

ВИПУСКНИЙ КВАЛІФІКАЦІЙНИЙ ПРОЕКТ

на тему:

Розробка інформаційної системи Консультативно-діагностичного центру №2 Дарницького району м. Києва

Студента 2 курсу, 5м групи,
Спеціальності 121 «Інженерія
програмного забезпечення
Спеціалізації «Інженерія
програмного забезпечення»
Науковий керівник,
кандидат технічних наук,
доцент

підпис студента

Кінзерського
Костянтина Юрійовича

підпис керівника

Рзаєва
Світлана Леонідівна

Гарант освітньої програми,
доктор технічних наук,
професор

підпис керівника

Криворучко
Олена Володимирівна

Київ 2019

АНОТАЦІЯ

До випускної кваліфікаційної роботи Кінзерського Костянтина Юрійовича на тему «Розробка інформаційної системи консультативно-діагностичного центру»

У випускній кваліфікаційній роботі розглядається розробка клієнт-серверної інформаційної системи в який входить мова програмування PHP, фреймворку Bootstrap, та база даних MySQL .

Ключові слова: інформаційна система, мова програмування, PHP, javascript, база даних MySQL.

ANNOTATION

Prior to the final qualification work of Kinzerskyi Kostiantyn Yuriyovuch on the topic " Development of information system of consulting and diagnostic center"

The final qualification examines the development of a client-server information system that includes PHP programming language, Bootstrap framework, and MySQL database.

Keywords: information system, programming language, PHP, javascript, MySQL database.

Київський національний торговельно-економічний університет

**Інформаційна система Консультативно-діагностичного
центру**

ТЕХНІЧНЕ ЗАВДАННЯ

на 2 аркушах

ТЗ

1. НАЙМЕНУВАННЯ ТА ГАЛУЗЬ ЗАСТОСУВАННЯ

Назва розробки: Інформаційна система Консультативно-діагностичного центру.

Галузь застосування: Медицина.

2. ПІДСТАВА ДЛЯ РОЗРОБКИ

Підставою для розробки є завдання на ВКР, затверджене кафедрою програмної інженерії та кібербезпеки КНТЕУ.

3. ПРИЗНАЧЕННЯ РОЗРОБКИ

3.1. Призначення додатку.

Основним призначенням є створення інформаційної системи для Консультативно-діагностичного центру, що дає можливість пацієнтами записуватись на прийом до потрібного лікаря онлайн, лікарям відслідковувати історію лікування та доповнювати її.

3.2. Мета інформаційної системи.

Метою є створення інформаційної системи з можливістю використання певного функціоналу.

3.3. Цільова аудиторія додатку.

Цільова аудиторія інформаційної системи представлена наступними групами користувачів:

- Пацієнти
- Лікарі

4. ВИМОГИ ДО ПРОГРАМНОГО ПРОДУКТУ

Додаток повинен забезпечувати такі основні функції:

- 1) Пацієнти має можливість самостійно зареєструватись в системі та записатися до лікаря;
- 2) Лікар відслідковує усі записи до нього, веде медичну історію пацієнтів, має можливість вносити в базу пацієнтів.
- 3) Адміністратор системи має керувати спеціалізаціями лікарів, додавати

в систему нових, переглядати інформацію, повідомлення, записи пацієнтів до лікарів.

Розробку виконано за допомогою редактору коду Sublime Text, переважно використовуючи мову програмування PHP та javascript, сервер налаштовано на Open Server, використана база даних MySQL. Інтерфейс реалізовано за допомогою HTML та CSS з використанням набору інструментів Bootstrap.

5. ВИМОГИ ДО ПРОЕКТНОЇ ДОКУМЕНТАЦІЇ

У процесі виконання роботи повинна бути розроблена програма та методика тестування.

6. ЕТАПИ ПРОЕКТУВАННЯ

Розробка концепції інформаційної системи.

Розробка технічного завдання

Проектування

Реалізація

Впровадження в експлуатацію

7. ПОРЯДОК ТЕСТУВАННЯ РОЗРОБКИ

Тестування розробленого програмного продукту виконується відповідно до “Програми та методики тестування”.

Зміст

ВСТУП	6
РОЗДІЛ 1	7
ТЕОРЕТИЧНІ ВІДОМОСТІ РОЗРОБКИ ТА ПОБУДОВИ ІНФОРМАЦІЙНОЇ СИСТЕМИ	7
1.1. Структура інформаційної системи	7
1.2. Класифікація інформаційних систем	8
1.3. Аналіз існуючих методів розробки	10
1.4. Висновки до розділу 1	13
РОЗДІЛ 2	14
АРХІТЕКТУРА ТА МОДЕЛЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	14
2.1. Програмне середовище розробки	14
2.2. Організаційна структура Інформаційної системи	17
2.3. Система керування інформаційною системою	18
2.4. FTP-менеджер FileZilla	19
2.5. Редактор коду Sublime Text	20
2.6. Локальний Веб-сервер Open Server	22
2.7. Висновки до розділу 2	23
РОЗДІЛ 3	24
СТВОРЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ КОНСУЛЬТАТИВНО-ДІАГНОСТИЧНОГО ЦЕНТРУ	24
3.1. Створення та підключення бази даних	24
3.2. Проектування бази даних	24
3.3. Інтерфейс адміністратора	32
3.4. Інтерфейс пацієнта	37
3.5. Інтерфейс лікаря	42
3.6. Висновок до розділу 3	44
ВИСНОВОК	45
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	46

Актуальність - в наш час створення інформаційної системи відіграють значну роль в успішності будь-якого підприємства. Термін "інформаційна система" належить до класу програмних продуктів, що автоматизують ведення бізнесу. Система називається інформаційною, якщо вона підтримує інформаційне забезпечення бізнесу.

Інформаційна система – це сукупність засобів збору, зберігання, передачі, оброблення інформації для досягнення поставленої мети у процесі управління.

Основні властивості поведінки інформаційних систем:

- Цілісність – поведінка окремих об'єктів розглядається з позиції структури всієї системи;
- Забезпечення стійкості функціонування системи;
- Адаптивність до змін зовнішнього середовища;
- Здатність до навчання шляхом зміни структури системи відповідно до зміни мети системи.

Для створення власної інформаційної системи неможливо обійтися без використання баз даних. Важливою особливістю СУБД є забезпечення виконання запитів до бази даних. Ще однією важливою особливістю більшості сучасних СУБД є забезпечення режиму мультидоступу.

Основні функції СУБД - безпосереднє керування даними у зовнішній пам'яті. Ця функція включає збереження та ведення структурної інформації (даних), перетворення даних за запитом на структурну інформацію для користувача. При цьому повинні забезпечуватися: простота використання системи; можливість автономного функціонування при порушеннях мережі або при адміністративних потребах; високий ступінь ефективності.

