL постійно з'являються нові типи таблиць. Управління базами даних за допомогою MySQL дуже зручно, що зробило цю систему затребуваною і популярною.

Система керування базами даних Microsoft SQL Server поставляється компанією Microsoft на комерційній основі (за винятком безкоштовної редакції Express Edition). Дана СУБД використовує мову запитів Transact-SQL, підтримується операційними системами сімейства Windows Desktop/Server. В СУБД Microsoft SQL Server присутнє графічне ПО для конструювання та оптимізації запитів (SQL Management Studio та Studio Express).

Об'єктно-реляційна система управління базами даних компанії Oracle - Oracle Database - працює на Windows, Unix, Linux, MacOS. Oracle Database на відміну від MySQL, наприклад, має більш широку сферу застосування. СУБД Oracle має високу продуктивність, широкий функціонал, унікальні технології (RAC, RAC і т.д.). У програмуванні ресурсу для невеликих і середніх компаній застосовується досить рідко через свою високу вартість. До того ж, досить складно знайти хостинг з підтримкою даної СУБД.

Вільна система управління базами даних PostgreSQL існує в редакціях для Linux, Solaris/OpenSolaris, Win32, Win x86-64, Mac OS X, FreeBSD, QNX 4.25, QNX 6. Базується на мові SQL. Серед переваг PostgreSQL виділяють підтримку БД практично необмеженого розміру, наявність надійних механізмів реплікації, легку розширюваність, підтримку великого набору вбудованих типів даних і багато іншого.

Програмування ресурсу, що взаємодіють по-різному з базами даних, включає кілька основних етапів роботи з БД: побудова запитів до БД за допомогою мови SQL, програмування сценаріїв для обробки цих запитів і програмування модулів для відображення результатів обробки запитів.

Надмірне число звернень від ресурсу до БД робить завантаження додатку повільнішим, збільшує навантаження на сервер. В результаті можливі збої в роботі додатку, аж до повного припинення доступу. Зменшення кількості запитів до БД дозволяє зменшити навантаження на сервер, а також зменшити час завантаження динамічних сторінок з сервера. Тому оптимізація взаємодії ресурсу з базами даних - це одне із завдань професійного програмування додатку.

2.5.1. MySQL

MySQL - це популярна система керування базами даних (СКБД), дуже часто застосовується в поєднанні з PHP.

База даних являє собою структуровану сукупність даних. Ці дані можуть бути будь-якими, від простого списку майбутніх покупок до переліку експонатів картинної галереї або величезної кількості інформації в корпоративній мережі. Для запису, вибірки і обробки даних, що зберігаються в комп'ютерній базі даних, необхідна система управління базою даних, якою і є ПО MySQL. Оскільки комп'ютери чудово справляються з обробкою великих обсягів даних, управління базами даних відіграє центральну роль в обчисленнях. Реалізовано таке управління може бути по-різному, як у вигляді окремих утиліт, так і у вигляді коду, що входить до складу інших додатків.

MySQL - це система керування базами даних. У реляційній базі даних дані зберігаються не всі скопом, а в окремих таблицях, завдяки чому досягається вигреш в швидкості і гнучкості. Таблиці зв'язуються між собою за допомогою відносин, завдяки чому забезпечується можливість об'єднувати при виконанні запиту дані з декількох таблиць. SQL як частина системи MySQL можна охарактеризувати як мову структурованих запитів плюс найбільш поширений стандартний мова, яка використовується для доступу до баз даних.

MySQL - це ПЗ з відкритим кодом. Таке ПЗ можна одержувати з інтернету і використовувати безкоштовно. При цьому кожен користувач може вивчити

вихідний код і змінити його відповідно до своїх потреб. Використання програмного забезпечення MySQL регламентується ліцензією GPL, в якій зазначено, що можна і чого не можна робити з цим програмним забезпеченням в різних ситуаціях.

MySQL є системою клієнт-сервер, яка містить багато-потоківий SQL-сервер, що забезпечує підтримку різних обчислювальних машин баз даних, а також кілька різних клієнтських програм і бібліотек, засоби адміністрування і широкий спектр програмних інтерфейсів (API).

Для некомерційного використання MySQL є безкоштовна. Можливості сервера MySQL:

- а) простота у встановленні та використанні;
- б) підтримується необмежена кількість користувачів, що одночасно працюють із БД;
- в) кількість рядків у таблицях може досягати 50 млн .;
- г) висока швидкість виконання команд;
- д) наявність простої і ефективної системи безпеки.

2.6 Зв'язок клієнтської й серверної частин інформаційного ресурсу

MySQL складається з двох частин: серверної і клієнтської.

Сервер MySQL постійно працює на комп'ютері. Клієнтські програми (наприклад, скрипти PHP) посилають серверу MySQL SQL-запити за допомогою мережеских засобів, сервер їх обробляє і запам'ятовує результат. Тобто скрипт (клієнт) вказує, яку інформацій він хоче отримати від сервера баз даних. Потім сервер баз даних посилає відповідь (результат) клієнту (скрипту). Але передається не весь результат. Справа в тому, що розмір результуючого набору даних може бути занадто великим, і на його передачу по мережі піде надто багато часу. Та й рідко коли буває потрібно отримувати відразу весь запит (тобто всі записи, що задовольняють висловом запити). Наприклад, нам може знадобитися лише

підрахувати, скільки записів задовольняє того чи іншого умові, або ж вибрати з даних тільки перші 5 записів. Механізм використання сокетів має на увазі технологію клієнт-сервер, а це означає, що в системі повинна бути запущена спеціальна програма - MySQL-сервер, яка приймає і обробляє запити від програм. Так як вся робота відбувається в дійсності на одній машині, накладні витрати по роботі з мережевими засобами незначні. Установка і підтримка з'єднання з MySQL-сервером обходиться досить дешево.

Структура MySQL трирівнева: бази даних - таблиці - записи. Бази даних і таблиці MySQL фізично представляються файлами з розширеннями .frm, .MYD, .MYI. Логічно - таблиця являє собою сукупність записів. А записи - це сукупність полів різного типу. Ім'я бази даних MySQL унікально в межах системи, а таблиці - в межах бази даних, поля - в межах таблиці. Один сервер MySQL може підтримувати відразу кілька баз даних, доступ до яких може розмежовуватися логіном і паролем. Знаючи ці логін і пароль, можна працювати з конкретною базою даних. Наприклад, можна створити або видалити в ній таблицю, додати записи і т.д.. Ім'я-ідентифікатор і пароль призначаються на локальному сервері Denwer, які і забезпечують підтримку MySQL для свого користувача.

2.7 Проектування Бази даних для інформаційного ресурсу

Для проектування Бази даних ІКС було використано інтернет ресурс. Він надає повні можливості при проектуванні бази даних, такі як:

- а) створення таблиць;
- б) створення полів для таблиці;
- в) створення зв'язків між таблицями;
- г) встановлення первинного ключа таблиць;
- д) вибір типу полей.

При проектуванні Бази даних було створено три таблиці:

- а) calendar;

б) rezultat;

в) news.

Для першої таблиці "calendar" було створено такі поля:

а) "nomer_tura";

б) "data_tura".

Первинним ключем було обрано поле "nomer_tura".

Для другої таблиці "reaultat" було створено такі поля:

а) "nomer_tura";

б) "nomer_igry";

в) "com_1";

г) "com_2";

д) "mesto_prov";

ж) "sudja";

и) "schet".

Первинним ключем було обрано поле "nomer_igry".

Таблиця "calendar" являється головною та має відношення 1:М з підлеглою таблицею "reaultat".

Для третьої таблиці було створено такі поля:

а) "id";

б) "title";

в) "intro_text";

г) "full_text";

Графічне представлення структури бази даних (рис. 2.1):

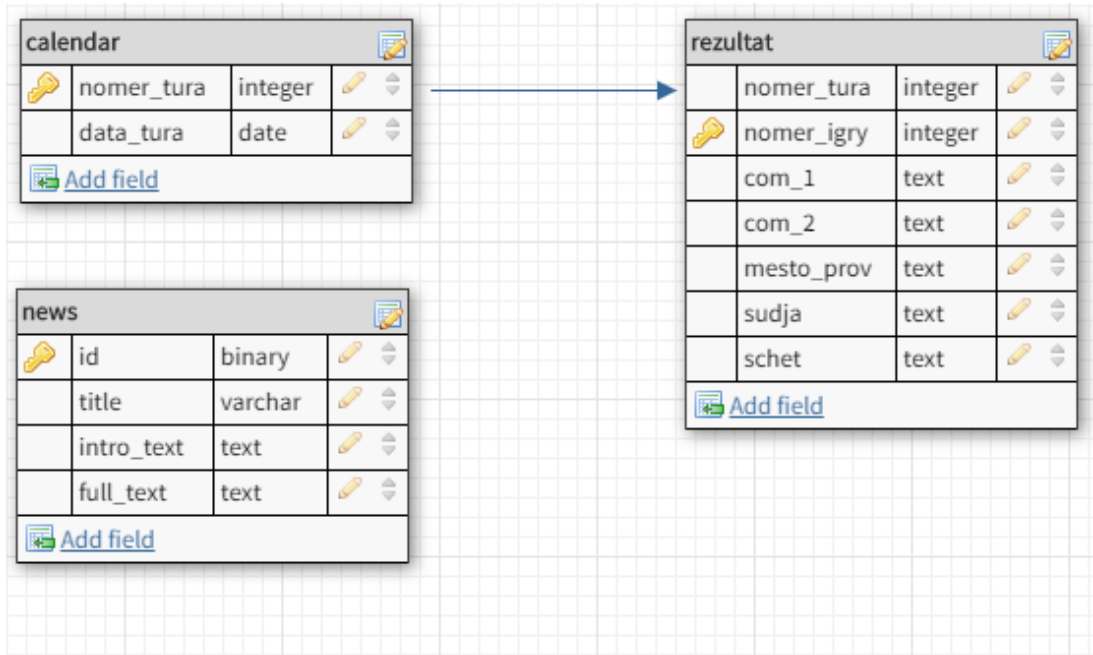


Рисунок 2.1 - Проектування Баз даних

2.8 Висновки

На даний час, бази даних є невід'ємною частиною при проектуванні і розробці web-додатків. Їх використання дозволяє домогтися кращих результатів функціонування web-додатків. Для розробки ресурсу на локальному хостингу Denwer було обрано СУБД MySQL. MySQL є дуже швидкою, надійною і легкою у використанні. Саме ці якості потрібні для реалізації поставленої задачі. MySQL володіє також рядом зручних можливостей, розроблених в тісному контакті з користувачами. Спочатку сервер MySQL розроблявся для управління великими базами даних з метою забезпечити більш високу швидкість роботи в порівнянні з існуючими на той момент аналогами. Уже протягом кількох років даний сервер успішно використовується в умовах промислової експлуатації з високими вимогами. Незважаючи на те що MySQL постійно вдосконалюється, він уже сьогодні забезпечує широкий спектр корисних функцій. Завдяки своїй доступності, швидкості і безпеці MySQL дуже добре підходить для доступу до баз даних у Internet.

3 РОЗРОБКА ІНФОРМАЦІЙНОГО РЕСУРСУ

3.1 Створення бази даних MySQL для ІКС

Для того щоб створити Базу даних необхідно відкрити панель phpMyAdmin, натиснути кнопку "Бази даних" та в полі "Створити базу даних" ввести ім'я нової бази даних. Порівняння потрібно вибрати таке, як на рисунку 3.1 та натиснути кнопку "Создать"

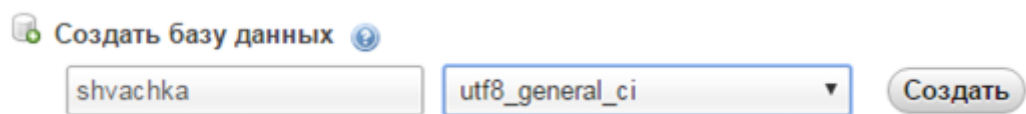


Рисунок 3.1 - Створення Бази даних

Після створення Бази даних до перегляду доступна її структура. На цьому створення завершено.

Структура Бази даних та типи полів в таблицях (рис.3.2):

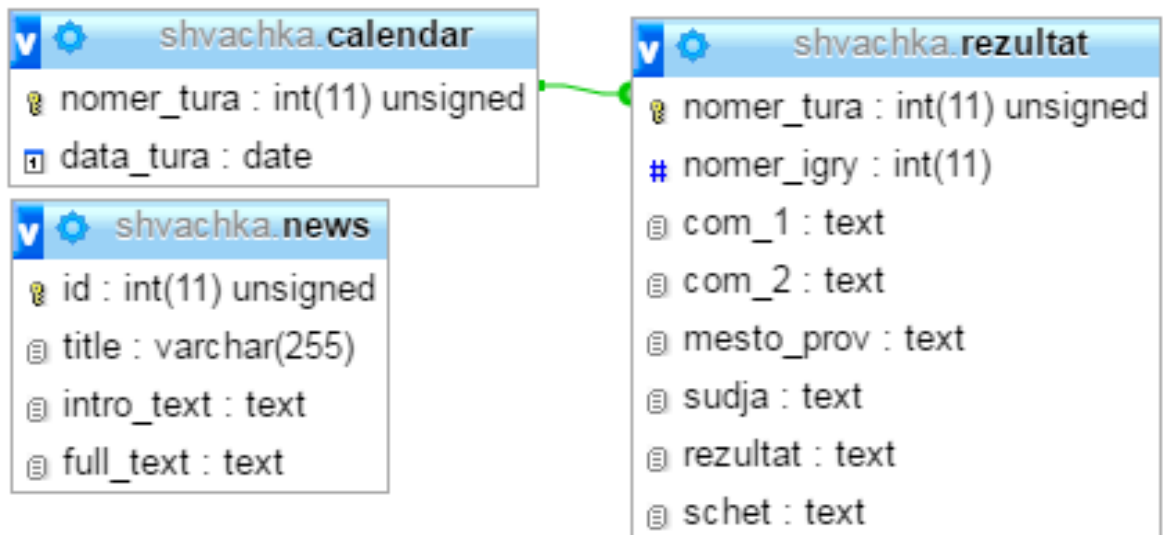


Рисунок 3.2 - Структура Бази даних для ІКС та типи полів

3.2 Головна сторінка інформаційного ресурсу

Створимо директорію для нашого інформаційного ресурсу на локальному сервері. Для цього потрібно зайти на віртуальний диск Denwer створений при установці локального серверу. В даному випадку Z:\. Далі відкрити папку Z:\home\, в якій створити ще одну пусту папку назва якої і буде адресою нашого ресурсу. Далі відкриваємо папку нашого додатку та для правильної роботи створюємо ще одну папку всередині з назвою "www". Повна адреса до ресурсу виглядає так: Z:\home\shvachka.ru\www.

При відкритті адреси ресурсу в браузері завантажується файл index.php. Він є головним файлом інформаційного ресурсу. Для того, щоб файл index.php не содержав увесь код web-додатку, частини ресурсу будуть розроблятися окремими модулями та підключатися у файлі index.php за допомогою php функцій. Весь код файлу index.php (рис. 3.4):



```

index.php * leftblock.php match.php head.php comanda.php
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4 <link href="style.css" rel="stylesheet">
5 <?php require_once "connection.php";
6 $title = "Шахтёр Донецк";
7 include_once "head.php" ?>
8 </head>
9 |
10 <body>
11 <?php
12 include_once "header.php"?>
13 <?php
14 include_once "content.php";
15 include_once "leftblock.php";
16 include_once "footer.php"
17 ?>
18 </body>
19 </html>

```

Рисунок 3.3 - Код файлу "index.php"

В голові web-додатку(<head></head>) підключаються стилі "style.css". Файл "connection.php" - відповідає за підключення до бази даних та функції вибору інформації з бази даних. Файл "head.php" - відповідає за назву ресурсу (тег <title>).

У тілі web-додатку (<body></body>) відбувається підключення всіх файлів котрі будуть розміщені на головній сторінці ІКС.

Файл "header.php" - відповідає за хедер ресурсу та його головне меню.

Файл "content.php" - відповідає за новини на головній сторінці.

Файл "leftblock.php" - відповідає за бокову колонку ресурсу.

Файл "footer.php" - відповідає за футер (нижню частину) ресурсу.