Мета дослідження – розробити інформаційну систему для консультативно-діагностичного центру.

Предметом випускної кваліфікаційної роботи є розробка інформаційної системи для консультативно-діагностичного центру.

Об'єктом є діяльність консультативно-діагностичного центру.

Задачі дослідження є – визначити функціональні можливості майбутньої інформаційної системи, вимоги до інтерфейсу.

Методи дослідження – Розроблено інформаційну систему для підвищення ефективності роботи консультативно-діагностичного центру.

РОЗДІЛ 1

ТЕОРЕТИЧНІ ВІДОМОСТІ РОЗРОБКИ ТА ПОБУДОВИ ІНФОРМАЦІЙНОЇ СИСТЕМИ

1.1. Структура інформаційної системи

Структуру інформаційної системи складає сукупність окремих її частин - підсистем. Підсистема - це частина системи, яка виділена за певною ознакою. Тому структура будь-якої інформаційної системи може бути представлена як сукупність підсистем, що забезпечують інформаційне, технічне, математичне, програмне, організаційне і правове забезпечення.

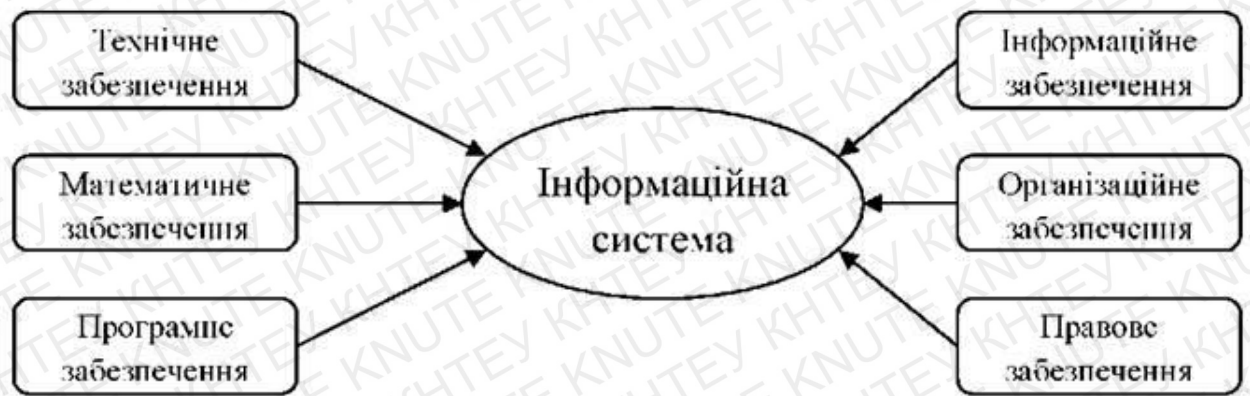


Рис. 1.1 Структура інформаційної системи

Інформаційне забезпечення - сукупність єдиної системи класифікації й кодування повідомлень, уніфікованих систем документації, схем інформаційних потоків, що циркулюють в організації, а також методологія побудови баз даних.

Технічне забезпечення - комплекс технічних засобів, призначених для роботи інформаційної системи, а також відповідна документація на ці засоби й технологічні процеси.

Математичне й програмне забезпечення - сукупність математичних методів, моделей, алгоритмів і програм для реалізації цілей і завдань інформаційної системи, а також нормального функціонування комплексу технічних засобів.

Організаційне забезпечення - сукупність методів і засобів, що регламентують взаємодію працівників з технічними засобами й між собою в процесі розробки й експлуатації інформаційної системи.

Правове забезпечення - сукупність правових норм, що визначають створення, юридичний статус і функціонування інформаційних систем, що регламентують порядок одержання, перетворення й використання відомостей.

1.2. Класифікація інформаційних систем

Інформаційні системи зазвичай класифікують за ступенем автоматизації (ручні, автоматичні, автоматизовані), характером використання відомостей (інформаційно-пошукові, інформаційно-управлінські), напрямом застосування

(організаційного управління, управління технологічними процесами, автоматизованого проектування), типом задач або за функціональною ознакою.



Рис. 1.2. Класифікація інформаційної системи

За типом задач інформаційні системи поділяються на системи, які призначаються для вирішення:

- структурованих задач, які можна відтворити в формі математичної моделі, що має точний алгоритм розв'язання;
- неструктурованих задач, для яких неможливо скласти математичну модель, розробити відповідний алгоритм і прийняти рішення без втручання людини;
- частково структурованих задач, в яких відома лише частина елементів і зв'язків між ними, іншу частину доповнює людина, що відіграє вирішальну роль.

Класифікація за функціональною ознакою здійснюється в залежності від призначення системи, а також її мети, завдань і функцій.

Структура інформаційної системи може бути представлена як сукупність її функціональних підсистем. Наприклад, інформаційні системи фахівців можна розділити за функціональною ознакою на дві групи: інформаційні системи офісної автоматизації та інформаційні системи обробки знань.

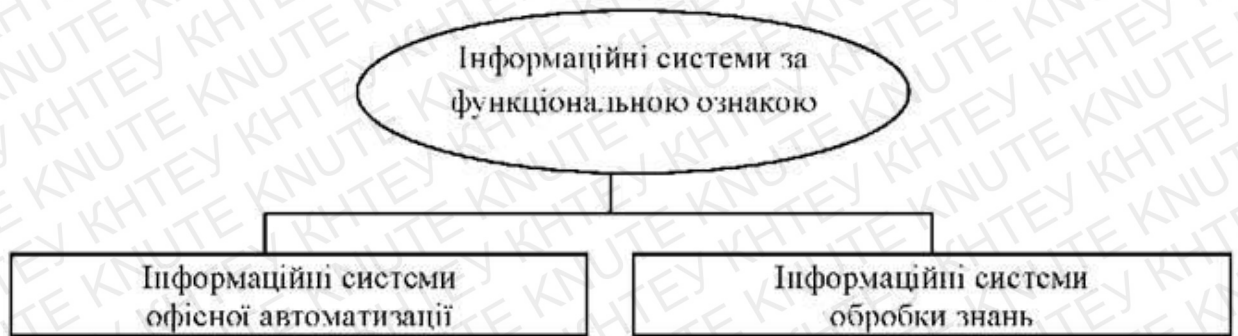


Рис. 1.3. Інформаційні системи за функціональною ознакою

Інформаційні системи офісної автоматизації внаслідок своєї простоти й багатoproфільності активно використовуються працівниками будь-якого організаційного рівня. Основна мета - обробка даних, підвищення ефективності роботи й спрощення канцелярської праці. Ці системи виконують такі функції:

- обробка відомостей на комп'ютерах засобами офісних комп'ютерних технологій;
- виробництво нових видів і форм даних;
- архівація і збереження матеріалів;
- електронні календарі й записні книжки для організації обліку й управління поточними даними;
- електронна, аудіо й відео пошта;
- відео і телеконференції тощо.