Дизайн головної сторінки бачимо на рисунку 3.4, рисунку 3.5:

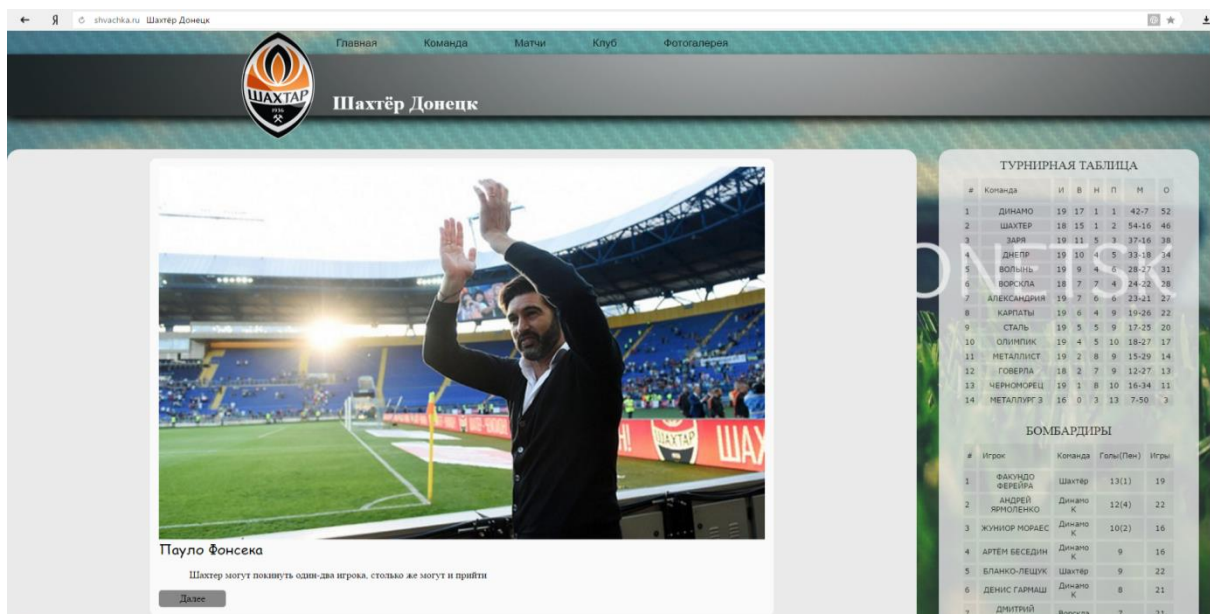


Рисунок 3.4 - Хедер, меню, та новини

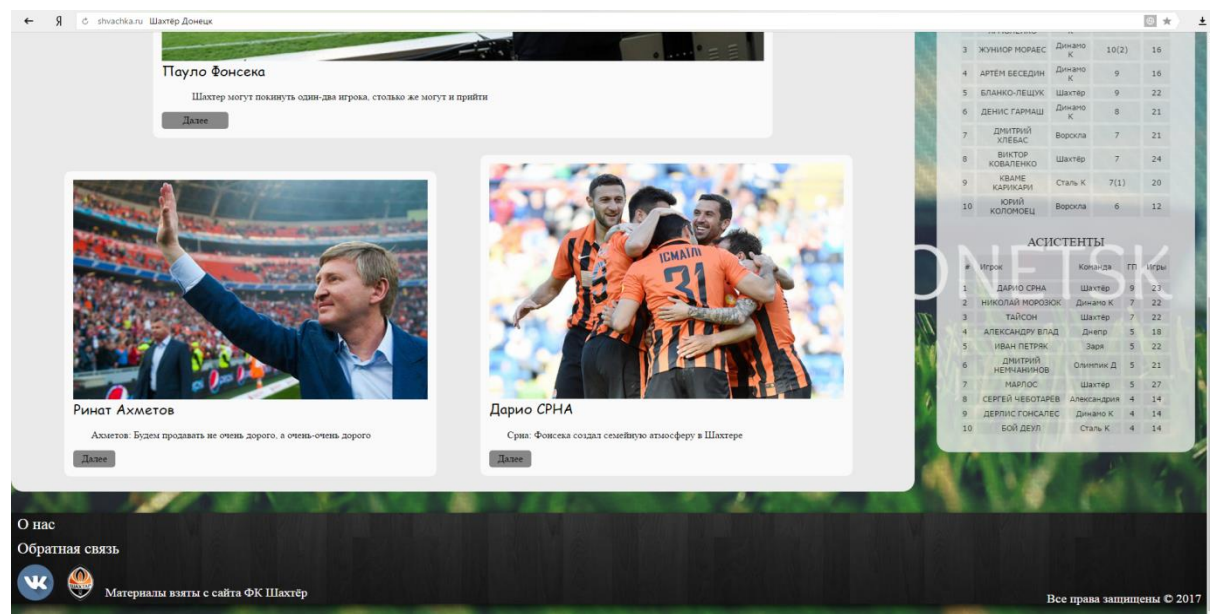


Рисунок 3.5 - Футер, новини, та бокова панель

3.2.1 Модуль "connection.php"

Даний файл відповідає за підключення до бази даних та php запити до неї.

Php код підключення до бази даних (рис. 3.6):

```

1 <?php
2 $mysqli = false;
3 function connectDB () {
4     global $mysqli;
5     $mysqli = new mysqli("localhost", "root", "", "shvachka");
6     $mysqli->query("SET NAMES 'utf-8'");
7 }
8
9 function closeDB () {
10 global $mysqli;
11 $mysqli->close ();
12 }

```

Рисунок 3.6 - Підключення до БД

На рисунку 3.6 видно, що підключення відбувається до бази даних з назвою "shvachka", яка знаходиться на локальному сервері, при цьому акаунт підключення використовується "root".

Php запит до бази даних зображений на рисунку 3.7:

```

14 function getNews ($limit, $id) {
15     global $mysqli;
16     connectDB();
17     if($id)
18         $where = "WHERE `id` = ".$id;
19     $result = $mysqli->query("SELECT * FROM `news`$where ORDER BY `id` DESC LIMIT $limit");
20     closeDB();
21     if(!$id)
22         return resultToArray ($result);
23     else
24         return $result->fetch_assoc();
25 }
26 }
27 function resultToArray ($result) {
28     $array = array ();
29     while (($row = $result->fetch_assoc()) != false)
30         $array[] = $row;
31     return $array;
32 }
33 ?>

```

Рисунок 3.7 - Запит до БД на вибір інформації з таблиці "news"

В таблиці "news" знаходяться новини, котрі відображаються на головній сторінці ресурсу в блоці контенту. Структура "news" представлена у таблиці 3.1:

Таблиця 3.1 - Структура таблиці "news"

Id	Title	Intro_text	Full_text
1	Дарио СРНА	Срна: Фонсека создал семейную атмосферу в Шахтере	Шахтер в нынешнем сезоне выиграл чемпионат и Кубок Украины. 27 мая состоялось награждение Горняков за победу в национальном первенстве
2	Ринат Ахметов	Ахметов: Будем продавать не очень дорого, а очень- очень дорого	Ахметов сообщил, что Шахтер не собирается только продавать игроков, и что если кто-то уйдет, то на его место придет новый сильный футболист.
3	Паулу Фонсека	Шахтер могут покинуть один-два игрока, столько же могут и прийти	Паулу Фонсека отметил, что зрелищность матча оставляла желать лучшего. Также португалец анонсировал, что клуб могут покинуть пара игроков, и столько же прийти им на замену.

Структура таблиці в MySQL відображена на рисунку 3.8:

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Дополнительно	Действие
<input type="checkbox"/>	1 id	int(11)		UNSIGNED	Нет	Нет	AUTO_INCREMENT	Изменить
<input type="checkbox"/>	2 title	varchar(255)	utf8_general_ci		Нет	Нет		Изменить
<input type="checkbox"/>	3 intro_text	text	utf8_general_ci		Нет	Нет		Изменить
<input type="checkbox"/>	4 full_text	text	utf8_general_ci		Нет	Нет		Изменить

Рисунок 3.8 - Структура таблиці "news"

3.2.2 Модуль "header.php"

Файл "header.php" є хедером ресурсу. Хедер - це блок у верхній частині сторінки, який видно на всіх сторінках ресурсу. Містить логотип, меню.

Код файлу "header.php" (рис. 3.9):

```

1 <html>
2 <head>
3 </head>
4 <body>
5 <div class="header">
6 <div class="logo">
7 <ul>
8 <li><a href="/">Главная</a></li>
9 <li><a href="/comanda.php">Команда</a></li>
10 <li><a href="/match.php">Матчи</a></li>
11 <li><a href="/club.php">Клуб</a></li>
12 <li><a href="/gallery.php">Фотогалерея</a></li>
13 </ul>
14 <h1 style="
15     padding-left: 172px;
16     padding-top: 50px;
17     color: white;
18 ">Шахтёр Донецк</h1>
19 </div>
20 </div>
21 </body>
22 </html>

```

Рисунок 3.9 - Код хедеру та меню ІКС

Дизайн хедеру розроблявся за допомогою програми Adobe PhotoShop CS6, тому зображення та логотип підключені за допомогою стилів css [7], як фон об'єктів "div class="header"" та "div class="logo"". Код стилів для хедеру (рис. 3.10):

```

32 /* Стили для хедера*/
33 div.header {
34 width: 100%;
35 height: 182px;
36 margin-top: 1px;
37 position: relative;
38 top: 0px;
39 left: 0px;|
40 background: url(../img/image1.png);
41 background-repeat: no-repeat;
42 width: 100%;
43 height: 180px;
44 opacity: 0.95;
45 filter: alpha(opacity=95);
46 background-size: cover;
47 }
48 div.logo {
49 width: 50%;
50 height: 178px;
51 background: url(../img/image2.png);
52 background-repeat: no-repeat;
53 margin-left: 18%;
54 }

```

Рисунок 3.10 - Стили для хедеру

Зовнішній вигляд хедеру зображений на рисунку 3.11:

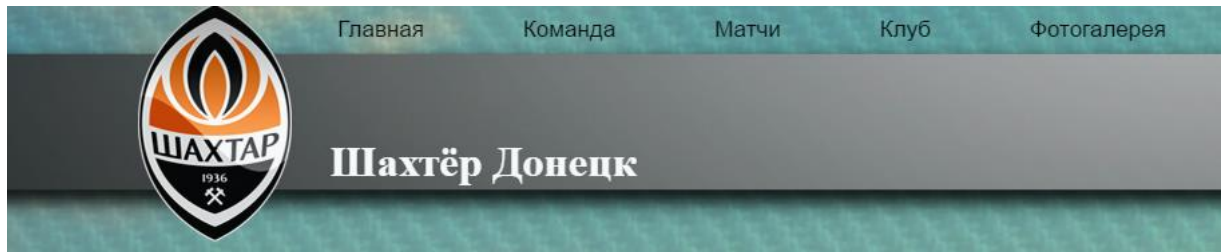


Рисунок 3.11 - Хедер ІКС

3.2.3 Модуль "content.php"

Файл "content.php" відповідає за новини на головній сторінці ресурсу. Котрі беруться з бази даних за допомогою запитів php. У поєднанні з файлом "connection.php" виводимо новини на головну сторінку. При натисканні на кнопку "Далее" відкривається сторінка на якій генерується більш детальна новина. Сторінка з більш детальною інформацією новини одна, а новин може генеруватися багато. Це реалізовано за допомогою GET запитів та присвоєнню ідентифікатора кожній новині. Код файлу "content.php" (рис. 3.12):

```

1 <html>
2 <head>
3 <?php
4     $news = getNews(3,0);
5 ?>
6 </head>
7 <body>
8 <div class="content">
9 <?php
10     for ($i = 0; $i < count($news); $i++) {
11         if($i == 0)
12             echo "<div class='bigArticle'>";
13         else
14             echo "<div class='article'>";
15         echo "<img src='/img/Article/'. $news[$i]['id']. '.jpg' alt='Статья 1' title='Статья '. $news[$i]['id']. '>";
16 <h2>'. $news[$i]['title']. '</h2>";
17 <p>'. $news[$i]['intro_text']. '</p>";
18 <div class="more">
19 <a href="/article.php?id='. $news[$i]['id']. '>Далее</a></div>";
20 </div>";
21         if($i == 0)
22             echo "<div class='clear'><br></div>";
23     }
24 ?>
25 </div>
26 </div>
27 </body>
28 </html>

```

Рисунок 3.12 - Вибір інформації з БД на форму новин

3.2.4 Модуль "leftblock.php"

Файл "leftblock.php" відповідає за бокову панель ресурсу, на якій знаходяться інформаційні показники чемпіонату України з футболу, а саме:

- а) турнірна таблиця ліги;
- б) таблиця найкращих бомбардирів ліги;
- в) таблиця найкращих асистентів ліги.

Реалізовано це за допомогою таблиць HTML.

Уривок коду зображений на рисунку 3.13:

```
<div class="top_bombardir"><span>БОМБАРДИРЫ</span></div>
<div id="turnir_tab">
  <table border="0" cellpadding="5" cellspacing="5" style="width: 90%;margin-left:5%;">
  <thead align="center">
    <tr>
      <td>#</td>
      <td id="name" style="text-align: left;">Игрок</td>
      <td id="a_1" title="Игры">Команда</td>
      <td id="a_2" title="Выиграно">Голы(Пен)</td>
      <td id="a_3" title="Ничья">Игры</td>
    </tr>
  </thead>
```

Рисунок 3.13 - "thead" таблиці найкращих бомбардирів ліги

Таблиці найкращих бомбардирів представлена у таблиці 3.2:

Таблиця 3.2 - Структура таблиці найкращих бомбардирів:

#	Гравець	Команда	Голи (З пен.)	Ігри
1	Фаундо Ферейра	Шахтар	13(1)	19
2	Адрій Ярмоленко	Динамо	12(4)	22
3	Жуніор Мораес	Динамо	10(2)	16
4	Артем Беседін	Динамо	9	16
5	Густаво Юланко-Лещук	Шахтар	9	22
6	Денис Гармаш	Динамо	8	21
7	Дмитро Хльобас	Ворскла	7	21
8	Віктор Коваленко	Шахтар	7	24
9	Кваме Карікарі	Сталь	7(1)	20
10	Юрій Коломоєць	Ворскла	6	12

Дизайн таблиці зображений на рисунку 3.14:

БОМБАРДИРЫ				
#	Игрок	Команда	Голы(Пен)	Игры
1	ФАКУНДО ФЕРЕЙРА	Шахтёр	13(1)	19
2	АНДРЕЙ ЯРМОЛЕНКО	Динамо К	12(4)	22
3	ЖУНИОР МОРАЕС	Динамо К	10(2)	16
4	АРТЁМ БЕСЕДИН	Динамо К	9	16
5	БЛАНКО-ЛЕЩУК	Шахтёр	9	22
6	ДЕНИС ГАРМАШ	Динамо К	8	21
7	ДМИТРИЙ ХЛЁБАС	Ворскла	7	21
8	ВИКТОР КОВАЛЕНКО	Шахтёр	7	24
9	КВАМЕ КАРИКАРИ	Сталь К	7(1)	20
10	ЮРИЙ КОЛОМОЕЦ	Ворскла	6	12

Рисунок 3.14 - Таблица лучших бомбардиров чемпионата

Таблиця має такі стовбці:

- а) номер позиції гравця в рейтингу;
- б) гравець;
- в) команда;
- г) голи (з пенальті);
- д) ігри;

3.2.5 Модуль "footer.php"

Файл "footer.php" відповідає за футер (ноги) ресурсу. Футер - блок в нижній частині сторінки. Містить корисну, але не першорядну інформацію. Видно на всіх

сторінках ресурсу. У футер можна винести: назву студії, яка розробляла портал, контакти.

Зовнішній вигляд футера (рис. 3.15, рис. 3.16):

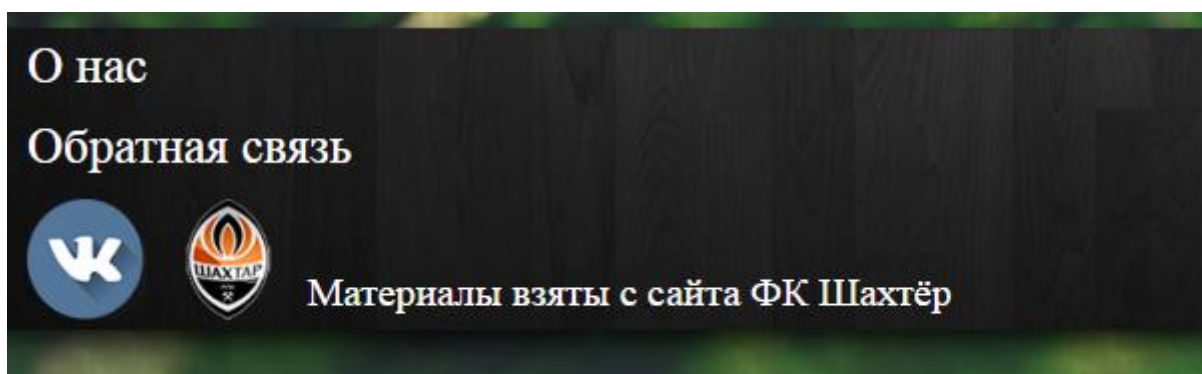


Рисунок 3.15 - Футер. Кнопки "О нас", "Обратная связь"

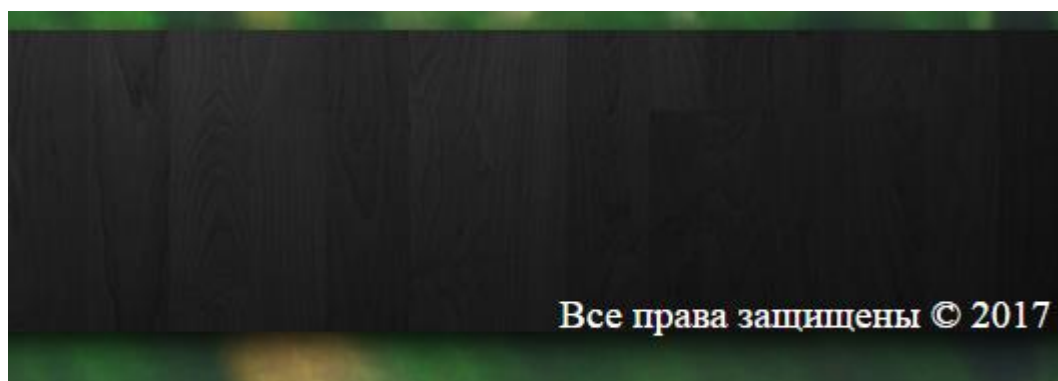


Рисунок. 3.16 - "Все права защищены 2017"

Користувач може перейти на офіційну сторінку розробника, натиснувши на логотип ресурсу "Вконтакте". Натиснувши на логотип футбольного клубу "Шахтар Донецьк" користувач опиниться на офіційному порталі футбольного клубу.

При натисканні на кнопку "О нас" користувач перейде на сторінку де вказана інформація про розробника цього інформаційного ресурсу, та його контакти.

При натисканні на кнопку "Обратная связь" користувач перейде на сторінку де отримає можливість залишити повідомлення розробнику ресурсу, зі своїми питаннями, рекомендаціями або побажаннями.

Форма зворотнього зв'язку реалізована за допомогою технологій jQuery та AJAX.

Дизайн форми зворотнього зв'язку зображено на рисунку 3.17:

The image shows a web form for feedback. It consists of four input fields stacked vertically: 'Имя' (Name), 'Email', 'Тема сообщения' (Subject), and a larger text area for the message with the placeholder text 'Введите сюда Ваше сообщение'. Below the message field is a button labeled 'Отправить' (Send).

Рисунок 3.17 - Форма зворотнього зв'язку

Уривок коду, де відбувається відправка повідомлення на поштову адресу розробника (рис. 3.18):

```

1  <?php
2  $name = htmlspecialchars($_POST['name']);
3  $email = htmlspecialchars($_POST['email']);
4  $subject = htmlspecialchars($_POST['subject']);
5  $message = htmlspecialchars($_POST['message']);
6  if ($name == '' || $email == '' || $subject == '' || $message == ''){
7      echo "Заполните все поля";
8      exit;
9  }
10 // Отправка
11 $subject = "=?utf-8?B?".base64_encode($subject)."?=";
12 $headers = "From: $email\r\nReply-to: $email\r\nContent-type: text/html; charset=utf-8\r\n";
13 if(mail("ptichkd@mail.ru", $subject, $message, $headers))
14     echo "Сообщение отправлено";
15 else
16     echo "Сообщение не отправлено";
17 <?>

```

Рисунок 3.18 - Відправка повідомлення адміністрації ресурсу

3.3 Сторінка "Команда"

Сторінка "Команда", на яку можна перейти через меню у хедері, демонструє користувачу ресурсу інформацію про основний склад команди Шахтар Донецьк.

На сторінці підключені такі модулі:

- а) модуль "header.php" - відповідає за хедер ресурсу та його головне меню.
- б) модуль "comanda.php" - відповідає за новини на головній сторінці.
- в) модуль "leftblock.php" - відповідає за бокову колонку ресурсу.

г) модуль "footer.php" - відповідає за футер (нижню частину) ресурсу.

Модуль "comanda.php" - реалізовано за допомогою таблиці HTML, в якій можна знайти основну інформацію про кожного гравця команди.

За допомогою технології jQuery користувач ресурсу може відсортувати таблицю за будь-яким його стовбцем.

Відсортувавши таблицю за іменами гравців - користувач отримає список гравців в алфавітному порядку.

Відсортувавши таблицю за віком гравця - користувач отримає список гравців від наймолодшого до найстаршого гравця команди.

Відсортувавши список за проведеними матчами в поточному сезоні - користувач отримає список гравців, очолюватиме який гравець з найбільшою, або найменшою кількістю матчів в сезоні.

Відсортувавши список за забитими голами - користувач отримає таблицю де на першому місці стоятиме найкращий бомбардир команди за поточний сезон.

Також таблицю можна відсортувати за стовбцями ГП - голюві передачі, ЖК - жовті картки, ЧК - червоні картки.

Відсортувавши таблицю за параметром "Вартість гравця" - користувач отримає таблицю де зможе побачити найдорожчих та найдешевших гравців команди. Дизайн сторінки "Команда" зображений на рисунку 3.19:



№	Игрок	Возраст	Амплуа	М	Г	Пен	ГП	ЖК	ЧК	Цена игрока (€)
26	Никита Шевченко	24	Вратарь	2	0	0	0	0	0	2000000
30	Андрей Пятов	32	Вратарь	6	0	0	0	1	0	5000000
32	Антон Каниболоцкий	29	Вратарь	0	0	0	0	0	0	1800000
55	Олег Кудрик	20	Вратарь	0	0	0	0	0	0	250000
2	Богдан Бутко	26	Защитник	3	0	0	0	0	0	2000000
5	Александр Кучер	34	Защитник	4	0	0	0	1	0	800000
18	Иван Ордец	24	Защитник	6	0	0	0	1	0	3000000
31	Исмаили	27	Защитник	6	0	0	0	1	0	3000000
33	Дарио Срна	35	Защитник	5	0	0	2	2	0	3000000
38	Сергей Кривцов	26	Защитник	2	2	0	0	1	0	2000000

Рисунок 3.19 - Основний склад команди Шахтар Донецьк

Сортування таблиці за стовбцем "Вартість гравця" (рис. 3.20):



№	Игрок	Возраст	Амплуа	М	Г	Пен	ГП	ЖК	КК	Цена игрока (€)
10	Бернард	24	Полузащитник	8	2	0	1	0	0	14000000
28	Тайсон	29	Полузащитник	8	4	0	1	2	0	12000000
44	Ярослав Ракицкий	27	Защитник	5	0	0	0	1	0	10000000
8	Фред	24	Полузащитник	6	1	0	1	4	1	10000000
11	Марлос	28	Полузащитник	7	0	1	4	0	0	10000000
6	Тарас Степаненко	27	Полузащитник	6	2	0	0	3	0	8000000
74	Виктор Коваленко	21	Полузащитник	6	2	0	0	1	0	6000000
30	Андрей Пятов	32	Вратарь	6	0	0	0	1	0	5000000
19	Факундо Феррейра	26	Форвард	4	3	0	2	0	0	3500000
18	Иван Ордец	24	Защитник	6	0	0	0	1	0	3000000

Рисунок 3.20 - Найдорожчі гравці команди шахтар Донецьк

Частина коду таблиці зображено на рисунку 3.21:

```

92     <thead>
93     <tr>
94         <th>№</th>
95     <th></th>
96     <th class="text-left">Игрок</th>
97     <th>Возраст</th>
98     <th>Амплуа</th>
99     <th aria-haspopup="true" data-tooltip="" title="Матчи">М</th>
100    <th aria-haspopup="true" data-tooltip="" title="Голы">Г</th>
101    <th aria-haspopup="true" data-tooltip="" title="Пенальти">Пен</th>
102    <th aria-haspopup="true" data-tooltip="" title="Голевые передачи">ГП</th>
103    <th aria-haspopup="true" data-tooltip="" title="Желтые карточки">ЖК</th>
104    <th aria-haspopup="true" data-tooltip="" title="Красные карточки">КК</th>
105    </tr>
106 </thead>

```

Рисунок 3.21 - "thead" таблиці команди

Таблиця має такі стовбці:

- а) номер гравця;
- б) фото;
- в) прізвище та ім'я гравця;
- г) вік;
- д) амплуа;
- ж) матчі за сезон;
- и) голи;
- к) пенальті;
- л) голюві передачі;
- м) жовті картки;

- о) червоні картки;
- п) вартість гравця.

3.4 Сторінка "Матчи"

Сторінка "Матчи", на яку можна перейти через меню у хедері.

На сторінці підключені такі модулі:

- а) модуль "header.php" - відповідає за хедер ресурсу та його головне меню.
- б) модуль "match.php" - відповідає за новини на головній сторінці.
- в) модуль "leftblock.php" - відповідає за бокову колонку ресурсу.
- г) модуль "footer.php" - відповідає за футер (нижню частину) ресурсу.

Модуль "match.php" - реалізовано за допомогою таблиці HTML, в якій можна знайти результати кожного футбольного матчу команди Шахтар Донецьк в поточному сезоні.

За допомогою технології jQuery користувач ресурсу може ввести в поле "Input" будь-які цікавічі його параметри для пошуку по таблиці. Адаптивний пошук по таблиці дозволяє швидко та легко знайти потрібні дані.

Пошук проводиться по будь-якому полю таблиці та відображає всі максимально схожі дані за параметр пошуку.

Частина коду таблиці (рис. 3.22):

```

31 <div class="top_zagolovokk" style="font-size: 30px;
32 text-align: center; ">
33 <span>МАТЧИ СЕЗОНА 2016-2017</span></div>
34
35 <div id="turnir_tab">
36   <table border="0" cellpadding="5" cellspacing="5" style="width: 90%;margin-left: 5%;">
37   <thead align="center" style="font-weight: bold; ">
38     <tr>
39       <td style="text-align: center; font-size: 15px; ">ДАТА</td>
40       <td id="name" style="text-align: center; font-size: 15px; ">ТУРНИР</td>
41       <td id="a_1" style="font-size: 15px; " title="Игры">РАУНД</td>
42       <td id="a_2" style="font-size: 15px; " title="Выиграно">КОМАНДА</td>
43       <td id="a_3" style="font-size: 15px; " title="Ничья">СЧЕТ</td>
44       <td id="a_4" style="font-size: 15px; " title="Поражение">КОМАНДА</td>
45     </tr>
46   </thead>
47

```

Рисунок 3.22 - "thead" таблиці матчів

Дизайн сторінки "Команда" зображений на рисунку 3.23:

ДАТА	ТУРНИР	РАУНД	КОМАНДА	СЧЁТ	КОМАНДА
16.07.2016	СУПЕР КУБОК УКРАИНЫ	Финал	Шахтёр	1:1 (3:4)	Динамо
22.07.2016	ЛИГА ПАРИ-МАТЧ	1-й тур	Шахтёр	4:1	Зирка
26.07.2016	ЛИГА ЧЕМПИОНОВ	3 Квал.	Шахтёр	2:0	Янг Бойз
30.07.2016	ЛИГА ПАРИ-МАТЧ	2-й тур	Черноморец	1:4	Шахтёр
03.08.2016	ЛИГА ЧЕМПИОНОВ	3 квал.	Янг Бойз	2:0 (4:2)	Шахтёр
07.08.2016	ЛИГА ПАРИ-МАТЧ	3-й тур	Шахтёр	1:0	Александрия
12.08.2016	ЛИГА ПАРИ-МАТЧ	4-й тур	Карпаты	2:3	Шахтёр
18.08.2016	ЛИГА ЕВРОПЫ	Плей-офф	Истанбул	1:2	Шахтёр
21.08.2016	ЛИГА ПАРИ-МАТЧ	5-й тур	Шахтёр	4:0	Днепр
25.08.2016	ЛИГА ЕВРОПЫ	Плей-офф	Шахтёр	2:0	Истанбул
28.08.2016	ЛИГА ПАРИ-МАТЧ	6-й тур	Сталь	0:1	Шахтёр
09.09.2016	ЛИГА ПАРИ-МАТЧ	7-й тур	Шахтёр	1:1	Динамо
15.09.2016	ЛИГА ЕВРОПЫ	1-й тур	Коньяспор	0:1	Шахтёр
18.09.2016	ЛИГА ПАРИ-МАТЧ	8-й тур	Ворскла	0:1	Шахтёр

Рисунок 3.23 - Всі матчі сезону 2016-2017 років

Пошук в таблиці всіх матчів проти команди "Динамо Київ" зображено на рисунку 3.24:

ДАТА	ТУРНИР	РАУНД	КОМАНДА	СЧЁТ	КОМАНДА
16.07.2016	СУПЕР КУБОК УКРАИНЫ	Финал	Шахтёр	1:1 (3:4)	Динамо
09.09.2016	ЛИГА ПАРИ-МАТЧ	7-й тур	Шахтёр	1:1	Динамо
12.12.2016	ЛИГА ПАРИ-МАТЧ	18-й тур	Динамо	3:4	Шахтёр
21.04.2017	ЛИГА ПАРИ-МАТЧ	26-й тур	Динамо	0:1	Шахтёр
17.05.2017	КУБОК УКРАИНЫ	Финал	Динамо	--	Шахтёр
26.05.2017	ЛИГА ПАРИ-МАТЧ	31-й тур	Шахтёр	--	Динамо

Рисунок 3.24 - Пошук по таблиці матчів сезону 2016 - 2017

Таблица має такі стовбці:

- а) дата матчу;
- б) назва турніру;

- в) раунд турніру;
- г) домашня команда;
- д) рахунок;
- ж) гостьова команда.

3.5 Сторінка "Календарь туров чемпіоната"

Сторінка "Календарь", на яку можна перейти через меню у хедері. Дозволяє користувачу отримати більш детальну інформацію про тур чемпіонату України який його цікавить.

На сторінці підключені такі модулі:

- а) модуль "header.php" - відповідає за хедер ресурсу та його головне меню.
- б) модуль "club.php" - відповідає за новини на головній сторінці.
- в) модуль "leftblock.php" - відповідає за бокову колонку ресурсу.
- г) модуль "footer.php" - відповідає за футер (нижню частину) ресурсу.

Модуль "club.php" - реалізовано за допомогою технології PHP. На сторінці користувачу запропоновано ввести номер туру, інформацію про який він хоче отримати. Після вводу номеру туру, та натисненню кнопки "Показать" скрипт PHP виконує запит до Баз даних. Знаходить номер туру, id якого дорівнює числу котре ввів користувач, та виводить інформацію за цим id на сторінку для перегляду.

Код підключення до БД (рис. 3.25):

```
<?php
$sdd_db_host= 'localhost';
$sdd_db_name= 'shvachka';
$sdd_db_user= 'root';
$sdd_db_pass= '';

@mysql_connect($sdd_db_host, $sdd_db_user, $sdd_db_pass);
@mysql_select_db($sdd_db_name);
?>
```

Рисунок 3.25 - Підключення до БД "shvachka"

Код запиту до БД та вивід інформації на екран (рис. 3.26):

```

<?php
$name = $_POST['name'];

if(isset($_POST['insert']))
{
    $result = mysql_query("SELECT * FROM `rezultat` WHERE nomer_tura = $name");
    while($row = mysql_fetch_array($result)){
        echo '<h2>Место проведения: '.$row['mesto_prov'].'</h2>';
        echo '<h2>'.$row['com_1'].' : '.$row['com_2'].'</h2>';
        echo '<h2>Счёт '.$row['schet'].'</h2>';
        echo '<h2>Результат: '.$row['rezultat'].'</h2>';
        echo '<h2>Судья: '.$row['sudja'].'</h2>';
    }
    $resultt = mysql_query("SELECT * FROM `calendar` WHERE nomer_tura = $name");
    while($roww = mysql_fetch_array($resultt)){
        echo '<h2>Дата тура: '.$roww['data_tura'].'</h2>';
    }
}
?>

```

Рисунок 3.26 - Запит до БД "shvachka"

Вивід інформації за запитом до БД на сторінці "Календарь" зображено на рисунку 3.27:

Главная Команда Матчи Календарь Фотогалерея

ШАХТАР 1936
Шахтёр Донецк

Введите номер тура:

4

Показать

Место проведения: Львов
 Карпаты : Шахтёр
 Счёт 2:3
 Результат: Победа
 Судья: Юрий Абдула
 Дата тура: 2016-08-12

Рисунок 3.27 - Сторінка "Календарь" Запит 4-й тур

Структура таблиці "calendar" (табл. 3.3):

Таблиця 3.3 - Структура таблиці "Calendar":

Nomer_tura	Data_tura
1	2016-07-22
2	2016-07-30
3	2016-08-07
4	2016-08-12

Таблиця у Базі даних MySQL (рис. 3.28):

+ Параметры		nomer_tura	data_tura
<input type="checkbox"/>	Изменить	1	2016-07-22
<input type="checkbox"/>	Изменить	2	2016-07-30
<input type="checkbox"/>	Изменить	3	2016-08-07
<input type="checkbox"/>	Изменить	4	2016-08-12

Рисунок 3.28 - Таблица "calendar" у БД MySQL

Структура таблиці "rezultat" (табл. 3.4):

Таблиця 3.4 - Структура таблиці "rezultat":

Nomer_tura	Com_1	Com_2	Mesto_prov	Sudja	rezultat	sche t
1	Шахтёр	Зирка	Львов	Юрий Можаровский	Победа	4:1
2	Черноморец	Шахтёр	Одесса	Виталий Абдула	Победа	1:4
3	Шахтёр	Александрия	Львов	Анатолий Жабченко	Победа	1:0
4	Карпати	Шахтёр	Львов	Виталий Адула	Победа	2:3

Таблиця у Базі даних MySQL (рис. 3.29):

+ Параметры		nomer_tura	nomer_igry	com_1	com_2	mesto_prov	sudja	rezultat	schet
<input type="checkbox"/>	Изменить	1	1	Шахтёр	Зирка	Львов	Юрий Можаровский	Победа	4:1
<input type="checkbox"/>	Изменить	2	2	Черноморец	Шахтёр	Одесса	Виталий Абдула	Победа	1:4
<input type="checkbox"/>	Изменить	3	3	Шахтёр	Александрия	Львов	Анатолий Жабченко	Победа	1:0
<input type="checkbox"/>	Изменить	4	4	Карпати	Шахтёр	Львов	Юрий Абдула	Победа	2:3

↑ Отметить все / Снять выделение С отмеченными: Изменить Удалить Экспорт

Показать : Начальная строка: Количество строк: Заголовки каждые строк

Рисунок 3.29 - Таблица "rezultat" у БД MySQL

3.6 Сторінка "Фотогалерея"

Сторінка "Фотогалерея", на яку можна перейти через меню у хедері, демонструє фотографії деяких гравців команди.

Модуль "galery.php" - реалізовано за допомогою технологій JavaScript та jQuery. Фотографії можна збільшувати, сортувати за категоріями.