Інформаційні системи обробки знань, у тому числі й експертні системи, містять знання, необхідні людині при розробці або створенні нового інформаційного продукту. Їх робота полягає у створенні нових матеріалів й нового знання. Наприклад, спеціалізовані робочі станції з інженерного й наукового проектування технічних і наукових розробок.

1.3. Аналіз існуючих методів розробки

Bootstrap — це CSS/HTML-фреймворк для створення сайтів. Іншими словами, це набір інструментів для верстки. У ньому є ряд переваг, завдяки яким BS вважається найпопулярнішим серед собі подібних. Переваги бутстрапа:

- Швидкість роботи — завдяки безлічі готових елементів верстка з бутстрапа займає значно менше часу;
- Масштабованість — додавання нових елементів не порушує загальну структуру документів;
- Легко налаштовувати — редагування стилів проводиться шляхом створення нових CSS-правил, які виконуються замість стандартних. При цьому не потрібно використовувати атрибуту типу “! important”;
- Велика кількість шаблонів — про шаблони Bootstrap я напишу далі;
- Величезне співтовариство розробників;
- Широка сфера застосування — Bootstrap використовується в створенні тем для практично будь-яких CMS (OpenCart, Prestashop, Magento, Joomla, Bitrix, WordPress і будь-які інші), в тому числі для односторінкових додатків. Особливою популярністю користується Bootstrap для створення односторінкових або лендингів (landing page).

Шаблони в бутстрапа дозволяють редагувати вже змінені елементи під свої потреби. Безліч розробників пропонують свої шаблони (як платно, так і безкоштовно). Підключити шаблон в Bootstrap дуже легко: після підключення бутстрапа потрібно тільки додати виклик CSS вашого шаблону.

Блокова система в Bootstrap — основа грамотної верстки, потужний інструмент для блочного каркасу блоків контенту і будь-яких вкладених елементів. За допомогою префіксів ми можемо вказувати, як потрібно міняти відображення блоків в залежності від типу пристрою, на якому відображається сайт.

Наприклад, клас `.col-xs-` буде використано для телефонів з шириною екрану менше 768 пікселів, а `.col-lg-` — для будь-яких пристроїв з екраном ширше 1170 px. BS розбиває ширину батьківського блоку на 12 рівних частин, які ми можемо використовувати як завгодно. Частини можна об'єднувати, отримуючи, наприклад, три колонки: дві по 25% `.col-lg-3` і одну на 50% `.col-lg-6`.

Візуально сторінка може бути представлена як завгодно:

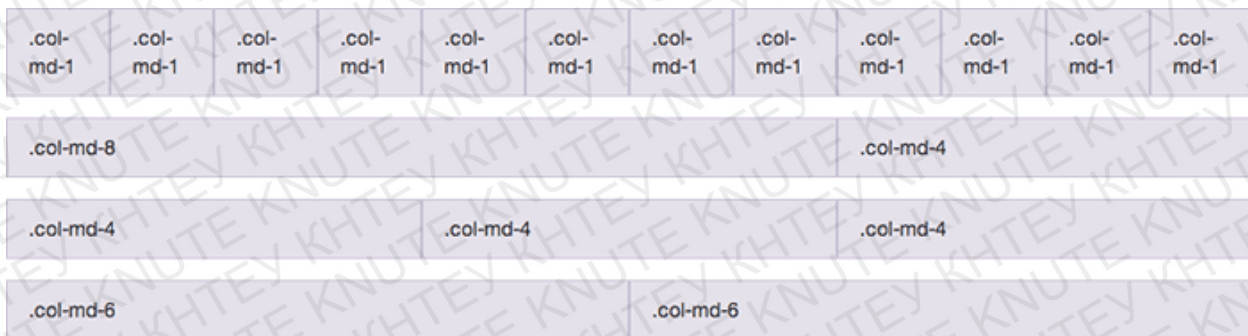


Рис. 1.4. Візуалізація блокової системи Bootstrap

Якщо вийти за межі 12 частин, блоки будуть відображатися один під одним.

PHP інтерпретується веб-сервером у HTML код, який передається на сторону клієнта. На відміну від скриптової мови JavaScript, користувач не бачить PHP коду, бо браузер отримує готовий HTML код. Це є перевага з точки зору безпеки, але погіршує інтерактивність сторінок. Але ніщо не забороняє використовувати PHP для генерування і JavaScript кодів які виконуються вже на стороні клієнта.

До базових типів належать булеві дані, цілі та дійсні числа із плаваючою комою, а також рядки. Булеві дані виражають істинність значення. Цілі числа можуть бути подані у вісімковому, десятковому та шістнадцятковому вигляді. Розмір цілого числа може змінюватись залежно від платформи, зазвичай, розрядність становить 32 біти. PHP не підтримує беззнакові цілі числа. Дійсні числа із плаваючою комою можуть бути подані в десятковій або експоненційній формі. Для кожної змінної можна надати власний тип даних. Для цього існує декілька видів:

```
$var1 = true;
$var2 = '1abc';
$var3 = 'abc1';
settype($var1, 'string'); // видасть рядок 1
settype($var2, 'integer'); // видасть число 1
settype($var3, 'bool'); // видасть true
# другий тип надання змінній типу даних
$var1 = (int);
```

```
$var1 = '1abc';  
echo $var1; // видасть число 1
```

MySQL - це популярна система управління базами даних, яка дозволяє отримувати, додавати та обробляти дані. Користувачі MySQL не можуть існувати окремо від бази даних MySQL. Тому слід притримуватися такої послідовності:

- Створити базу даних;
- Додати користувачів MySQL для кожної створеної бази даних.

MySQL має подвійне ліцензування. MySQL може розповсюджуватися відповідно до умов ліцензії GPL. Але за умовами GPL, якщо якась програма використовує бібліотеки MySQL, то вона теж повинна розповсюджуватися за ліцензією GPL. Проте це може розходитися з планами розробників, які не бажають відкривати сирцеві тексти своїх програм. Для таких випадків передбачена комерційна ліцензія компанії Oracle, яка також забезпечує якісну сервісну підтримку. В разі використання та розповсюдження програмного забезпечення з іншими вільними ліцензіями, такими як BSD, Apache, MIT та інші, MySQL дозволяє використання бібліотек MySQL за ліцензією GPL.

1.4. Висновки до розділу 1

Проаналізувавши доступні методи реалізації інформаційної системи, було обрано фреймворк Bootstrap для створення графічного інтерфейсу пацієнта, лікаря і адміністратора.

Для реалізації серверної частини було обрано мову програмування PHP, оскільки важливою перевагою є те, що ця мова належить до інтерпретованих. Це дозволяє обробляти сценарії з достатньо високою швидкістю. За деякими оцінками, більшість PHP-сценаріїв (особливо не дуже великих розмірів) обробляються швидше за аналогічні їм програми, написані на Perl.