Частина коду файлу "galery.php" (рис. 3.30):

```

1 <!DOCTYPE HTML>
2 <html>
3 <head>
4 <?php
5 $title = "фотогалерея";
6 require_once "head.php";
7 ?>
8
9
10 <script type="text/javascript" src="https://ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"></script>
11 <script src="js/jquery.quicksand.js" type="text/javascript"></script>
12 <script src="js/jquery.easing.js" type="text/javascript"></script>
13 <script src="js/script.js" type="text/javascript"></script>
14 <script src="js/jquery.prettyPhoto.js" type="text/javascript"></script>
15 <link href="css/prettyPhoto.css" rel="stylesheet" type="text/css" />
16 <link type="text/css" href="css/styles.css" rel="stylesheet" media="all" />
17 <link href="css/prettyPhoto.css" rel="stylesheet" type="text/css" />
18 </head>

```

Рисунок 3.30 - Підключення JavaScript скриптів

Код реалізації декількох фотографій (рис.3.31, рис. 3.32):

```

<li class="portfolio-item2" data-id="id-1" data-type="cat-item-2">
  <div>
    <span class="image-block">
      <a class="image-zoom" href="images/big/2.jpg" rel="prettyPhoto[gallery]" title="Виллиан">
      </a>
    </span>
    <div class="home-portfolio-text">
      <h2 class="post-title-portfolio"><a href="#" rel="bookmark" title="Виллиан">Виллиан</a></h2>
      <p class="post-subtitle-portfolio">2015</p>
    </div>
  </div>
</li>

```

Рисунок 3.31 - Код реалізації фотографії з галереї

```

<li class="portfolio-item2" data-id="id-6" data-type="cat-item-1">
  <div>
    <span class="image-block">
      <a class="image-zoom" href="images/big/7.jpg" rel="prettyPhoto[gallery]" title="Андрей Борячук">
      </a>
    </span>
    <div class="home-portfolio-text">
      <h2 class="post-title-portfolio"><a href="#" rel="bookmark" title="Андрей Борячук">Андрей Борячук</a></h2>
      <p class="post-subtitle-portfolio">2017</p>
    </div>
  </div>
</li>

```

Рисунок 3.32 - Код реалізації фотографії Андрія Борячука

Дизайн та роботу фотогалереї (рис. 3.33, рис. 3.34):

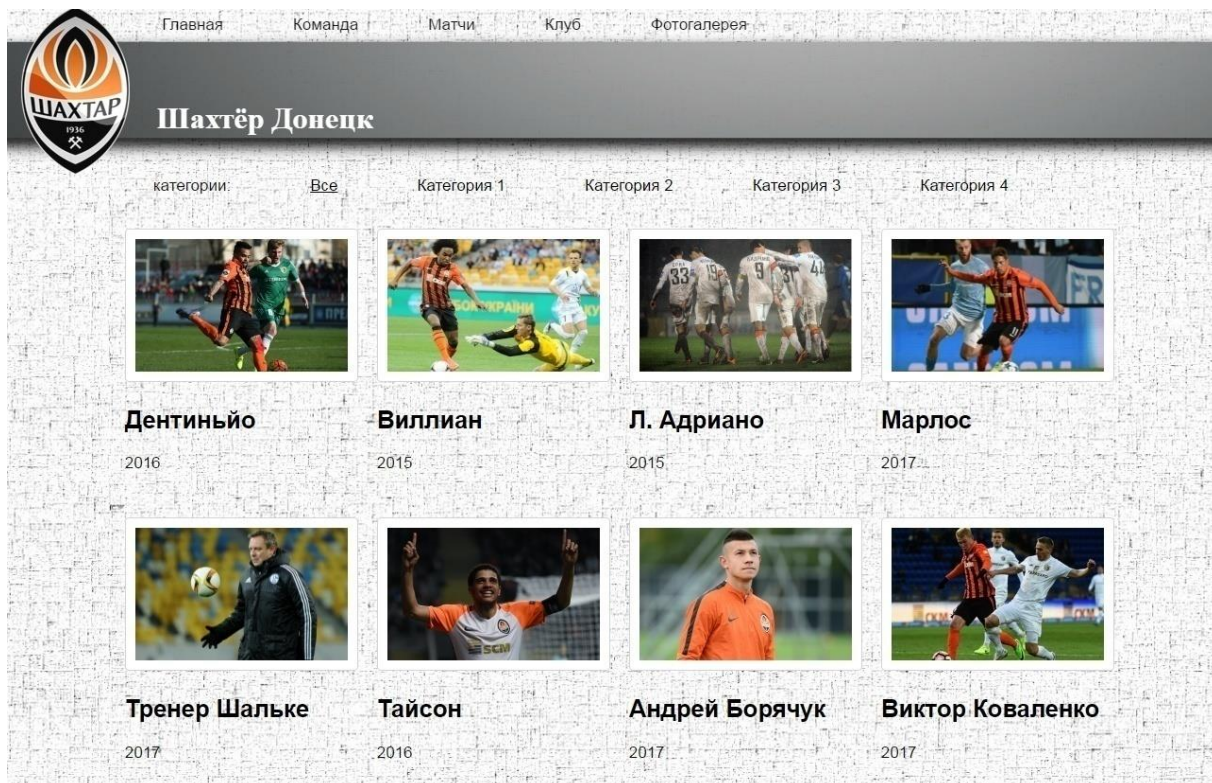


Рисунок 3.33 - Фотогалерея

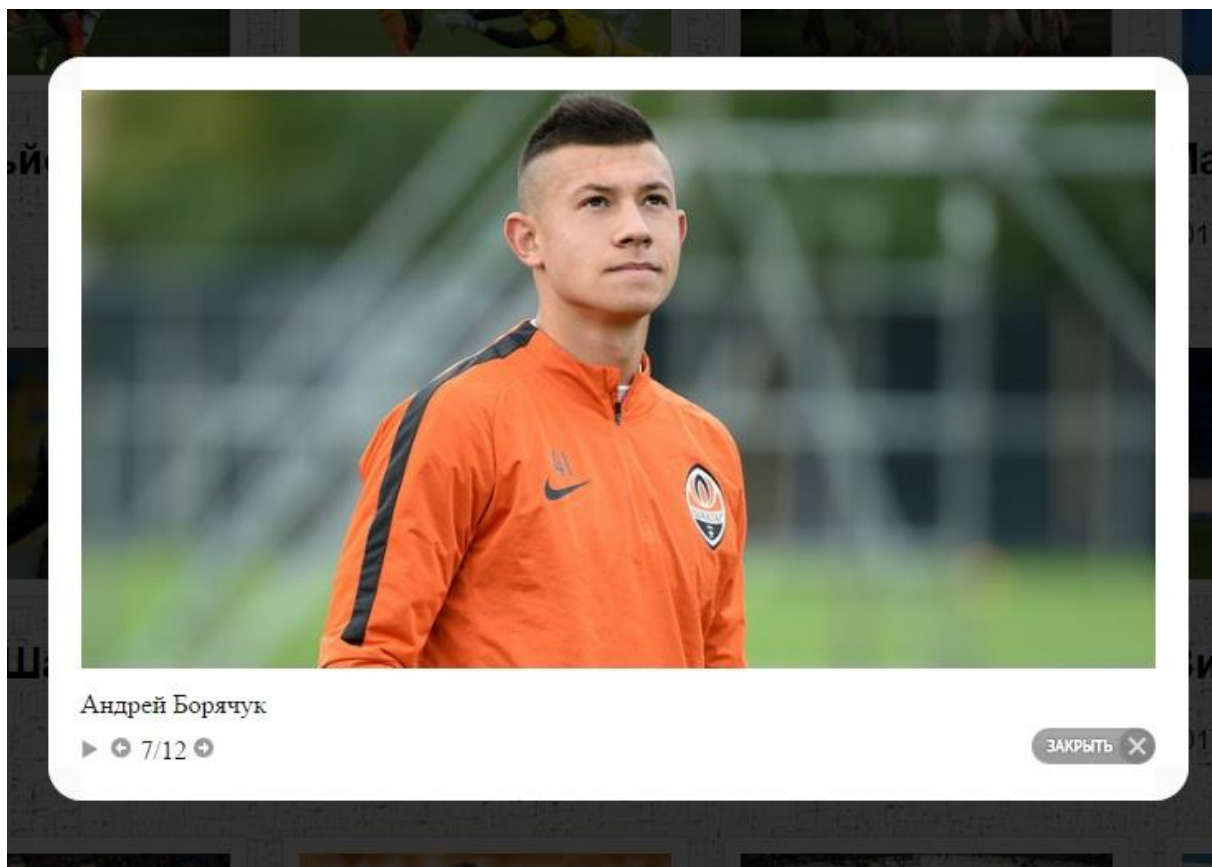


Рисунок 3.34 - Вікно демонстрації фотографій

3.7 Висновки

В процесі розробки ІКС було реалізовано всі необхідні функції, котрі були вказані в технічному завданні.

Реалізація необхідних функцій ресурсу була зроблена за допомогою мов програмування web-програмування, таких як HTML, PHP, jQuery, AJAX, MySQL.

4 ОХОРОНА ПРАЦІ

4.1 Загальні питання з охорони праці

У даному розділі проведено аналіз потенційних небезпечних та шкідливих виробничих факторів, причин пожеж. Розглянуті заходи, які дозволяють забезпечити гігієну праці и виробничу санітарію. На підставі аналізу розроблені заходи з техніки безпеки та рекомендації з пожежної профілактики.

Завданням даної роботи бакалавра було розробити ІКС футбольного клубу, і як результат було створено інформаційний ресурс футбольного клубу. За цим проектом в подальшому розроблятиметься реальна система, яка значно полегшить процес отримання інформації вболівальниками клубу про "ФК Шахтар Донецьк". Так як в процесі проектування використовувався персональний комп'ютер, то аналіз потенційно небезпечних та шкідливих виробничих чинників виконується для персонального комп'ютера на якому буде розроблятися та використовуватися розроблений інформаційний ресурс футбольного клубу.

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та других засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. У законі України «Про охорону праці» визначається, що охорона праці - це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних и лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

При роботі з обчислювальною технікою змінюються фізичні и хімічні фактори навколишнього середовища: виникає статична електрика, електромагнітне випромінювання, змінюється температура и вологість, рівень вмісту кисню й озону в повітрі. Повітря забруднюється шкідливими хімічними

речовини антропогенного походження за рахунок деструкції полімерних матеріалів, які використовують для обробки приміщень та обладнання. Неправильна організація робочого місця сприяє загальній та локальній напрузі м'язів шиї, тулуба, верхніх кінцівок, викривлення хребта и розвитку остеохондрозу.

4.1.1 Правові та організаційні основи охорони праці

Основним організаційним напрямом у здійсненні управління в сфері охорони праці є усвідомлення пріоритету безпеки праці і підвищення соціальної відповідальності держави, і особистої відповідальності працівників.

Державна політика в галузі охорони праці визначається відповідно до Конституції України Верховною Радою України і спрямована на створення належних, безпечних і здорових умов праці, запобігання нещасним випадкам та професійним захворюванням. Відповідно до статті 3 Закону України «Про охорону праці» (далі – Закону) законодавство про охорону праці складається з Закону, Кодексу законів про працю України, Закону України "Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності" та прийнятих відповідно до них нормативно-правових актів, норм міжнародного договору (ратифіковані Конвенції і Рекомендації МОТ, директиви Європейської Ради).

Користувачі персональних комп'ютерів, для яких ця робота є головною, підлягають медичним оглядам: попереднім — під час влаштування на роботу і періодичним — протягом професійної діяльності раз на два роки. Жінок з часу встановлення вагітності та в період годування дитини грудьми до роботи з ПК не допускають.

4.1.2 Організаційно-технічні заходи з безпеки праці

В організації/підприємстві проводяться навчання і перевірка знань з питань охорони праці відповідно до вимог типового положення про порядок проведення навчання і перевірки знань з питань охорони праці, затвердженого наказом Держнагляд охорони праці України від 26.01.2005 N 15, зареєстрованого в Міністерстві юстиції України 15.02.2005 за N 231 [20].

Також впроваджені організаційні заходи з пожежної безпеки - навчання і перевірку знань відповідно до вимог типового положення про інструктажі.

Обов'язковими вимогами враховане наступне:

а) не слід допускати до роботи осіб, що в установленому порядку не пройшли навчання, інструктаж та перевірку знань з охорони праці, пожежної безпеки та цих Правил.

б) на підприємстві/організації, де експлуатуються ЕОМ з відео дисплейними терміналами (ВДТ) і периферійними пристроями (ПП), розробляється інструкція з охорони праці відповідно до положення про розробку інструкцій з охорони праці, затвердженого наказом Держнагляд охорони праці від 29.01.98 N 9, зареєстрованого в Міністерстві юстиції України 07.04.98 за N 226 [21].

в) ознайомлення з правилами безпеки праці, одержання відповідних інструктажів засвідчується у журналі інструктажів.

г) перед допуском до самостійної роботи кожен працівник має право на навчання з питань охорони праці і роботодавець зобов'язаний, і проводить таке навчання у вигляді двох інструктажів з питань охорони праці:

- 1) вступного, який проводять працівники служби охорони праці;
- 2) первинного, який проводять керівники структурних підрозділів на місці праці;
- 3) повторний (не рідше одного разу в 6 місяців);
- 4) позаплановий (при зміні правил охорони праці);

5) поточний (проводять з працівниками перед виконанням робіт, на яких оформляється наряд-допуск);

д) обов'язкові організаційні заходи перед початком, під час і після завершення роботи повинні включати перевірку (візуально) наявності і справності електрообладнання та його заземлення.

Не допускається:

а) виконувати обслуговування, ремонт та налагодження ЕОМ з ВДТ і ПП безпосередньо на робочому місці оператора;

б) зберігати біля ЕОМ з ВДТ і ПП папір, дискети, інші носії інформації, запасні блоки, деталі тощо, якщо вони не використовуються для поточної роботи;

в) відключати захисні пристрої, самочинно проводити зміни у конструкції та складі ЕОМ з ВДТ і ПП або їх технічне налагодження.

4.2 Аналіз стану умов праці

4.2.1 Вимоги до приміщень

Геометричні розміри приміщення зазначені в таблиці 4.1:

Таблиця 4.1 – Розміри приміщення

Найменування	Значення
Довжина, м	6
Ширина, м	4
Висота, м	3
Площа, м ²	24
Об'єм, м ³	72

Згідно з санітарними нормами мікроклімату виробничих приміщень розмір площі для одного робочого місця оператора персонального комп'ютера має бути не менше 6 кв. м, а об'єм — не менше 20 куб. м.

Отже, дане приміщення цілком відповідає зазначеним нормам.

Для зручності спільної роботи з іншими працівниками (обговорення ідей, з'ясування проблем і т.д.) в кімнаті є дивани і журнальний стіл, обставлені живими квітами. Також робочий процес пов'язаний з багатьма документами, теками, журналами для чого приміщення облаштоване принтером і шафою для зручності. Задля дотримання визначеного рівня мікроклімату в будівлі встановлено систему опалення та кондиціонування.

Для забезпечення потрібного рівня освітленості кімната має вікно та систему загального рівномірного освітлення, що встановлена на стелі. Для дотримання вимог пожежної безпеки встановлено порошковий вогнегасник та систему автоматичної пожежної сигналізації.

4.2.2 Вимоги до організації місця праці

При порівнянні відповідності характеристик робочого місця нормативним, основні вимоги до організації робочого місця (табл. 4.2) і відповідними фактичними значеннями для робочого місця, констатуємо повну відповідність.

Робочий стіл на досліджуваному місці також містить достатньо простору для ніг. Крісло, що використовується в якості робочого сидіння, є підйомно поворотним, має підлокітники і можливість регулювання за висотою і кутом нахилу спинки, також воно м'яке і виконане з екологічної шкіри, що дає можливість працювати у комфорті. Екран монітору знаходиться на відстані 0.8 м, клавіатура має можливість регулювання кута нахилу 5-15°.

Отже, за всіма параметрами робоче місце відповідає нормативним вимогам. Приміщення кабінету знаходиться на другому поверсі трьох поверхової будівлі і має об'єм 72м³, площу – 24 м². У цьому кабінеті обладнано три місця праці, з яких два укомплектовані ПК.

У кабінеті є електрична мережа з напругою 220 В, яка створює небезпеку ураження електричним струмом.

Таблиця 4.2 - Характеристики робочого місця

Найменування параметра	Фактичне значення	Нормативне значення
Висота робочої поверхні, мм	750	680 - 800
Висота простору для ніг, мм	730	Не менше 600
Ширина простору для ніг, мм	660	Не менше 500
Глибина простору для ніг, мм	700	Не менше 650
Висота поверхні сидіння, мм	470	400 - 500
Ширина сидіння, мм	400	Не менше 400
Глибина сидіння, мм	400	Не менше 400
Висота поверхні спинки, мм	600	Не менше 300
Ширина опорної поверхні спинки, мм	500	Не менше 380
Радіус спинки в горизонтальній площині, мм	400	400
Відстань від очей до екрану дисплея, мм	800	700 - 800

Температура в приміщенні протягом року коливається у межах 18–24°C, відносна вологість близько 50%. Швидкість руху повітря не перевищує 0,2 м/с. Шум в лабораторії знаходиться на рівні 50 дБА. Система вентилявання приміщення - природна неорганізована, а опалення - централізоване.

Розміщення вікон забезпечує природне освітлення з коефіцієнтом природного освітлення не менше 1,5%, а загальне штучне освітлення, яке здійснюється за допомогою восьми люмінесцентних ламп, забезпечує рівень освітленості не менше 200 Лк.

За ступенем пожежної безпеки приміщення належить до категорії В. Кабінет оснащений переносним вуглекислотним вогнегасником ВВК-5 .

Наявна аптечка для надання долікарської допомоги, а також у кабінеті роблять вологе прибирання та щоденно провітрюють приміщення.

4.2.3 Навантаження та напруженість процесу праці

Як приклад наведено опис процесу праці зі створення інформаційної комп'ютерної системи, під час виконання випускної роботи бакалавра:

за фізичним навантаженням робота відноситься до категорії легкі роботи (Ia), її виконують сидячи з періодичним ходінням. Щодо характеру організування виконання дипломної роботи, то він підпадає під нав'язаний режим, оскільки певні розділи роботи необхідно виконати у встановлені конкретні терміни. За ступенем нервово-психічної напруги виконання роботи можна віднести до II – III ступеня і кваліфікувати як помірно напружений – напружений за умови успішного виконання поставлених завдань.