Для реалізації бази даних було обрано систему керування реляційними базами даних MySQL, яка є одною з найпоширеніших систем керування.

Використовується для створення динамічних інформаційних систем, має підтримку з боку різноманітних мов програмування.

РОЗДІЛ 2

АРХІТЕКТУРА ТА МОДЕЛЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1. Програмне середовище розробки

Для вирішення задачі Випускної кваліфікаційної роботи було використано наступні програмні засоби:

- система управління реляційними базами даних MySQL
- веб-додаток для адміністрування бази даних phpMyAdmin
- мова програмування PHP
- мова гіпертекстових розмітки HTML
- каскадні таблиці стилів CSS
- мова програмування JavaScript
- графічний редактор Adobe Photoshop

MySQL - це система управління реляційними базами даних. У реляційній

базі даних дані зберігаються в окремих таблицях, завдяки чому досягається вираш у швидкості й гнучкості. Таблиці зв'язуються між собою за допомогою відносин, завдяки чому забезпечується можливість поєднувати при виконанні запиту дані з декількох таблиць. SQL як частина системи MySQL можна охарактеризувати як мова структурованих запитів, що використовується для доступу до баз даних.

MySQL - це ПЗ з відкритим кодом. Застосовувати його і модифікувати може будь-хто. Таке ПЗ можна отримувати за допомогою Internet і використовувати безкоштовно. При цьому кожен користувач може вивчити вихідний код і змінити його у відповідності зі своїми потребами.

phpMyAdmin — веб-додаток з відкритим кодом на мові PHP із графічним веб-інтерфейсом для адміністрування бази даних MySQL або MariaDB. phpMyAdmin дозволяє через браузер здійснювати адміністрування сервера MySQL, запускати запити SQL, переглядати та редагувати вміст таблиць баз даних. Ця програма користується великою популярністю у веб-розробників, оскільки дозволяє керувати базу даних MySQL без вводу SQL команд через дружній інтерфейс і з будь-якого комп'ютера під'єданого до інтернету без необхідності встановлення додаткового програмного забезпечення.

Програма розповсюджується під ліцензією GNU General Public License і тому деякі інші розробники інтегрують його у свої розробки.

PHP – скриптова мова програмування, була створена для генерації HTML сторінок на стороні веб-сервера. PHP є однією з найпоширеніших мов, що використовуються у сфері веб-розробок (разом із Java, .NET, Perl, Python, Ruby). PHP підтримується переважною більшістю хостинг-провайдерів. PHP – проект відкритого програмного забезпечення.

Приклад власної наробки з використанням PHP:

```
<?php  
define('DB_SERVER','localhost');  
define('DB_USER','admin1');  
define('DB_PASS','111111');  
define('DB_NAME','hms');
```

```

$con = mysqli_connect(DB_SERVER,DB_USER,DB_PASS,DB_NAME);
// Check connection
if (mysqli_connect_errno())
{
    echo "Failed to connect to MySQL: " . mysqli_connect_error();
}
?>

```

Вище представлений фрагмент коду виконує підключення до бази даних MYSQL, де в “DB_SERVER” ми вказуємо адресу, в “DB_USER” – логін адміністратора, в “DB_PASS” – пароль, та у “DB_NAME” – назву необхідної бази даних. У разі введення некоректної інформації хоча б в одному пункті – ми отримаємо помилку.

HTML – стандартна мова розмітки веб-сторінок в Інтернеті. Більшість веб-сторінок створюються за допомогою мови HTML (або XHTML). Документ HTML оброблюється браузером та відтворюється на екрані у звичному для людини вигляді.

HTML разом із каскадними таблицями стилів та вбудованими скриптами – це три основні технології побудови веб-сторінок.

Каскадні таблиці стилів CSS – спеціальна мова, що використовується для опису сторінок, написаних мовами розмітки даних. Найчастіше CSS використовують для візуальної презентації сторінок, написаних HTML та XHTML, але формат CSS може застосовуватися до інших видів XML документів.

JavaScript – динамічна, об’єктно-орієнтована мова програмування. Реалізація стандарту ECMAScript. Найчастіше використовується як частина браузера, що надає можливість коду на стороні клієнта (такому, що виконується на пристрої кінцевого користувача) взаємодіяти з користувачем, керувати браузером, асинхронно обмінюватися даними з сервером, змінювати структуру та зовнішній вигляд веб-сторінки. Мова JavaScript також використовується для програмування на стороні серверу (подібно до таких мов програмування, як Java і C#), розробки ігор, стаціонарних та мобільних додатків, сценаріїв в

прикладному ПЗ, всередині PDF документів тощо.

Adobe Photoshop - графічний редактор, розроблений і поширюваний фірмою Adobe Systems. Цей продукт є лідером ринку в області комерційних засобів редагування растрових зображень, і найвідомішим продуктом фірми Adobe. Часто цю програму називають просто Photoshop (Фотошоп). У наш час Photoshop доступний на платформах Mac OS X/Mac OS і Microsoft Windows. Ранні версії редактора були портовані під SGI IRIX, але офіційна підтримка була припинена, починаючи з третьої версії продукту. Для версії CS і CS6 можливий запуск під Linux за допомогою альтернативи Windows API — Wine.

2.2 Організаційна структура Інформаційної системи

Інформаційна система консультативно-діагностичного центру включає в себе систему, яка складається з трьох видів облікових записів:

- Пацієнт
- Лікар
- Адміністратор

Панель керування пацієнта надає можливість записатися на візит, переглянути історію візитів, додати або редагувати дані в своєму профілі, та переглянути свою медичну історію. Кожен пацієнт може створити обліковий запис самостійно.

Лікар у своєму обліковому записі має можливість переглянути свої зустрічі, оновити дані свого профіля, додати нового пацієнта да його вхідні дані, керувати вже створеними картками пацієнтів (редагувати вже існуючі дані, та додавати нові записи обстежень), та виконувати пошук інформації по імені або контактному номеру. Можливості самостійно зареєструвати обліковий запис немає, таку привілегію має лише адміністратор інформаційної системи.

Адміністратор має наступні можливості: керувати зареєстрованими користувачами та пацієнтами, яких внесли до бази лікарі, керувати лікарями, переглядати існуючі зустрічі та переглядати повідомлення, які були відправлені через форму зворотного зв'язку. У панелі керування є можливість переглянути

спеціальності лікарів, додати нову, зареєструвати в систему нового лікаря, переглянути логи лікарів та користувачів, переглянути звіти та виконати пошук пацієнта по імені або контактному номеру.

2.3. Система керування інформаційною системою

Система створена таким чином, що її можна встановили на будь якому веб-сервері, що підтримує PHP і MySQL.