Під час виконання робіт використовують ПК та периферійні пристрої (лазерні та струменеві), що призводить до навантаження на окремі системи організму.

Найбільшому ризику виникнення різноманітних порушень піддаються: органи зору, м'язово-скелетна система, нервово-психічна діяльність, репродуктивна функція у жінок.

Тобто наявні психофізіологічні небезпечні та шкідливі фактори:

а) фізичного перевантаження:

- 1) статичного;
- 2) динамічного;

б) нервово-психічного перевантаження:

- 1) розумового перенапруження;
- 2) монотонності праці;
- 3) перенапруження аналізаторів;
- 4) емоційних перевантажень.

Рекомендовано застосування екранних фільтрів, локальних світлофільтрів (засобів індивідуального захисту очей) та інших засобів захисту, а також інші профілактичні заходи.

4.3 Виробнича санітарія

На підставі аналізу небезпечних та шкідливих факторів при виробництві (експлуатації), пожежної безпеки можуть бути надалі вирішені питання необхідності забезпечення працюючих достатньою кількістю освітлення, вентиляції повітря, організації заземлення, тощо.

4.3.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу

Аналіз небезпечних та шкідливих виробничих факторів виконується у табличній формі (табл. 4.3). Роботу, пов'язану з ЕОП з ВДТ, у тому числі на тих, які мають робочі місця, обладнані ЕОМ з ВДТ і ПП, виконують із забезпеченням виконання «Правил охорони праці під час експлуатації електронно-обчислювальних машин», які встановлюють вимоги безпеки до обладнання робочих місць, до роботи із застосуванням ЕОМ з ВДТ і ПП. Переважно роботи за проектами виконують у кабінетах чи інших приміщеннях, де використовують різноманітне електрообладнання, зокрема персональні комп'ютери (ПК) та периферійні пристрої. Основними робочими характеристиками персонального комп'ютера є:

- а) робоча напруга $U=+220\text{В} \pm 5\%$;
- б) робочий струм $I=2\text{А}$;
- в) споживана потужність $P=350\text{ Вт}$.

Робочі місця мають відповідати вимогам Державних санітарних правил і норм роботи з візуальними дисплейними терміналами електронно-обчислювальних машин.

Таблиця 4.3 – Аналіз небезпечних і шкідливих виробничих факторів

Небезпечні і шкідливі виробничі фактори	Джерела факторів (види робіт)	Кількісна оцінка	Нормативні документи
1	2	3	4
Фізичні:			
- підвищена температура поверхонь обладнання	експлуатація ЕОМ, принтерів, сканерів для роботи	2	[22]
- підвищений рівень шуму на робочому місці	-//-	2	[23]
- підвищений рівень вібрації	-//-	2	[24] [25]
- підвищена або знижена вологість повітря	-//-	2	[22]
- підвищена або знижена рухливість повітря	-//-	1	[22]
- підвищений рівень електромагнітного випромінення	-//-	2	[26]
- підвищений рівень напруги електричної мережі, замикання може відбутися через тіло	-//-	4	[27] [28]
- підвищений рівень статичної електрики	-//-	2	[27]
- підвищена напруженість електричного поля	-//-	2	[26]
- підвищена яскравість світла	порушення умов праці налагодження моніторів)	1	[29]
Психофізіологічні:			
- нервово-психічна Перевантаження(розумове, Перенапруження аналізаторів-зорових)	- нервово-психічна - пошук інформації для Перевантаження постановки теми; - виконання роботи за темою диплома, - оформлення роботи	4	[30] [29]

4.3.2 Пожежна безпека

Небезпека розвитку пожежі на обчислювальному центрі обумовлюється застосуванням розгалужених систем електроживлення ЕОМ, вентиляції і кондиціонування. Небезпека загоряння пов'язана з особливістю комп'ютерів - із значною кількістю щільно розташованих на монтажній платі і блоках електронних вузлів і схем, електричних і комутаційних кабелів, резисторів, конденсаторів, напівпровідникових діодів і транзисторів. При відхиленні реальних умов експлуатації від розрахункових можуть виникнути пожежонебезпечні ситуації.

Висока щільність елементів в електронних схемах призводить до значного підвищення температури окремих вузлів (80...100 °С). При проходженні електричного струму по провідниках і деталей виділяється тепло, що в умовах їх високої щільності може привести до перегріву, і може служити причиною запалювання ізоляційних матеріалів.

Для гасіння пожеж в офісному приміщенні пропонується використовувати порошкові або вуглекислотні вогнегасники, так як вони є універсальними. Дане приміщення оснащено системою автоматичної пожежної сигналізації, має 1 вогнегасник ВП-5 із зарядом вогнегасної речовини 8-12 кг, відповідно до вимог чинного законодавства України. Проходи до засобів пожежогасіння вільні, не захарашуються та у разі потреби забезпечувати евакуацію всіх людей, які перебувають у приміщенні через один евакуаційний вихід з дверима на шляху евакуації, що відчиняється в напрямку виходу з будівлі від робочого місця. В приміщенні наявна затверджена «План-схема евакуації з кабінету (приміщення)».

Пожежна безпека при застосуванні ЕОМ забезпечується:

- а) системою запобігання пожежі;
- б) системою протипожежного захисту;
- в) організаційно-технічними заходами.

Виникнення пожежі можливе, якщо на об'єкті є горючі речовини, окислювач і джерела запалювання. Вірогідність пожежної небезпеки приймається значною, якщо ймовірна взаємодія цих трьох чинників. Горючими компонентами є: будівельні матеріали для акустичної і естетичної обробки приміщень, перегородки, підлоги, двері, ізоляція силових, сигнальних кабелів і т.д.

Для відводу теплоти від ЕОМ діє потужна система кондиціонування. Тому кисень, як окиснювач процесів горіння, є в будь-якій точці приміщень обчислювального центру.

Потенційними джерелами запалювання можуть бути:

- а) іскри і дуги короткого замикання;
- б) електрична іскра при замиканні і розмиканні ланцюгів;
- в) перегріву від тривалого перевантаження;
- г) відкритий вогонь і продукти горіння;
- д) наявність речовин, нагрітих вище за температуру самозаймання;
- ж) розрядна статична електрика.

Причинами можливого загорання і пожежі можуть бути:

- а) несправність електроустановки;
- б) конструктивні недоліки устаткування;
- в) коротке замикання в електричних мережах;
- г) запалювання горючих матеріалів, що знаходяться в безпосередній близькості від електроустановки.

Для захисту персоналу від дії небезпечних і шкідливих чинників пожежі проектом передбачається застосування промислового протигазу, що фільтрує, з коробкою марки «В» із сірою відміткою забарвлення – захист від неорганічних газів (хлор, фтор, бром, сірководень, сірковуглець, хлорціан, галогени), а цей фільтр не захистить від СО (тобто від чадного газу).

4.3.3 Електро безпека

На робочому місці виконуються наступні вимоги електробезпеки: ПК, периферійні пристрої та устаткування для обслуговування, електропроводи і кабелі за виконанням та ступенем захисту відповідають класу зони за ПУЕ (правила улаштування електроустановок), мають апаратуру захисту від струму короткого замикання та інших аварійних режимів. Лінія електромережі для живлення ПК, периферійних пристроїв і устаткування для обслуговування, виконана як окрема групова три- провідна мережа, шляхом прокладання фазового, нульового робочого та нульового захисного провідників. Електромережа штепсельних розеток для живлення персональних ПК, укладено по підлозі поруч зі стінами відповідно до затвердженого плану розміщення обладнання та технічних характеристик обладнання.

4.4 Гігієнічні вимоги до параметрів виробничого середовища

4.4.1 Параметри мікроклімату

Оптимальні значення для температури, відносної вологості й рухливості повітря для зазначеного робочого місця відповідають санітарним нормам мікроклімату виробничих приміщень і наведені в табл. 4.4:

Таблиця 4.4 – Норми мікроклімату робочої зони об'єкту

Період року	Категорія робіт	Температура С ⁰	Відносна вологість %	Швидкість руху повітря, м/с
Холодна	Легка-1 а	22-24	40-60	0,1
Тепла	Легка-1 а	23-25	40-60	0,1

Мікроклімат робочих приміщень – це клімат внутрішнього середовища цих приміщень, що визначається діючої на організм людини з'єднанням температури, вологості, швидкості переміщення повітря. В даному приміщенні проводяться

роботи, що виконуються сидячи і не потребують динамічного фізичного напруження, то для нього відповідає категорія робіт Іа.

Дане приміщення обладнане системами опалення, кондиціонування повітря або припливно-витяжною вентиляцією. У приміщенні на робочому місці забезпечуються оптимальні значення параметрів мікроклімату: температури, відносної вологості й рухливості повітря. Для забезпечення оптимальних параметрів мікроклімату в приміщенні проводяться перерви в роботі співробітників, з метою його провітрювання. Існують спеціальні системи кондиціонування, які забезпечують підтримання в приміщенні балансу оптимальних параметрів мікроклімату.

4.4.2 Освітлення

Світло є природною умовою існування людини. Воно впливає на стан вищих психічних функцій і фізіологічні процеси в організмі. Хороше освітлення діє тонізуюче, створює гарний настрій, покращує протікання основних процесів вищої нервової діяльності.

Збільшення освітленості сприяє поліпшенню працездатності навіть в тих випадках, коли процес праці практично не залежить від зорового сприйняття.

Освітленість приміщення має велике значення при роботі на ПЕОМ. Вона багато в чому визначається колірною і мережевий обстановкою. Для зменшеного поглинання світла стеля і стіни вище панелей (1,5-1,7м.).

Основний потік природного світла при цій повинен бути зліва. Не допускається спрямування основного світлового потоку природного світла праворуч, ззаду і спереду працівника на ПЕОМ.

Робота на ПЕОМ може здійснюватися за таких видах освітлення:

а) загальному штучному освітленні, коли відео монітори розташовуються по периметру приміщення або при центральному розташуванні робочих місць у два ряди по довжині кімнати з екранами, звернені в протилежні сторони;

б) суміщене освітлення (природне + штучне) тільки при одному і трьох рядном розташуванні робочих місць, коли екран і поверхню робочого столу знаходяться перпендикулярно світла несучій стіні.

У проєкті, що розробляється, передбачається використовувати суміщене освітлення. У світлий час доби використовуватиметься природне освітлення приміщення через віконні отвори, в решту часу використовуватиметься штучне освітлення. Штучне освітлення створюється газорозрядними лампами.

Штучне освітлення в робочому приміщенні передбачається здійснювати з використанням люмінесцентних джерел світла в світильниках загального освітлення, оскільки люмінесцентні лампи мають високу потужність (80 Вт), тривалий термін служби (до 10000 годин). Джерелом природного освітлення є сонячне світло.

Розрахунок освітлення.

Для виробничих та адміністративних приміщень світловий коефіцієнт приймається не менше $1/8$, в побутових – $1/10$:

$$S_b = \left(\frac{1}{5} \div \frac{1}{10} \right) \cdot S, \quad (4.1)$$

де S_b – площа віконних прорізів,

m^2 ; S_n – площа підлоги, m^2 .

$$S_n = a \cdot b = 6 \cdot 4 = 24 \text{ м}^2,$$

$$S = 1/8 \cdot 24 = 3 \text{ м}^2.$$

Приймаємо 2 вікна площею $S=1,6 \text{ м}^2$ кожне.

Світильники загального освітлення розташовуються над робочими поверхнями в рівномірно-прямокутному порядку. Для організації освітлення в темний час доби передбачається обладнати приміщення, довжина якого складає 5 м, ширина 5 м, світильниками ЛПО2П, оснащеними лампами типу ЛБ (дві по 80 Вт) з світловим потоком 5400 лм кожна.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для

створення заданої освітленості при загальному рівномірному освітленні. Розрахунок кількості світильників n виробляється по формулі (4.2):

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M}, \quad (4.2)$$

де E – нормована освітленість робочої поверхні, визначається нормами – 300 лк;

S – освітлювана площа, m^2 ; $S = 24 m^2$;

Z – поправочний коефіцієнт світильника ($Z = 1,15$ для ламп розжарювання та ДРЛ; $Z = 1,1$ для люмінесцентних ламп) приймаємо рівним 1,1;

K – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1,5;

U – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. – 0,575;

M – число люмінесцентних ламп в світильнику – 2;

F – світловий потік лампи – 5400лм (для ЛБ-80).

Підставивши числові значення у формулу (4.2), отримуємо:

$$n = \frac{300 \cdot 24 \cdot 1,1 \cdot 1,5}{5400 \cdot 0,575 \cdot 2} \approx 2,0$$

Приймаємо освітлювальну установку, яка складається з 2-х світильників, які складаються з двох люмінесцентних ламп загальною потужністю 160 Вт, напругою – 220 В.

4.4.3 Шум та вібрація, електромагнітне випромінювання

Рівень шуму, що супроводжує роботу користувачів персональних комп'ютерів (зумовлений як роботою системних блоків, клавіатури, так і друкуванням на принтерах, а також зовнішніми чинниками), коливається у межах 50–65 дБА. Шум такої інтенсивності на тлі високого ступеня напруженості праці негативно впливає на функціональний стан користувачів. Тому на практиці

рекомендують знижувати фактичний рівень шуму у приміщеннях, де створюють комп'ютерні програми, виконують теоретичні та творчі роботи, проводять навчання до 40 дБА. У залах опрацювання інформації та комп'ютерного набору рівні шуму не повинні перевищувати 65 дБА.

Для зниження шуму на шляху його поширення передбачається розміщення в приміщенні штучних поглиначів. Для зниження рівня шуму стелю або стіни вище 1.5 - 1.7 метра від підлоги повинні облицьовуватися звукопоглинальним матеріалом з максимальним коефіцієнтом звукопоглинання в області частот 63-8000 Гц.

Віброізоляція можливо здійснювати за допомогою спеціальної прокладки під системний блок, який послаблює передачу вібрацій робочого столу. Для захисту від електромагнітного випромінювання передбачаються наступні заходи:

- а) застосування нових плазмових моніторів, LG W2271TC;
- б) віддалення робочого місця не менше, ніж на 0,4 – 0,5 м, оскільки напруженість електричного поля зменшується при віддаленні від джерела поля;
- в) встановлення раціональних режимів роботи персоналу (обмеження часу перебування);
- г) раціональне розміщення в робочому приміщенні устаткування, що випромінює електромагнітну енергію.

4.4.4 Вентилювання

У приміщенні, де знаходяться ЕОМ, повітрообмін реалізується за допомогою природної організованої вентиляції (вентиляційні шахти), тобто при V приміщення більше 40 м³ на одного працюючого допускається природна вентиляція. Цей метод забезпечує приток потрібної кількості свіжого повітря, що визначається в СНіП.

Також має здійснюватися провітрювання приміщення, в залежності від погодних умов, тривалість повинна бути не менше 10 хв. Найкращий обмін повітря здійснюється при наскрізному провітрюванні.

4.5 Заходи з організації виробничого середовища та попередження виникнення питань надзвичайних ситуацій

Відповідно до санітарно-гігієнічних нормативів та правил експлуатації обладнання наводимо приклади деяких заходів безпеки.

а) Заходи безпеки під час експлуатації персонального комп'ютера та периферійних пристроїв передбачають:

1) правильне організування місця праці та дотримання оптимальних режимів праці та відпочинку під час роботи з ПК;

2) експлуатацію сертифікованого обладнання;

3) дотримання заходів електробезпеки;

4) забезпечення оптимальних параметрів мікроклімату;

5) забезпечення раціонального освітлення місця праці (освітленість робочого місця не перевищувала $2/3$ нормальної освітленості приміщення);

б) облаштовуючи приміщення для роботи з ПК, потрібно передбачити припливно-витяжну вентиляцію або кондиціювання повітря;

7) зниження рівня шуму та вібрації.

б) Заходи безпеки під час експлуатації інших електричних приладів передбачають дотримання таких правил:

1) постійно стежити за справним станом електромережі, розподільних щитків, вимикачів, штепсельних розеток, лампових патронів, а також мережевих кабелів живлення, за допомогою яких електроприлади під'єднують до електромережі;

2) постійно стежити за справністю ізоляції електромережі та мережевих кабелів, не допускаючи їхньої експлуатації з пошкодженою ізоляцією;

- 3) не тягнути за мережевий кабель, щоб витягти вилку з розетки;
- 4) не закривати меблями, різноманітним інвентарем вимикачі, штепсельні розетки;
- 5) не підключати одночасно декілька потужних електричних пристроїв до однієї розетки, що може викликати надмірне нагрівання провідників, руйнування їхньої ізоляції, розплавлення і загорання полімерних матеріалів;
- б) не залишати включені електроприлади без нагляду;
- 7) не допускати потрапляння всередину електроприладів крізь вентиляційні отвори рідин або металевих предметів, а також не закривати їх та підтримувати в належній чистоті, щоб уникнути перегрівання та займання приладу;
- 8) не ставити на електроприлади матеріали, які можуть під дією теплоти, що виділяється, загорітися (канцелярські товари, сувенірну продукцію тощо).

Вимоги безпеки при надзвичайних ситуаціях:

а) При раптовому припиненні подачі електричної енергії вимкнути всі пристрої ПК в такій послідовності: периферійні пристрої, ВДТ, системний блок, стабілізатор (або блок безперервного живлення). Витягнути вилки з розеток. При наявності ознак горіння (дим, запах горілого) необхідно вимкнути всі пристрої ПК, знайти місце загорання і виконати всі можливі заходи для його ліквідації, попередивши терміново про це керівництво. У випадку виникнення пожежі негайно попередити про це пожежну частину та керівництво, виконати усі можливі заходи по евакуації людей з приміщення і розпочати гасіння пожежі первинними засобами пожежогасіння.