PHP (Hypertext Preprocessor (Препроцесор гіпертексту)) - це широко використовувана мова сценаріїв загального призначення з відкритим вихідним кодом. Синтаксис мови бере початок з C, Java і Perl. PHP досить простий для вивчення. Перевагою PHP є надання web-розробникам можливості швидкого створення динамічних web-сторінок.

Значною відзнакою PHP від якого-небудь коду, що виконується на стороні клієнта, наприклад, JavaScript, є те, що PHP-скрипти виконуються на стороні сервера. Ви навіть можете конфігурувати свій сервер таким чином, щоб HTML-файли оброблялися процесором PHP, так що клієнти навіть не зможуть дізнатися, чи отримують вони звичайний HTML-файл або результат виконання скрипта.

Безпека мови PHP:

- Засоби безпеки системного рівня. PHP можна налаштувати так, щоб вона забезпечувала максимальну свободу дій і безперу. PHP може працювати в безпечному режимі (safe mode), який обмежує можливості застосування PHP користувачами. Наприклад: максимальний час виконання та використання пам'яті.
- Засоби безпеки рівня програми. PHP включає надійні механізми шифрування. PHP також сумісний з багатьма додатками інших розробників, що дозволяє легко інтегрувати його з захищеними технологіями електронної комерції. Вихідний код PHP не можна переглянути в браузері, оскільки він виконується на сервері.

PHP використовується не лише у поєднанні з HTML, але й із Javascript? WML, XML та іншими мовами програмування. PHP-код може передаватися будь-якими браузерами і пристроями, в тому числі стільниковими телефонами, портативними комп'ютерами. PHP-код можна використовувати в режимі командного рядка.

2.4. FTP-менеджер FileZilla

На сьогоднішній день FileZilla є найкращим безкоштовним FTP-менеджером. Це програма, яка дозволяє одержувати доступ до будь-якого FTP, SFTP сервера з будь-якого комп'ютера. Програма дуже зручна, вона розроблена як для початківців, так і для просунутих користувачів. FileZilla працює на різних операційних системах, має портативну версію, яку можна встановити на флеш-накопичувач. Останню ліцензійну версію можна завантажити на нашому сайті, де є можливість завантажити FileZilla безкоштовно і без обмежень.

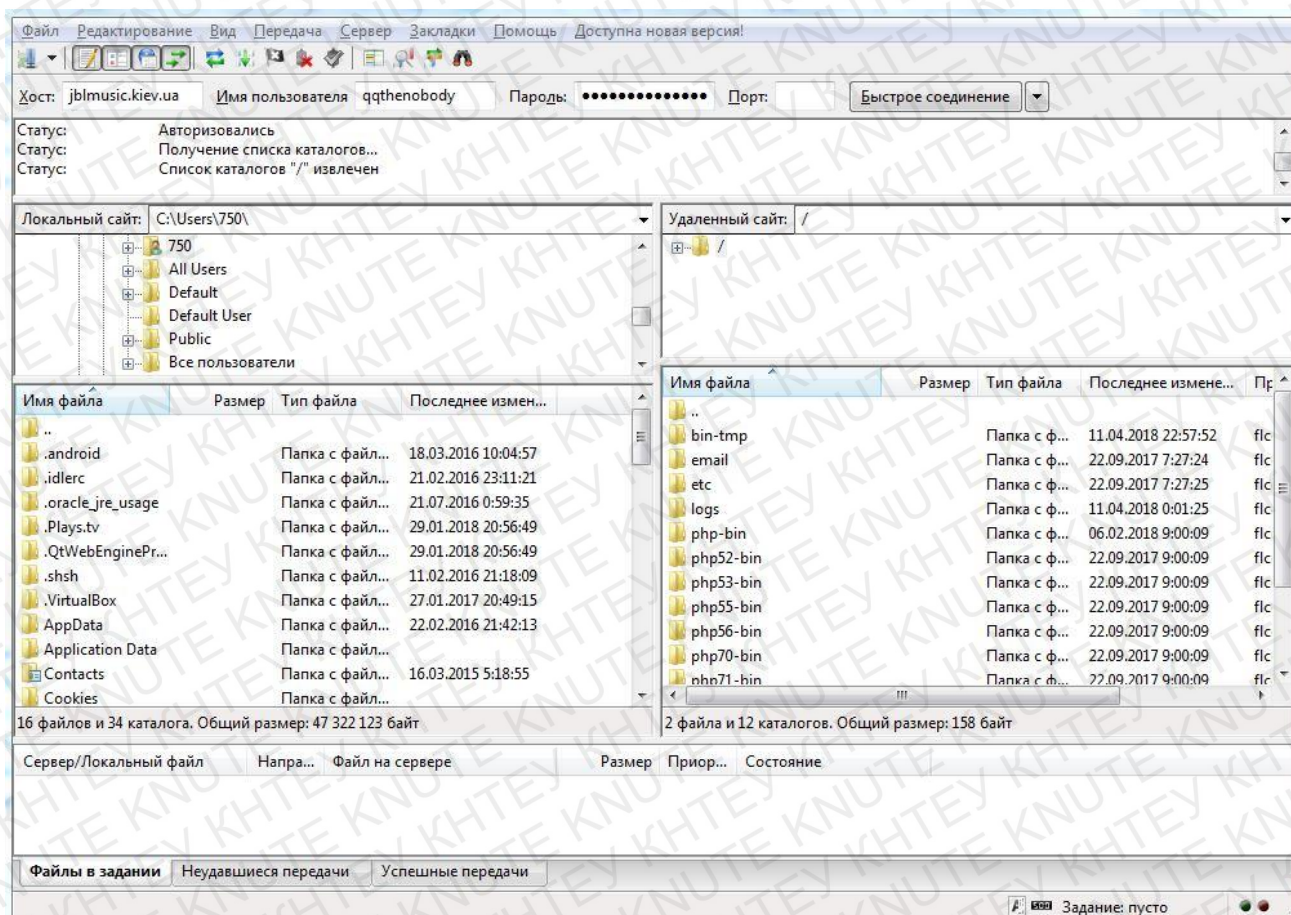


Рис. 2.1. FTP-менеджер FileZilla

За допомогою програми ви зможете відправляти файли сайту на сервер, змінювати їх назву, переміщати, видаляти. У додатку можна зберігати паролі, імена користувачів, типи з'єднання. Інтерфейс FileZilla дуже простий, він схожий на Провідник Windows з деревоподібною системою управління. За допомогою програми можна завантажувати і завантажувати велику кількість файлів одночасно і на кілька серверів в тому числі. Функція CHMOD дозволяє користувачеві змінювати права доступу до файлів. Програма може працювати з фаєрволом, через проксі-сервер, вона швидко підключається до серверів, може змінювати швидкість передачі даних. Інтерфейс FileZilla має підтримку декількох мов.

Програма володіє великою кількістю функцій. Для того, щоб розібратися з ними, в FileZilla є файл допомоги, який буде зручним путівником по програмі. Безкоштовний FTP-менеджер FileZilla не поступається за своїм функціоналом платним аналогам, що робить вельми привлекательним, а головне ефективним в повсякденному використанні додатку.

Особливості та функції програми:

- додаток легко освоюється,
- передача файлів по протоколах FTP, SFTP, FTPS,
- версії програми для всіх OS,
- багатомовний інтерфейс,
- можливість дозавантаження, відновлення закачування файлів після розриву з'єднання.
- передача великих файлів об'ємом до 4 Гб,
- підтримка drag & drop,
- керовані закладки і черги передачі файлів,
- можливість встановлення максимальної швидкості скачування,
- додаток може працювати через проксі-сервер

2.5. Редактор коду Sublime Text

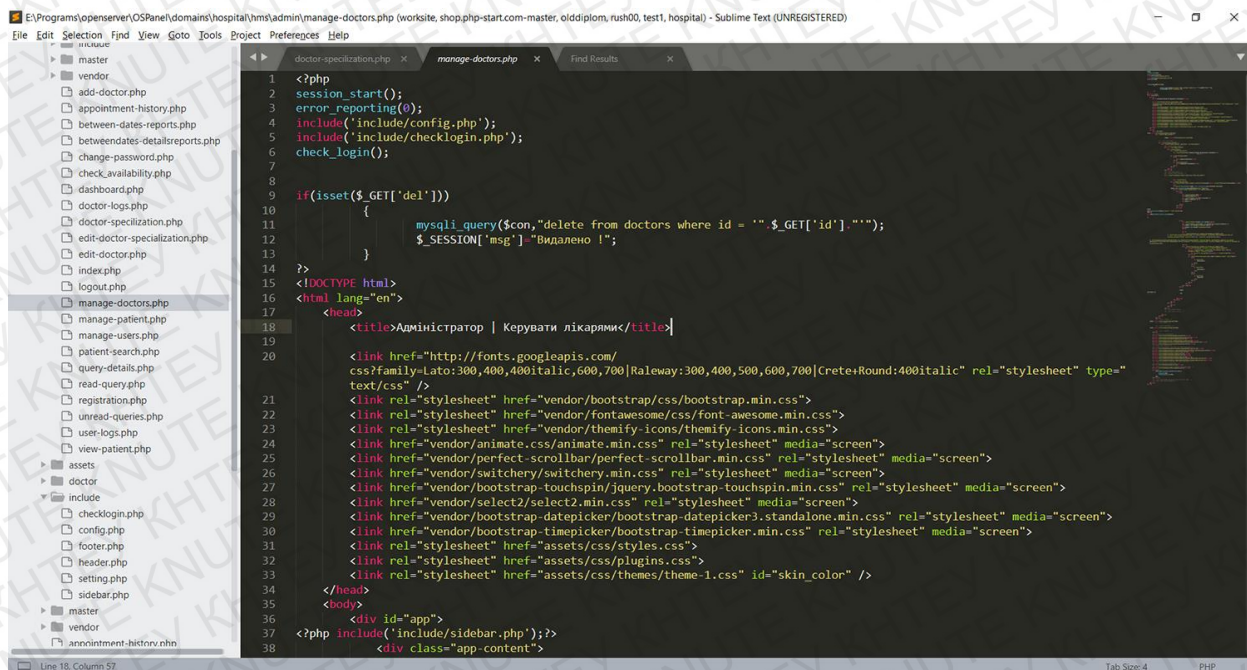


Рис. 2.2. Редактор коду Sublime Text

Редактор містить різні візуальні теми, з можливістю завантаження додаткових. Користувачі бачать весь свій код в правій частині екрану у вигляді міні-карти, при кліці на яку можна здійснювати навігацію. Є кілька режимів екрану. Один з них включає від 1 до 4 панелей, за допомогою яких можна показувати до чотирьох файлів одночасно. Повноцінний (free modes) режим показує тільки один файл без будь-яких додаткових навколо нього меню.

Виділення стовпців цілком або розстановка кілька показників по тексту, що робить можливим миттєву правку. Показники поводяться, ніби кожен з них - єдина в тексті. Команди типу: переміщення на знак, переміщення на рядок, вибірка тексту, переміщення на слово або його частини (CamelCase, розділений дефісом або підкресленням), перехід на початок або кінець рядка тощо, Впливає на всі показники незалежно і відразу, дозволяючи правити складноструктурований текст швидко, без використання макрокоманд або регулярних виразів.

Коли користувач набирає код, Sublime Text, в залежності від використовуваної мови, буде пропонувати різні варіанти для завершення запису. Редактор також автоматично завершує створені користувачем змінні.

Основні переваги Sublime Text:

- Додатково реалізована функція автозбереження, яка допомагає користувачам не втратити виконану роботу.
- Настроюються комбінації клавіш і інструмент навігації дозволяють призначати свої комбінації клавіш для меню і панелей інструментів (тільки для першої версії, у другій і третій - Command Palette).
- Можливість пошуку під час набору використовується для пошуку в документі.
- Функція перевірки синтаксису працює подібним же чином, перевіряючи коректність прямо під час введення.
- Є можливість автоматизації за допомогою макросів і повтору останніх дій.
- Команди редагування, включаючи редагування відступів, переформатування параграфів і об'єднання рядків.

2.6. Локальний Веб-сервер Open Server

Open Server являє собою так званий WAMP комплекс, суть якого можна зрозуміти з розшифровки цієї аббревіатури:

1. Windows - операційна система, для роботи в якій призначений даний локальний сервер.
2. apache - web-сервер, який «піднімається» при запуску програми Open Server.
3. MySQL - дуже популярна система управління базами даних, яка є обов'язковою умовою для роботи багатьох двигунів сайтів, в тому числі таких популярний, як Joomla та WordPress.
4. PHP - інтерпретатор серверного мови програмування, на якому написано більшість CMS і створено безліч інших веб-додатків.