б) При замиканні, перевантаженні електричного струму на електричному обладнанні, внаслідок ураження грозової блискавки та ймовірної небезпеки ураженням електричним струмом, приймають наступне:

- 1) попередження замикання здійснюється правильним вибором, монтажем експлуатації мереж;

- 2) застосування захисту схем у вигляді швидкодіючих реле, а також вимикачів, плавких запобіжників, автоматичних вимикачів.

в) У випадку дотику до корпусу та інших струмоведучих частин електроустановки, що опинилися під напругою використовують захисне заземлення-зниження до безпечних значень напруги дотику і кроку, обумовлених замиканням на корпус та ін. Це досягається шляхом, зменшення потенціалу заземленого обладнання (за рахунок підйому потенціалу підстави, на якому стоїть людина, до значення, близького до значення потенціалу заземленого обладнання) та відключення від загальної електромережі ураженого обладнання.

Також застосовують різні електричні захисні засоби від ураження струмом:

а) Ізолюючі - ізолюють людини від струмоведучих або заземлених частин, а так-же від землі. Вони діляться на основні та додаткові.

б) Основні - володіють ізоляцією, здатної довго витримувати робочу напругу електроустановки. До них відносяться: в електроустановках до 1000 Вт - діелектричної рукавички, ізолюючі штанги, ізолюючі і електровимірювальні кліщі і т.д.; понад 1000 Вт - ізолюючі штанги, і електровимірювальні кліщі.

в) Запобіжні - володіють ізоляцією нездатною витримати робочу напругу електроустановки, і тому вони не можуть самостійно захищати людину від ураження струмом під цією напругою. До запобіжних відносяться засоби в електроустановках до 1000 Вт - діелектричні калоші килимки, а також ізолюючі підставки.

Розрахунок захисного заземлення (забезпечення електробезпеки будівлі).

Згідно з класифікацією приміщень за ступенем небезпеки ураження електричним струмом, приміщення в якому проводяться всі роботи відноситься до першого класу (без підвищеної небезпеки). Під час роботи використовуються електроустановки з напругою живлення 36 В, 220 В, та 360 В. Опір контура заземлення повинен мати не більше 4 Ом.

Розрахунок проводять за допомогою методу коефіцієнта використання (екранування) електродів. Коефіцієнт використання групового заземлювача η – це відношення діючої провідності цього заземлювача до найбільш можливої його провідності за нескінченно великих відстаней між його електродами. Коефіцієнт використання вертикальних заземлювачів η_v в залежності від розміщення

заземлювачів та їх кількості знаходиться в межах 0,4-0,99. Взаємну екрануючу дію горизонтального заземлювача (з'єднувальної смуги) враховують за допомогою коефіцієнта використання горизонтального заземлювача η_s .

Послідовність розрахунку.

а) Визначається необхідний опір штучних заземлювачів $R_{шт.з.}$:

$$R_{шт.з.} = \frac{R_d \cdot R_{пр.з.}}{R_{пр.з.} - R_d}, \quad (4.3)$$

де $R_{пр.з.}$ – опір природних заземлювачів;

R_d – допустимий опір заземлення.

Якщо природні заземлювачі відсутні, то $R_{шт.з.} = R_d$.

Підставивши числові значення у формулу (4.3), отримуємо:

$$R_{шт.з.} = \frac{4 \cdot 40}{40 - 4} \approx 40 \text{ Ом},$$

б) Опір заземлення в значній мірі залежить від питомого опору ґрунту ρ , Ом*м. Приблизне значення питомого опору глини приймаємо $\rho = 40$ Ом*м (табличне значення).

в) Розрахунковий питомий опір ґрунту, $\rho_{розр.}$, Ом*м, визначається відповідно для вертикальних заземлювачів $\rho_{розр.в.}$, і горизонтальних $\rho_{розр.г.}$, Ом*м за формулою:

$$\rho_{розр.} = \psi \cdot \rho, \quad (4.4)$$

де ψ – коефіцієнт сезонності для вертикальних заземлювачів І кліматичної зони з нормальною вологістю землі, приймається для вертикальних заземлювачів $\rho_{розр.в.} = 1,7$ і горизонтальних $\rho_{розр.г.} = 5,5$ Ом*м.

$$\rho_{розр.в.} = 1,7 \cdot 40 = 68 \text{ Ом} \cdot \text{м},$$

$$\rho_{розр.г.} = 5,5 \cdot 40 = 220 \text{ Ом} \cdot \text{м},$$

г) Розраховується опір розтікання струму вертикального заземлювача R_v , Ом, за (4.5).

$$R_g = \frac{\rho_{розг}}{2 \cdot \pi \cdot l_g} \left(\frac{\ln 2 \cdot l}{d_{cm}} + \frac{1 \cdot \ln}{2} + \frac{4 \cdot t + 1_g}{4 \cdot t - 1_g} \right), \quad (4.5)$$

де l_v – довжина розр. вертикального заземлювача (для труб - 2–3 м; $l_v=3$ м);
 $d_{ст}$ – діаметр стержня (для труб - 0,03–0,05 м; $d_{ст}=0,05$ м);

t – відстань від поверхні землі до середини заземлювача, яка визначається за формулою (4.6):

$$t = h_v + \frac{1_g}{2}, \quad (4.6)$$

де h_v – глибина закладання вертикальних заземлювачів (0,8 м); тоді

$$t = 0,8 + \frac{3}{2} = 2,3 \text{ м}$$

$$R_g = \frac{68}{2 \cdot \pi \cdot 3} \cdot \left(\ln \frac{2 \cdot 3}{0,05} + \frac{1}{2} \ln \frac{4 \cdot 2,3 + 3}{4 \cdot 2,3 - 3} \right) = 18,5 \text{ Ом}$$

г) Визначається теоретична кількість вертикальних заземлювачів n штук, без урахування коефіцієнта використання η_v :

$$n = \frac{2 \cdot R_g}{R_0} = \frac{2 \cdot 18,5}{4} = 9,25 \quad (4.7)$$

Визначається коефіцієнт використання вертикальних електродів групового заземлювача без врахування впливу з'єднувальної стрічки $\eta_v = 0,57$ (табличне значення).

д) Визначається необхідна кількість вертикальних заземлювачів з урахуванням коефіцієнта використання η_v , шт:

$$n_g = \frac{2 \cdot R_g}{R_0 \cdot \eta_g} = \frac{2 \cdot 18,5}{4 \cdot 0,57} = 16,2 \approx 16 \quad (4.8)$$

е) Визначається довжина з'єднувальної стрічки горизонтального заземлювача l_c , м:

$$l_c = 1,05 \cdot L_g \cdot (n_g - 1) \quad (4.9)$$

де L_v – відстань між вертикальними заземлювачами, (прийняти за $L_v = 3$ м);
 n_v – необхідна кількість вертикальних заземлювачів.

$$I_c = 1,05 \cdot 3 \cdot (16 - 1) \approx 48 \text{ мА}$$

Висновок: дане захисне заземлення буде забезпечувати електробезпеку будівлі, так як виконується умова: $R_{\text{заг}} < 4 \text{ Ом}$, а саме:

$$R_{\text{заг}} = \frac{18,5 \cdot 8,1}{18,5 \cdot 0,3 + 8,1 \cdot 16 \cdot 0,57} = 1,9 \leq R_0$$

При виникненню пожеж при роботі на ПЕОМ від таких можливими джерел запалювання як:

- іскри і дуги коротких замикань;
- перегрів провідників, резисторів та інших радіодеталей ПЕОМ, від тривалої перевантаження та наявності перехідного опору;
- іскри при розмиканні і розмиканні ланцюгів;
- розряди статичної електрики;
- необережному поводженню з вогнем, а також вибухи газоповітряних і пароповітряних сумішей.

Важливу увагу слід звернути на пожежну безпеку підприємства в цілому і окремих його приміщень. В приміщеннях не повинно накопичуватися сміття, непотрібний папір, мотлох та ін. речі, які не використовуються у виробничому процесі. Наявний вільний аварійний вихід за межі приміщення в разі пожежі, бути передбачені вогнегасники. Вони повинні бути в робочому стані і перевірятися згідно з нормами. У приміщеннях повинна бути пожежна сигналізація, вогнегасник.

У разі виникнення пожежі необхідно повідомити в найближчу пожежну частину, убезпечити інших працівників і по можливості прийняти кроки по запобіганню можливих наслідків та усуненню пожежі.

4.6 Висновки до розділу "Охорони праці"

В результаті проведеної роботи було зроблено аналіз умов праці, шкідливих та небезпечних чинників, з якими стикається робітник. Було визначено параметри

і певні характеристики приміщення для роботи над запропонованим проектом написаному в кваліфікаційній роботі, описано, які заходи потрібно зробити для того, щоб дане приміщення відповідало необхідним нормам і було комфортним і безпечним для робітника.

Приведені рекомендації щодо організації робочого місця, а також важливу інформацію щодо пожежної та електробезпеки. Були наведені розміри приміщення та наведено значення температури, вологості й рухливості повітря, необхідна кількість і потужність ламп та інші параметри, значення яких впливає на умови праці робітника, а також – наведені інструкції з охорони праці, техніки безпеки при роботі на комп'ютері.

ВИСНОВКИ

У дипломній роботі було поставлено технічне завдання: розробити ІКС футбольного клубу. В процесі розробки було реалізовано всі необхідні функції, котрі були вказані в технічному завданні.

Реалізація необхідних функцій інформаційного ресурсу була зроблена за допомогою мов web-програмування, таких як HTML, PHP, jQuery, AJAX, MySQL.

ІКС має наступні функції:

- а) ознайомлення з новинами організації представленої на ресурсі;
- б) перегляд статистичних показників чемпіонату України з футболу;
- в) перегляд інформації про склад футбольної команди "ФК Шахтар Донецьк";
- г) перегляд таблиці всіх матчів футбольного сезону команди "ФК Шахтар Донецьк";
- д) форма зворотнього зв'язку з адміністрацією ресурсу.

На момент захисту дипломної роботи поставлене технічне завдання було реалізовано в повному обсязі.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. "Класифікація веб-сайтів" [Електронний ресурс] URL:http://life-prog.ru/ukr/1_2657_klasifikatsiya-veb-saytiv.html.
2. "Сайт" [Електронний ресурс] //Вікіпедія: вільна енциклопедія. URL: <https://ru.wikipedia.org/wiki/Сайт>.
3. "Web-сайт, загальні поняття" [Електронний ресурс] URL: <http://www.desnet.ru/Website>
4. "WAMP" [Електронний ресурс] //Вікіпедія: вільна енциклопедія. URL: <https://uk.wikipedia.org/wiki/WAMP>.
5. "Выбираем WAMP платформу для разработки сайтов под Windows" [Електронний ресурс] URL: <https://habrahabr.ru/post/144242/>.
6. Дакетт Джон HTML і CSS. Розробка і дизайн веб-сайтів (+ CD-ROM); Ексмо - Москва, 2013. - 480 с.
7. Макфарланд Д. "Новая большая книга CSS".- СПб.: Питер, 2016. — 720с.: ил. — (Серия «Бестселлеры O'Reilly»).
8. Прохоренок П.А "HTML, JavaScript, PHP и MySQL. Джентельменский набор Web-мастера". - 4-е изд., перераб. и доп. / Н.А. Прохоренок, Н. А. Дронов. - СПб.: БХВ-Петербург, 2015. - 768с.: ил. - (Профессиональное программирование)
9. Зандстра Мэтт "PHP: объекты, шаблоны и методики программирования", 4-е изд . Пер. с англ. - М.: ООО "И.Д. Вильямс", 2015. - 576 с.: ил. - Парал. тит. англ.
10. Бенедетти Р., Крэнли Р. "Изучаем работу с jQuery". — СПб.: Питер, 2012. — 528 с.: ил.
11. Дари К., Бринзаре Б., ЧерчерТоза Ф., Бусика М. "AJAX и PHP: разработка динамических веб-приложений". – СПб.: Символ - Плюс, 2007. – 336с., ил.
12. Закон України «Про охорону праці».

13. Закон України «Про забезпечення санітарного та епідемічного благополуччя населення».

14. Закон України «Про захист населення і територій від надзвичайних ситуацій техногенного та природного характеру».

15. Кодекс законів про працю України

16. Пожаровзрывоопасность веществ и материалов и средства их тушения. Справочник, Под ред. Баратова А.Н., в 2-х томах, М.: Химия, 1990.

17. СанПиН 4947-89 Временные санитарно-гигиенические требования к кооперативам и индивидуальной трудовой деятельности по производству, ремонту и наладке промышленных изделий и бытовой техники, а также в сфере кустарно-ремесленных промыслов.

18. Справочная книга для проектирования электрического освещения / Под ред. Г.М. Кнорринга.- М.: Энергия, 1976. - 384 с.

19. Справочная книга по светотехнике / Под ред. Ю. Б. Айзенберга. 3-е изд. перераб. и доп. — М.: Знак. 2006.— 972 с.

20. НПАОП 0.00-4.12-05 Типове положення про порядок проведення навчання і перевірки знань з питань охорони праці.

21. НПАОП 0.00-4.15-98 Про розробку інструкцій з охорони праці.

22. ДСН 3.3.6.042-99 Санітарні норми мікроклімату виробничих приміщень.

23. ДСН 3.3.6.037-99 Санітарні норми виробничого шуму, ультразвуку та інфразвуку.

24. ДСН 3.3.6.039-99 Санітарні норми виробничої загальної та локальної вібрації.

25. ДСТУ ГОСТ 12.1.012-90 ССБТ. Вибрационная безопасность. Общие требования.

26. ГОСТ 12.1.006-84 ССБТ. Электромагнитные поля радиочастот. Общие требования безопасности. Допустимые уровни на рабочих местах и требования к проведению контроля.

27. ГОСТ 12.1.030-81 ССБТ. Электробезопасность. Защитное заземление. Зануление.

28. ГОСТ 13109-97 „Электрическая энергия. Совместимость технических средств электромагнитных. Нормы качества электроэнергоснабжения общего назначения”.

29. ДСанПіН 3.3.2.007-98 Гігієнічні вимоги до організації роботи з візуальними дисплейними терміналами електронно-обчислювальних машин.

30. НПАОП 0.00-1.28-10 Правила охорони праці під час експлуатації електронно - обчислювальних машин.

ДОДАТОК А. Лістинг веб-порталу

Лістинг файлу "index.php"

```
<!DOCTYPE HTML>
<html>
<head>
<link href="style.css" rel="stylesheet">
<?php
$title = "Шахтёр Донецк";
include_once "head.php" ?>
</head>
<body>
<?php
include_once "header.php"?>
<?php
include_once "content.php";
include_once "leftblock.php";
include_once "footer.php"
?>
</body>
</html>
```

Лістинг файлу "header.php"

```
<html>
<head>
</head>
<body>
<div class="header">
<div class="logo">
<ul>
<li><a href="/">Главная</a></li>
```

```

<li><a href="/comanda.php">Команда</a></li>
<li><a href="/match.php">Матчи</a></li>
<li><a href="/club.php">Календарь</a></li>
<li><a href="/galery.php">Фотогалерея</a></li>
</ul>
<h1 style="
padding-left: 172px;
padding-top: 50px;
color: white;
">Шахтёр Донецк</h1>
</div>
</div>
</body>
</html>

```

Лістинг файлу "content.php"

```

<html>
<head>
<?php
require_once "connection.php";
        $news = getNews(3,0);
?>
</head>
<body>
<div class="content">
<?php
for ($i = 0; $i < count($news); $i++) {
if($i == 0)
echo "<div class=\"bigArticle\">";
else echo "<div class=\"article\">";
echo  '
<h2>'. $news[$i]["title"].'</h2>
<p>'. $news[$i]["intro_text"].'</p>

```

```

<div class="more">
<a href="/article.php?id='.$news[$i]["id"]."'>Далее</a></div>
</div>;
if($i ==0)
echo "<div class=\"clear\"><br></div>"; } ?>
</div>
</div>
</body>
</html>

```

Лістинг файлу "connection.php"

```

<?php
$mysqli = false;
function connectDB () {
    global $mysqli;
    $mysqli = new mysqli("localhost", "root", "", "shvachka");
    $mysqli->query("SET NAMES 'utf-8'");
}
function closeDB () {
    global $mysqli;
    $mysqli->close ();
}
function getNews ($limit, $id) {
    global $mysqli;
    connectDB();
    if($id)
        $where = "WHERE `id` = ".$id;
    $result = $mysqli->query("SELECT * FROM `news`$where ORDER BY `id` DESC LIMIT
$limit");
    closeDB();
    if(!$id)
        return resultToArray ($result);
    else
        return $result->fetch_assoc(); }

```

```
function resultToArray ($result) {
$array = array ();
while (($row = $result->fetch_assoc()) != false)
$array[] = $row;
return $array; } ?>
```

Лістинг файлу "footer.php"

```
<div class="footer">
<div class="Onas">
<a href="about.php">О нас
</div>
</a>
<div class="feadback">
<a href="feadback.php">Обратная связь
</div>
</a>
<div class="social">
<a href="https://vk.com/alexandr_alexandrowi4" title="Разработчик" target="_blank">

</a>
<a href="http://shakhtar.com/ru/" title="официальный сайт ФК Шахтёр" target="_blank">

</a>
<div class="material">
Материалы взяты с сайта ФК Шахтёр
</div>
</div>
<div class="Prava">
Все права защищены &copy; <?php echo date ('Y')?>
</div>
</div>
```