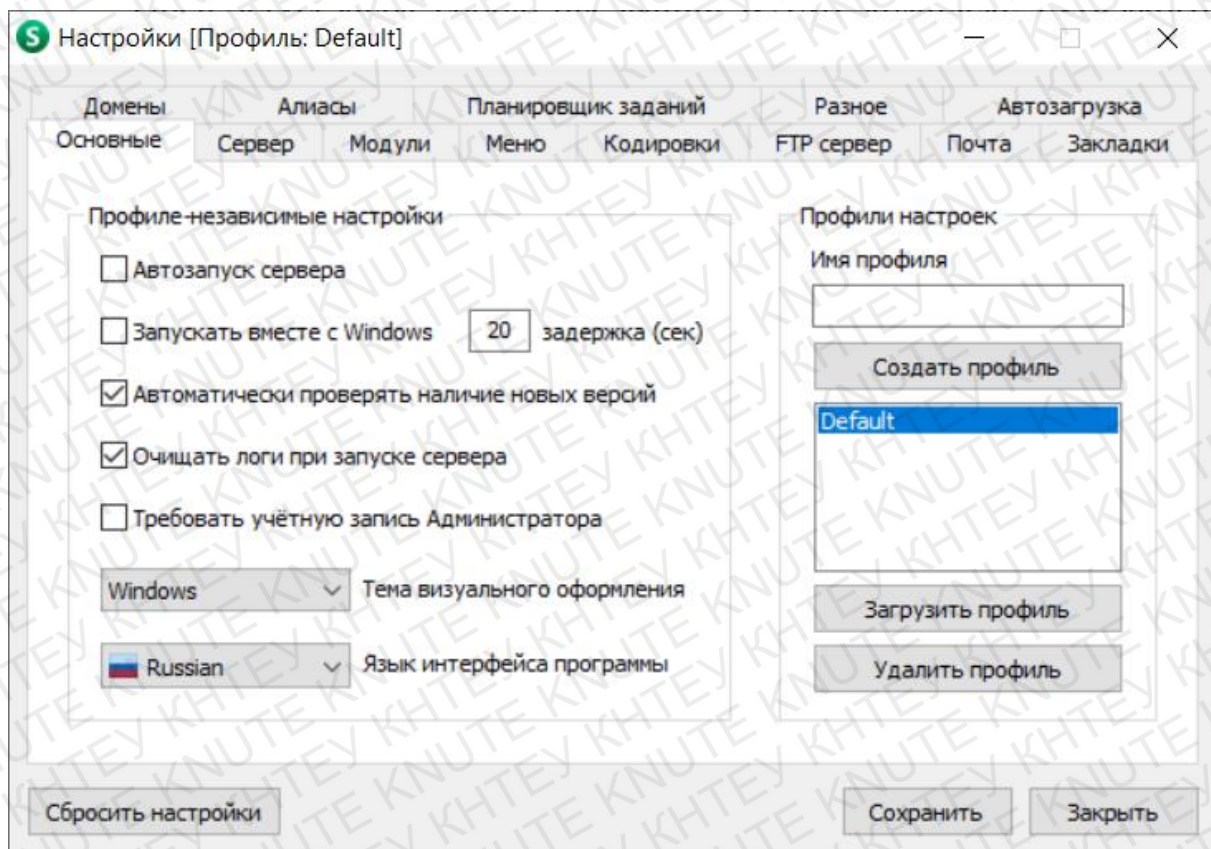


Рис. 2.3. Меню налаштувань Open Server

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 веб-проектів, а так само для надання веб-сервісів в локальних мережах.

2.7. Висновки до розділу 2

У ролі локального сервера було обрано систему Open Server, яка працює з базами даних MySQL, має багато корисних модулів, які полегшують розробку та тестування розроблених функцій.

Для написання коду було використано редактор Sublime Text, за його зручність у використанні та привабливий дизайн.

РОЗДІЛ 3

СТВОРЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ КОНСУЛЬТАТИВНО- ДІАГНОСТИЧНОГО ЦЕНТРУ

3.1. Створення та підключення бази даних

Генеруємо базу даних в MySQL (Tools/Forward Engineer/Schema Generation), перед цим створивши порожню базу даних. Генерація структури БД на основі створеного SQL коду відбувається після натиснення кнопки Generate. Діалог зв'язку з БД і виконання SQL коду відбувається результаті натиснення кнопки Connect. Отримуємо згенеровану базу даних в середовищі MySQL.

Лістинг коду приєднання системи до бази даних:

```
<?php
define('DB_SERVER','localhost');
define('DB_USER','admin1');
define('DB_PASS','111111');
define('DB_NAME','hms');
$con = mysqli_connect(DB_SERVER,DB_USER,DB_PASS,DB_NAME);
// Check connection
if (mysqli_connect_errno())
{
    echo "Failed to connect to MySQL: " . mysqli_connect_error();
}
?>
```

У поле «DB_SERVER» передається адресу сервера, потім відправляється ім'я користувача у поле «DB_USER» і «DB_PASS», і у поле «DB_NAME» вказується назва бази даних, у разі невдалого підключення, система дасть про це повідомлення.

3.2. Проектування бази даних

База даних – впорядкований набір логічно взаємопов'язаних даних, що використовуються спільно та призначені для задоволення інформаційних потреб користувачів. Головне завдання БД – гарантоване збереження значних обсягів інформації (так звані записи даних) та надання доступу до неї користувачеві або

ж прикладній програмі. Таким чином, БД складається з двох частин: збереженої інформації та системи керування нею. З метою забезпечення ефективності доступу записи даних організовують як множину фактів (елемент даних).

Проектування баз даних – процес створення схеми бази даних і визначення необхідних обмежень цілісності. Основні завдання:

- забезпечення зберігання в БД всієї необхідної інформації;
- забезпечення можливості отримання даних по всім необхідним запитам;
- скорочення надмірності і дублювання даних;
- забезпечення цілісності бази даних.

Логічне проектування – створення схеми бази даних на основі конкретної моделі даних, наприклад, реляційної моделі даних. Для реляційної моделі даних логічна модель – набір схем відносин, зазвичай із зазначенням первинних ключів, а також «зв'язків» між відносинами, що представляють собою зовнішні ключі.

Фізичне проектування – створення схеми бази даних для конкретної СУБД. Специфіка конкретної СУБД може включати в себе обмеження на іменування об'єктів бази даних, обмеження на підтримувані типи даних та інше. Крім того, специфіка конкретної СУБД при фізичному проектуванні включає вибір рішень, пов'язаних з фізичним середовищем зберігання даних (вибір методів управління дисковою пам'яттю, поділ БД по файлах і пристроях, методів доступу до даних), створення індексів і т. д.

База даних системи підтримки діяльності відділу продажу була розроблена за допомогою системи phpMyAdmin версії 4.9.0.1, використовуючи сервер баз даних MySQL версії 8.0.15.

Опис таблиць бази даних наведено у табл. 3.1.

Таблиця 3.1. Опис таблиць бази даних

№ таблиці	Назва	Призначення
1	Admin	Таблиця для зберігання інформації про адміністратора.

2	Appointment	Таблиця для зберігання інформації про записи відвідування пацієнтів.
3	Doctors	Таблиця для зберігання інформації про лікарів.
4	Doctorslog	Таблиця для зберігання інформації про вхід та вихід на сайт лікарів.
5	Doctorspecilization	Таблиця для зберігання інформації про спеціалізацію лікарів.
6	Tblcontactus	Таблиця для зберігання заявок на зворотній зв'язок.
7	Tblmedicalhistory	Таблиця для зберігання історії обстеження пацієнтів.
8	Tblpatient	Таблиця для зберігання діагнозів пацієнтів.
9	Userlog	Таблиця для зберігання інформації про вхід та вихід на сайт користувачів.
10	Users	Таблиця для зберігання інформації про користувачів.