Лістинг файлу "club.php"

```

<!DOCTYPE HTML>
<html>
<head>
<link rel="stylesheet" href="styletable.css" type="text/css">
<script src="http://code.jquery.com/jquery-1.9.1.js"></script>
<?php
$title = "Турнирная таблица";
include_once "head.php" ?>
<?php
$sdd_db_host='localhost';
$sdd_db_name='shvachka';
$sdd_db_user='root';
$sdd_db_pass="";
@mysql_connect($sdd_db_host,$sdd_db_user,$sdd_db_pass);
@mysql_select_db($sdd_db_name);
?>
</head>
<body>
<?php
include_once "header.php"?>
<div class="table" >
<h1>Введите номер тура:</h1>
<form action="" method="post">
<input type="text" class="filterr" name="name" >
<input type='submit' name='insert' value='Показать' style="margin-left:47%;margin-top:10px;
padding: 10px">
</form>
<?php
$name = $_POST['name'];
if(isset($_POST['insert']))
{
$result = mysql_query("SELECT * FROM `rezultat` WHERE nomer_tura = $name");
while($row = mysql_fetch_array($result)){

```

```

    echo '<h2>Место проведения: '.$row['mesto_prov'].'</h2>';
    echo '<h2>'.$row['com_1'].' : '.$row['com_2'].'</h2>';
    echo '<h2>Счёт '.$row['schet'].'</h2>';
    echo '<h2>Результат: '.$row['rezultat'].'</h2>';
    echo '<h2>Судья: '.$row['sudja'].'</h2>';
}
$resultt = mysql_query("SELECT * FROM `calendar` WHERE nomer_tura = $name");
while($roww = mysql_fetch_array($resultt)){
    echo '<h2>Дата тура: '.$roww['data_tura'].'</h2>';
}
}
?>
</div>
<?php
include_once "leftblock.php";
include_once "footer.php"
?>
</body>
</html>

```

Лістинг файлу "feadback.php"

```

<!DOCTYPE HTML>
<html>
<head>
<?php
$title = "Обратная связь";
require_once "head.php"
?>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/2.0.3/jquery.min.js?"></script>
<script>
$(document).ready (function () {
    $("#done").click (function() {
        $("#messageShow").hide ();
        var name = $("#name").val();

```



```

var email = $("#email").val();
var subject = $("#subject").val();
var message = $("#message").val();
var fail = "";
if (name.length< 3) fail = "Имя не менее 3 символов";
else if (email.split('@').length - 1 == 0 || email.split('.').length-1 == 0)
fail = "Вы ввели некорректный email";
else if (subject.length< 5)
fail = "Тема сообщения не менее 5 символов";
else if (message.length< 20)
fail = "Сообщение не менее 20 символов";
if(fail != ""){
$("#messageShow").html (fail+"<div class='clear'><br></div>");
$("#messageShow").show ();
return false;
}
$.ajax ({
url:'/ajax/feedback.php',
type: 'POST',
cache: false,
data: {'name': name, 'email': email, 'subject': subject, 'message': message},
dataType: 'html',
success: function (data) {
$("#messageShow").html (data+"<div class='clear'><br></div>");
$("#messageShow").show ();
}});}); });
</script>
</head>
<?php include_once "header.php"?>
<div class="feedbackk">
<input type="text" placeholder="Имя" id="name" name="name"><br />
<input type="text" placeholder="Email" id="email" name="email"><br />
<input type="text" placeholder="Тема сообщения" id="subject" name="subject"><br />
<textarea name="message" placeholder="Введите сюда Ваше сообщение"
id="message"></textarea><br />

```

```

<div id="messageShow"></div>
<input type="button" name="done" id="done" value="Отправить">
</div>
<?php include_once "leftblock.php"?>
<?php include_once "footer.php"?>
</html>

```

Лістинг файлу "article.php"

```

<!DOCTYPE HTML>
<html>
<head>
<?php
require_once "connection.php";
$news = getNews(1, $_GET[id]);
$title = $news["title"];
require_once "head.php"
?>
</head>
<?php include_once "header.php"?>
<div class="content">
<?php
echo '<div class="bigArticle">

<h2>'.$news["title"].'</h2>
<p>'.$news["full_text"].'</p>
</div>';
?>
</div>
<?php include_once "leftblock.php"?>
<?php include_once "footer.php"?>
</html>

```

Частина лістингу файлу "style.css"

```
@charset "UTF-8";
#wrap {
max-width: 1440px;
min-width: 520px;}
a { text-decoration: none;}
body {
margin-left: 0px;
margin-right: 0px;
margin-bottom: 0px;
margin-top: 0px;
background-color: #ffffff;
background-image: url(../img/Arena.jpg);
background-position: center center;
background-repeat: no-repeat;
background-attachment: fixed;
background-color:#eaeaea;
background-size: cover;}
/* Стили для хедера*/
div.header {
width: 100%;
height: 182px;
margin-top: 1px;
position: relative;
top: 0px;
left: 0px;
background: url(../img/image1.png);
background-repeat: no-repeat;
width: 100%;
height: 180px;
opacity: 0.95;
filter: alpha(opacity=95);
background-size: cover;}
div.logo { width: 50%;
```

```
height: 178px;
background: url(../img/image2.png);
background-repeat: no-repeat;
margin-left: 18%;}
div.leftblock {
padding: 15px;
display: inline-block;
float: right;
width: 20%;
margin-right: 1%;
margin-top: 15px;
border-radius: 20px;
background-color: #eaeaea;
opacity: 0.8;
filter: alpha(opacity=80);
background-size: cover;}
/* Стили для контента*/
div.content {
display: inline-block;
padding: 15px;
margin-top: 15px;
float: left;
width: 74%;
border-radius: 20px;
background-color: #eaeaea;}
div.bigArticle, div.article {
width: 68%;
margin-left: 15%;
background-color: #fafafa;
padding: 1%;
margin-bottom: 10px;
border-radius: 10px;}
div.bigArticle img, div.article img {
width: 100%;
float: left;}
```

```
div.bigArticle h2, div.article h2 {
font-weight: normal;
font-size: 1.4em;
color: #000000;
font-family: Comic Sans MS, sans-serif;}
div.bigArticle p , div.article p{
width: 90%;
margin-left: 5%;}
div.more {
width: 10%;
text-align: center;
padding: 5px;
border-radius: 5px;
background-color: #878787;
color: #fafafa;}
div.more:hover {
background-color: #a5a5a5;}
div.more a {
color: black;}
div.article{
display: inline-block;
width: 40%;
margin-left:5%;
background-color:#fafafa;
padding:1%;
margin-bottom: 10px;
border-radius: 10px;}
/*Стили для странички О НАС*/
div.about {
display: inline-block;
width: 54%;
padding: 15px;
margin-top:15px;
margin-left:10%;
float: left;
```

ДОДАТОК Б. Слайди презентації

У Додатку Б наведено зображення слайдів презентації.

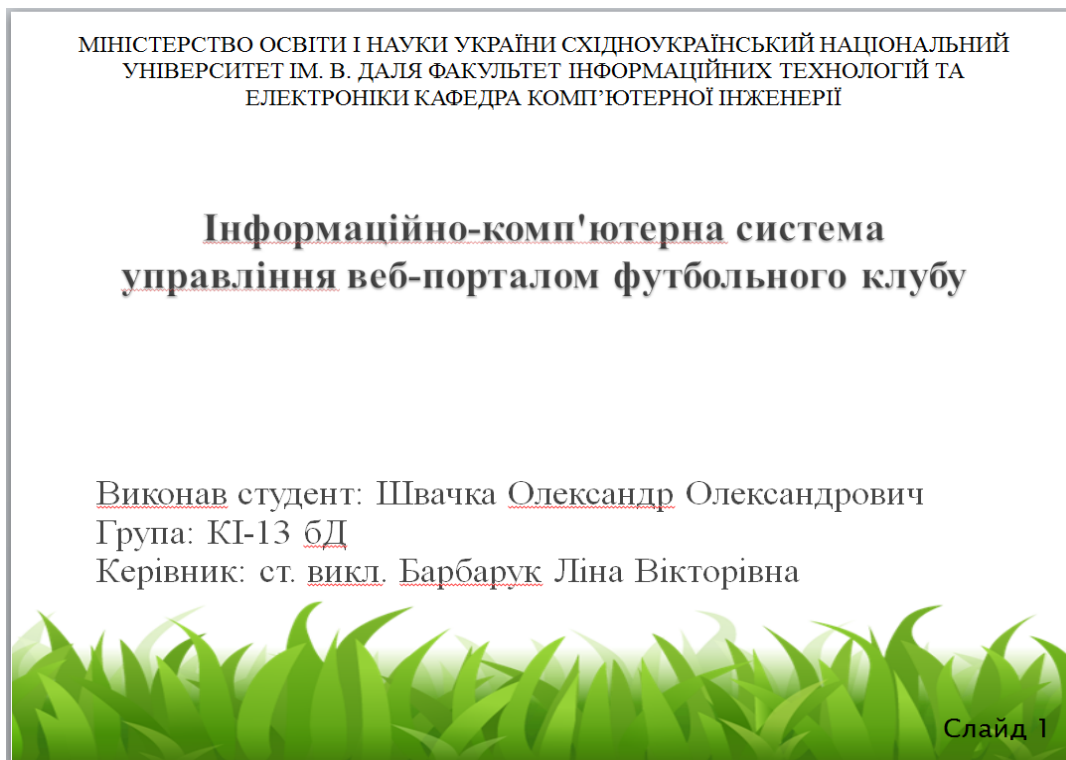


Рисунок Б.1 - Слайд 1

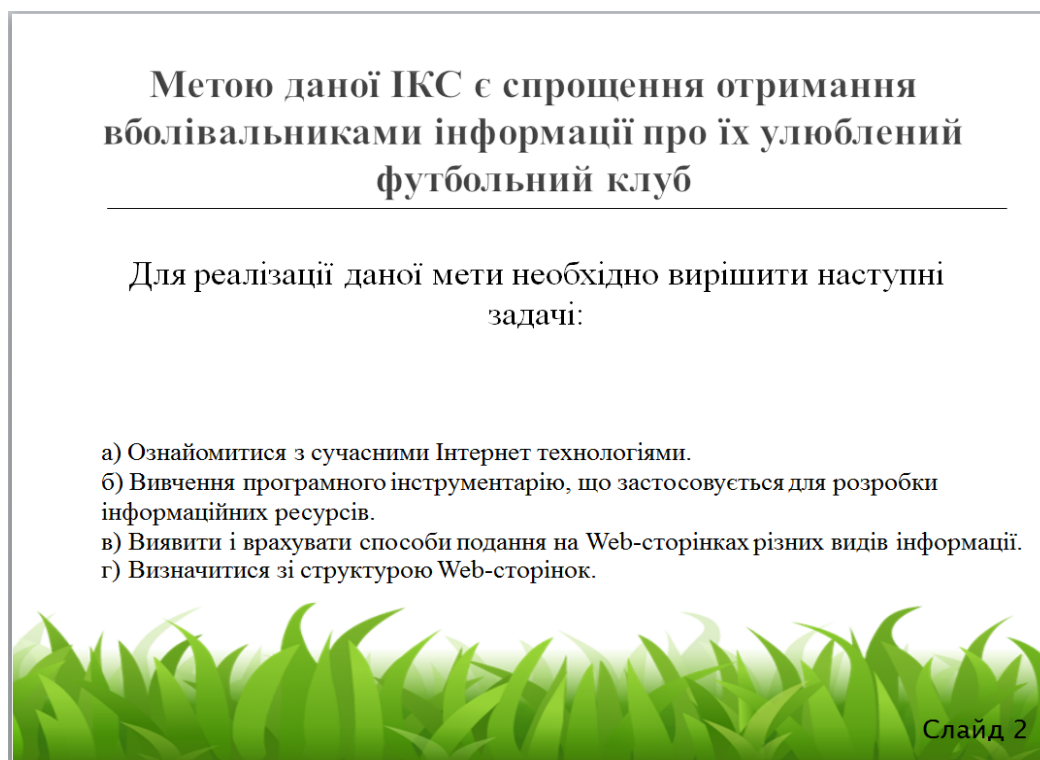


Рисунок Б.2 - Слайд 2

Постановка завдання на розробку ІКС

Однією із задач, яка буде розглядатися в даній дипломній роботі – задача взаємодії користувачів інтернету та інформаційного ресурсу футбольного клубу України "Шахтар Донецьк"

У дипломній роботі потрібно створити інформаційний ресурс, який би надавав користувачам інтернету такі можливості:

- а) перегляд новин улюбленого футбольного клубу;
- б) перегляд результатів усіх матчів футбольного сезону;
- в) ознайомлення зі складом футбольного клубу;
- г) ознайомлення з статистичними показниками чемпіонату України з футболу;
- д) подача своїх пропозицій щодо вдосконалення web-ресурсу.

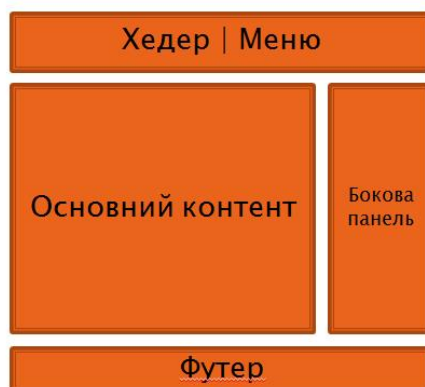
Слайд 3

Рисунок Б.3 - Слайд 3

Вирішення поставленої задачі

ІКС – являє собою web-додаток

Макет web-додатку:



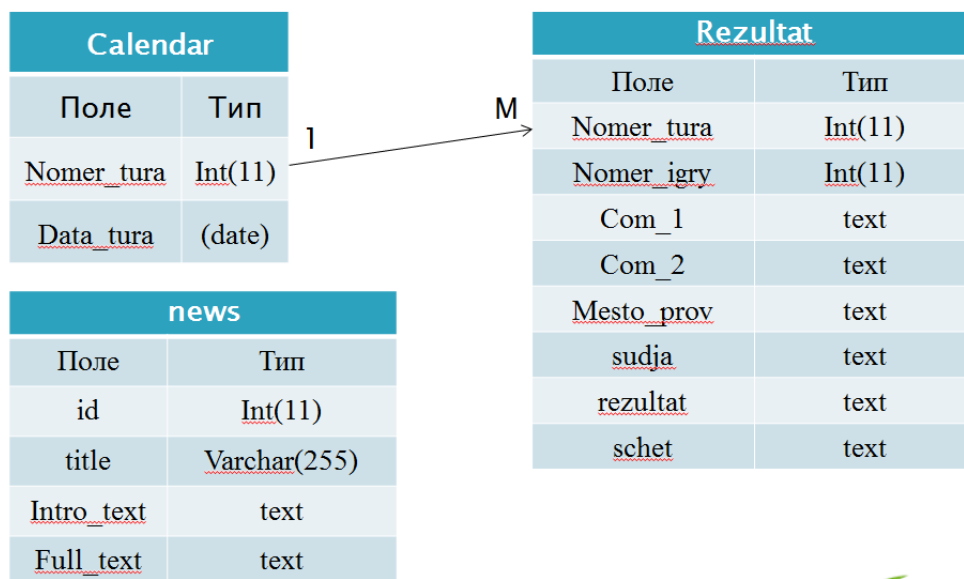
Засоби для розробки:

HTML
PHP
MySQL
jQuery
AJAX

Слайд 4

Рисунок Б.4 - Слайд 4

Структура Баз даних для ІКС



Слайд 5

Рисунок Б.5 - Слайд 5

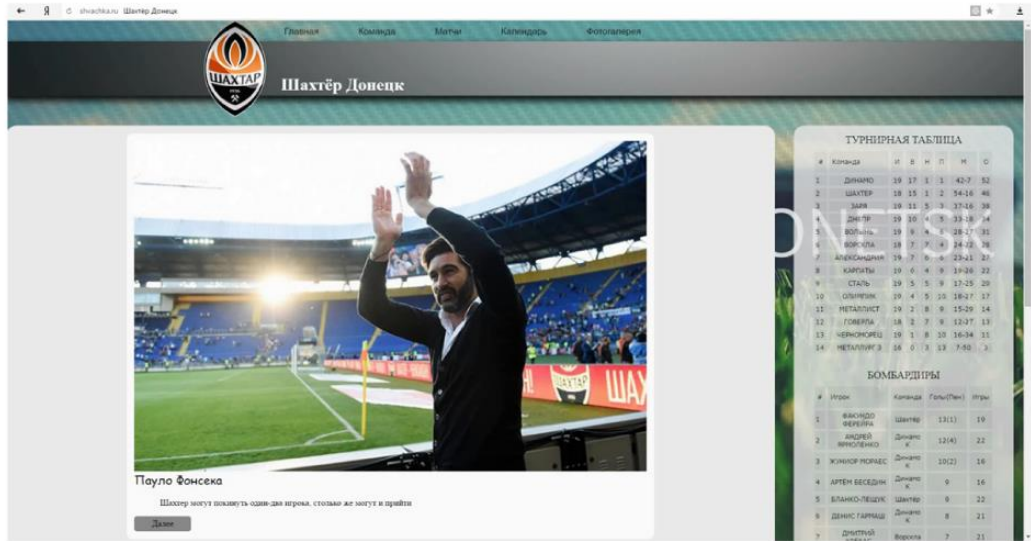
Функціональні вимоги до інформаційного ресурсу

- Головна сторінка повинна відображати новини футбольного клубу Шахтар Донецьк.
- Бокова панель ресурсу повинна відображати таблиці зі статистичними показниками Чемпіона України з футболу.
- Хедер ресурсу має містити меню всього ресурсу, логотип ресурсу та заголовок.
- Футер ресурсу має містити посилання на сторінку зворотнього зв'язку користувача з адміністрацією ресурсу.
- Сторінка "команда" має містити таблицю з усіма гравцями команди Шахтар Донецьк.
- Сторінка "матчи" повинна містити інформацію про всі матчі команди в поточному футбольному сезоні.
- Сторінка "календарь туров" повинна надавати користувачу детальну інформацію про тур чемпіонату з футболу України.
- Сторінка "фотоальбом" має містити галерею з фотографіями футболістів та тренера команди Шахтар Донецьк.
- Повинна бути створена База даних для ІКС.

Слайд 6

Рисунок Б.6 - Слайд 6

Розробка головної сторінки



Слайд 7

Рисунок Б.7 - Слайд 7

Запит до таблиці з новинами у Базі даних

```

1 <?php
2 $mysqli = false;
3 function connectDB () {
4     global $mysqli;
5     $mysqli = new mysqli("localhost", "root", "", "shvachka");
6     $mysqli->query("SET NAMES 'utf-8'");
7 }
8
9 function closeDB () {
10    global $mysqli;
11    $mysqli->close ();
12 }
13
14 function getNews ($limit, $id) {
15    global $mysqli;
16    connectDB();
17    if($id)
18        $where = "WHERE 'id' = ".$id;
19    $result = $mysqli->query("SELECT * FROM 'news' $where ORDER BY 'id' DESC LIMIT $limit");
20    closeDB();
21    if(!$id)
22        return resultToArray ($result);
23    else
24        return $result->fetch_assoc();
25 }
26
27 function resultToArray ($result) {
28    $sarray = array ();
29    while (($row = $result->fetch_assoc()) != false)
30        $sarray[] = $row;
31    return $sarray;
32 }
33 ?>

```

Слайд 8

Рисунок Б.8 - Слайд 8

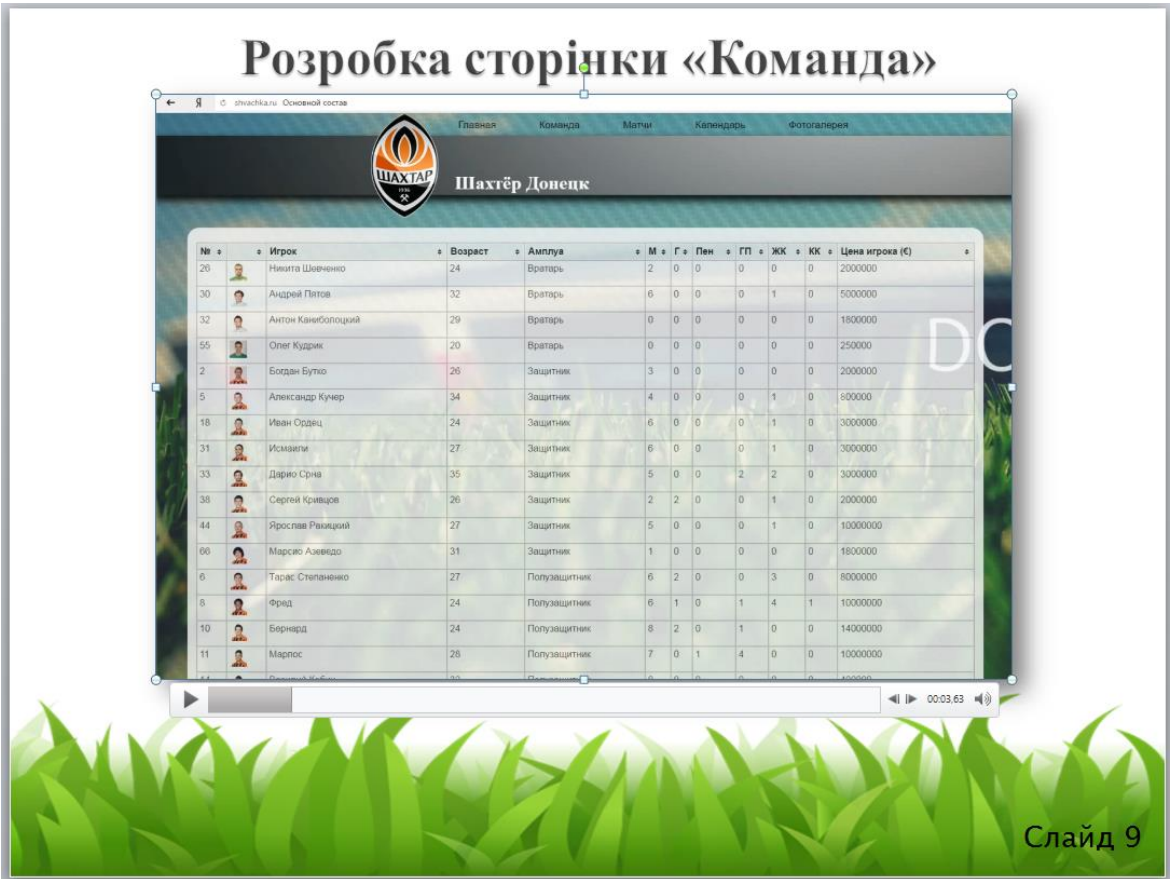


Рисунок Б.9 - Слайд 9

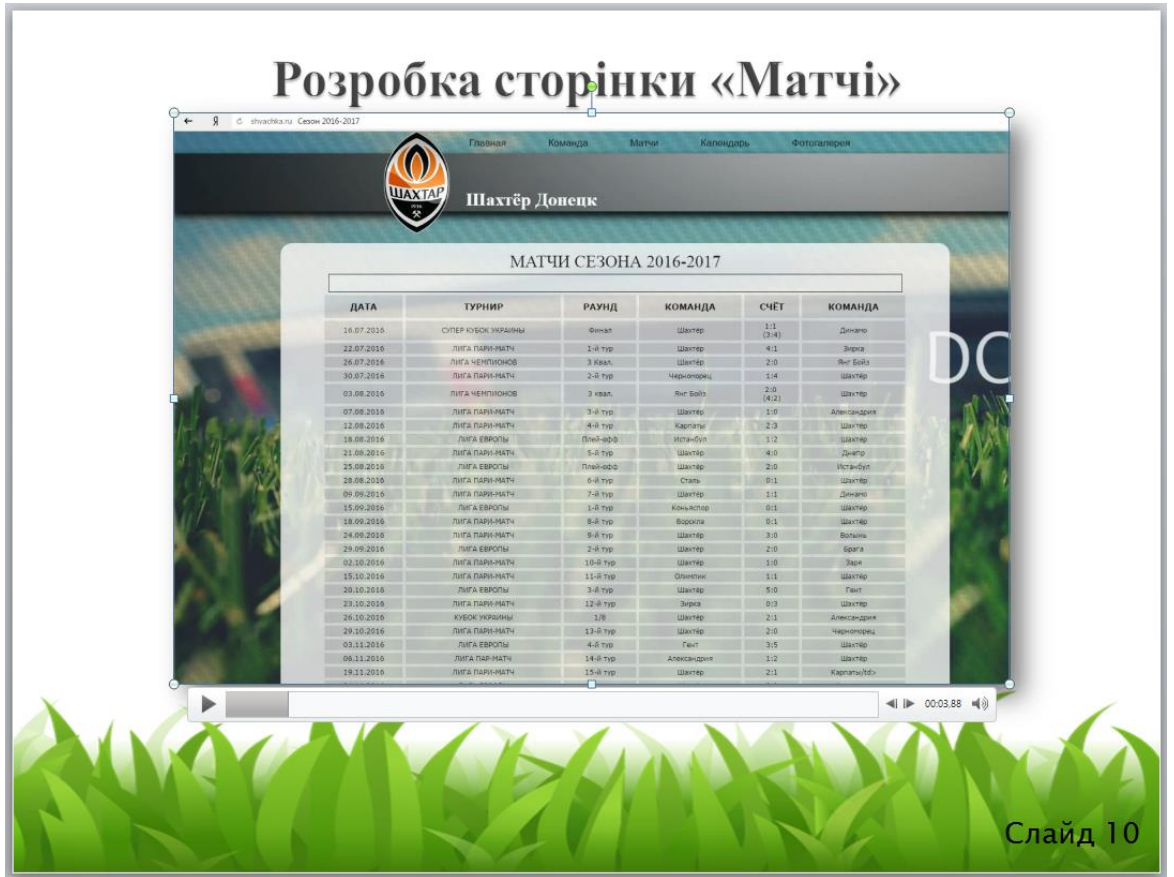
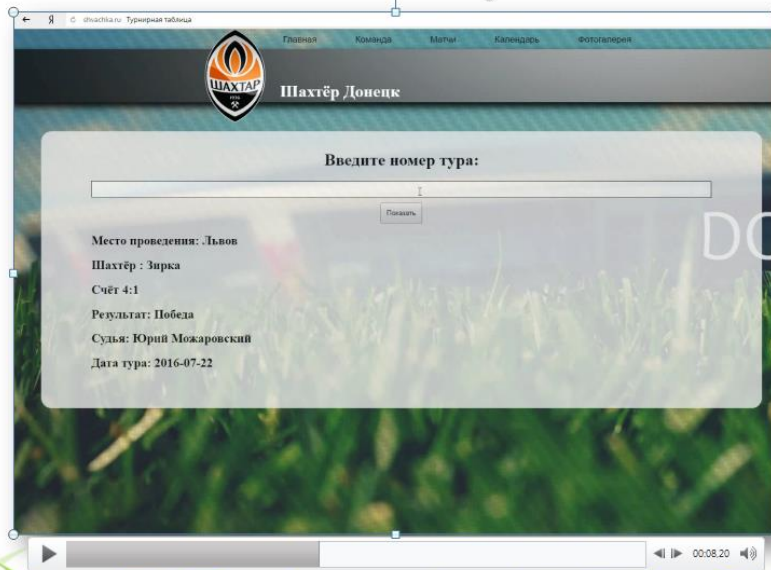


Рисунок Б.10 - Слайд 10

Розробка сторінки «Календар турів чемпіонату»



Слайд 11

Рисунок Б.11 - Слайд 11

Запит до таблиці з результатами турів у Базі даних

```

37 <?php
38 $name = $_POST['name'];
39
40 if(isset($_POST['insert']))
41 {
42     $result = mysql_query("SELECT * FROM `rezultat` WHERE nomer_tura = $name");
43     while($row = mysql_fetch_array($result)){
44         echo '<h2>Место проведения: ' . $row['mesto_prov'] . '</h2>';
45         echo '<h2>' . $row['com_1'] . ' : ' . $row['com_2'] . '</h2>';
46         echo '<h2>Счёт ' . $row['schet'] . '</h2>';
47         echo '<h2>Результат: ' . $row['rezultat'] . '</h2>';
48         echo '<h2>Судья: ' . $row['sudja'] . '</h2>';
49     }
50     $resultt = mysql_query("SELECT * FROM `calendar` WHERE nomer_tura = $name");
51     while($roww = mysql_fetch_array($resultt)){
52         echo '<h2>Дата тура: ' . $roww['data_tura'] . '</h2>';
53     }
54 }
55 }
56 }
57 ?>

```

Слайд 12

Рисунок Б.12 - Слайд 12

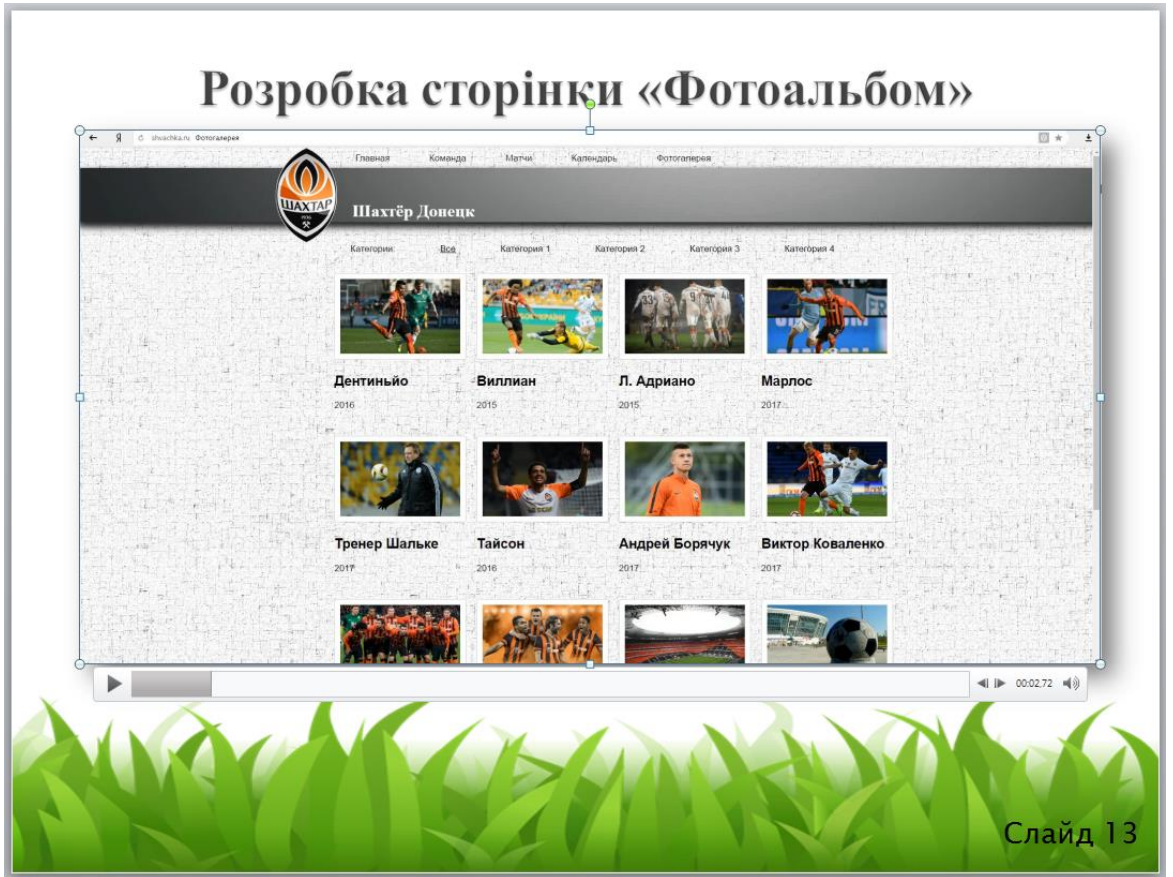


Рисунок Б.13 - Слайд 13

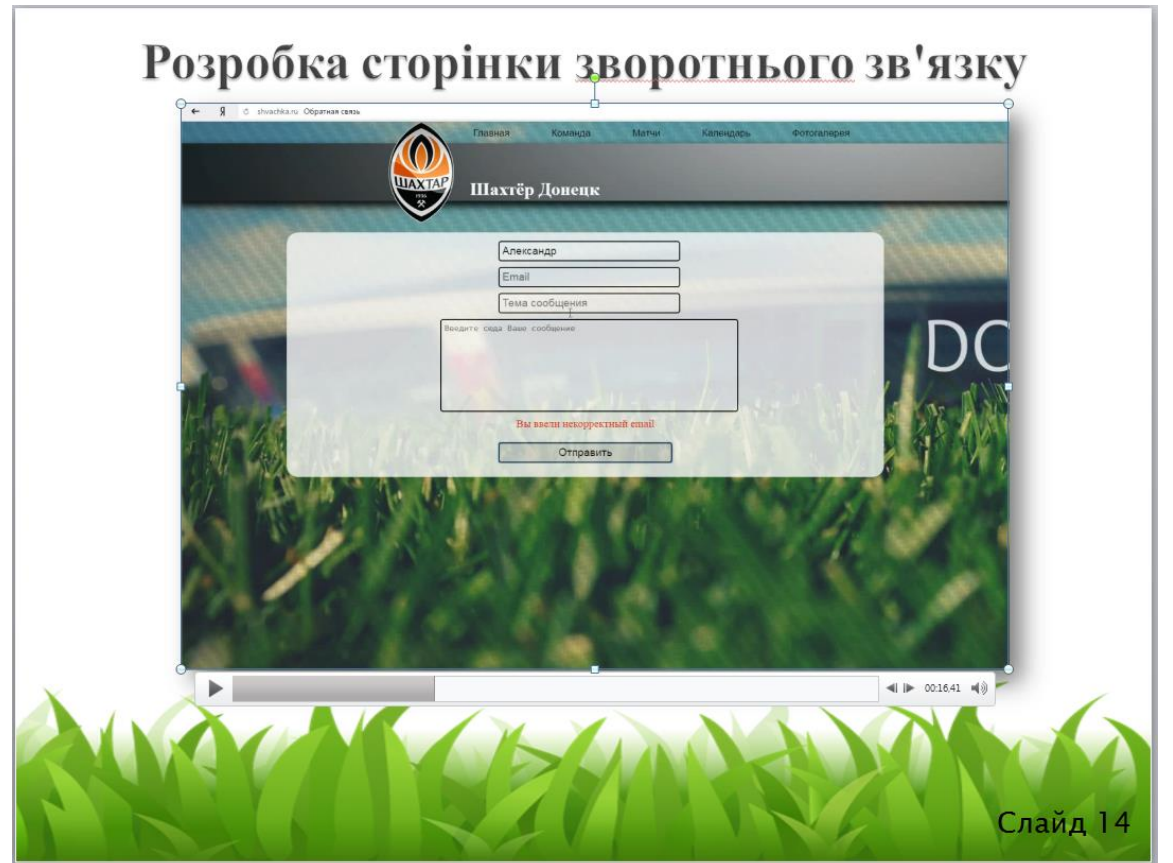



Рисунок Б.14 - Слайд 14

Висновки

У дипломній роботі було поставлено технічне завдання: розробити ІКС футбольного клубу. В процесі розробки було реалізовано всі необхідні функції, котрі були вказані в технічному завданні.

Реалізація необхідних функцій ресурсу була зроблена за допомогою мов web-програмування, таких як HTML, PHP, jQuery, AJAX, MySQL.



Слайд 15

Рисунок Б.15 - Слайд 15