Опис таблиці «Admin» наведено у табл. 3.2.

Таблиця 3.2. Структура таблиці «Admin»

№	Назва поля	Тип даних	Призначення
1	id	INT(11)	Первинний ключ
2	username	VARCHAR(255)	Логін користувача
3	password	VARCHAR(255)	Пароль користувача
4	updateDate	VARCHAR(255)	Інформація про дату зміни даних

Опис таблиці «appointment» наведено у табл. 3.3.

Таблиця 3.3. Структура таблиці «appointment»

№	Назва поля	Тип даних	Призначення
1	id	INT(11)	Первинний ключ
2	doctorSpecialization	VARCHAR(255)	Спеціалізація лікаря
3	doctorId	INT(11)	Унікальний номер у списку лікаря
4	userId	INT(11)	Унікальний номер у списку користувача
5	consultancyFees	INT(11)	Вартість послуг
6	appointmentDate	VARCHAR(255)	Дата прийому
7	appointmentTime	VARCHAR(255)	Час прийому
8	postingDate	TIMESTAMP	Дата заявки
9	userStatus	INT(11)	Статус користувача
10	doctorStatus	INT(11)	Статус лікаря
11	updateDate	TIMESTAMP	Дата зміни даних

Опис таблиці «doctors» наведений у табл. 3.4.

Таблиця 3.4. Структура таблиці «doctors»

№	Назва поля	Тип даних	Призначення
1	id	INT(11)	Первинний ключ
2	specilization	VARCHAR(255)	Спеціалізація
3	doctorName	VARCHAR(255)	Ім'я лікаря
4	address	LONGTEXT	Адреса
5	docFees	VARCHAR(255)	Вартість послуг
6	Contactno	BIGINT(11)	Контактний номер
7	docEmail	VARCHAR(255)	Електронна пошта
8	Password	VARCHAR(255)	Пароль
9	creationDate	TIMESTAMP	Дата створення облікового запису
10	updateDate	TIMESTAMP	Дата останніх змін облікового запису

Опис таблиці «doctorslog» наведено у табл. 3.5.

Таблиця 3.5. Структура таблиці «doctorslog»

№	Назва поля	Тип даних	Призначення
1	id	INT(11)	Первинний ключ
2	Uid	INT(11)	Ключ
3	Username	VARCHAR(255)	Ім'я користувача
4	Userip	BINARY(16)	IP-адреса користувача

5	loginTime	TIMESTAMP	Дата авторизації
6	Logout	VARCHAR(255)	Дата виходу з облікового запису
7	status	INT(11)	Статус

Опис таблиці «doctorspecilization» наведено у табл. 3.6.

Таблиця 3.6. Структура таблиці «doctorspecilization»

№	Назва поля	Тип даних	Призначення
1	id	INT(11)	Первинний ключ
2	Specilization	VARCHAR(255)	Спеціалізація
3	creationDate	TIMESTAMP	Дата створення
4	updatetime	TIMESTAMP	Дата внесення змін

Опис таблиці «tblcontactus» наведено у табл. 3.7.

Таблиця 3.7. Структура таблиці «tblcontactus»

№	Назва поля	Тип даних	Призначення
1	id	INTEGER	Первинний ключ
2	fullname	VARCHAR(255)	Повне ім'я
3	email	DATE	Електронна пошта
4	Contactno		Контактний номер
5	Message		Повідомлення

6	PostingDate		Дата публікації
7	AdminRemark		Внесення змін адміністратором
8	LastupdateDate		Дата останніх змін
9	isRead		Чи прочитано повідомлення

Опис таблиці «tblmedicalhistory» наведено у табл. 3.8.

Таблиця 3.8. Структура таблиці «tblmedicalhistory»

№	Назва поля	Тип даних	Призначення
1	id	INT(10)	Первинний ключ
2	PatientID	INT(10)	Ключ пацієнта
3	BloodPressure	VARCHAR(200)	Кров'яний тиск
4	BloodSugar	VARCHAR(200)	Цукор у крові
5	Weight	VARCHAR(100)	Вага
6	Temperature	VARCHAR(200)	Температура
7	MedicalPres	MEDIUMTEXT	Діагнози
8	CreationDate	TIMESTAMP	Дата створення запиту

Опис таблиці «tblpatient» наведено у табл. 3.9.

Таблиця 3.9. Структура таблиці «tblpatient»

№	Назва поля	Тип даних	Призначення
---	------------	-----------	-------------

1	id	INT(10)	Первинний ключ
2	Docid	INT(10)	Ключ лікаря
3	PatientName	VARCHAR(200)	Ім'я пацієнта
4	PatientContno	BIGINT(10)	Контактний номер пацієнта
5	PatientEmail	VARCHAR(200)	Електронна адреса пацієнта
6	PatientGender	VARCHAR(50)	Стать пацієнта
7	PatientAdd	MEDIUMTEXT	Адреса пацієнта
8	PatientAge	INT(10)	Вік пацієнта
9	PatientMedhis	MEDIUMTEXT	Медична історія пацієнта
10	CreationDate	TIMESTAMP	Дата створення запису
11	UpdationDate	TIMESTAMP	Дата внесення змін у запис

Опис таблиці «userlog» наведено у табл. 3.10.

Таблиця 3.10. Структура таблиці «userlog»

№ поля	Назва	Тип даних	Призначення
1	id	INT(11)	Первинний ключ
2	uid	INT(11)	Ключ сесії
3	username	VARCHAR(255)	Ім'я користувача
4	userip	BINARY(16)	IP-адреса користувача

5	loginTime	TIMESTAMP	Дата авторизації
6	Logout	VARCHAR(255)	Дата виходу з облікового запису
7	status	INT(11)	Статус

Опис таблиці «users» наведено у табл. 3.11.

Таблиця 3.11. Структура таблиці «Області»

№	Назва поля	Тип даних	Призначення
1	id	INT(11)	Первинний ключ
2	fullName	VARCHAR(255)	Повне ім'я користувача
3	address	LONGTEXT	Адреса
4	city	VARCHAR(255)	Місто
5	Gender	VARCHAR(255)	Пол
6	Email	VARCHAR(255)	Електронна адреса
7	Password	VARCHAR(255)	Пароль
8	regDate	TIMESTAMP	Дата реєстрації
9	updatetime	TIMESTAMP	Дата внесення змін

3.3. Інтерфейс адміністратора

Для роботи з системою в інтерфейсі адміністратора, потрібно відкрити веб-переглядач (браузер), та ввести адресу адміністраторської панелі веб-сайту.

Після цього необхідно авторизуватися у системи. Сторінку авторизації наведено у рис 3.1.

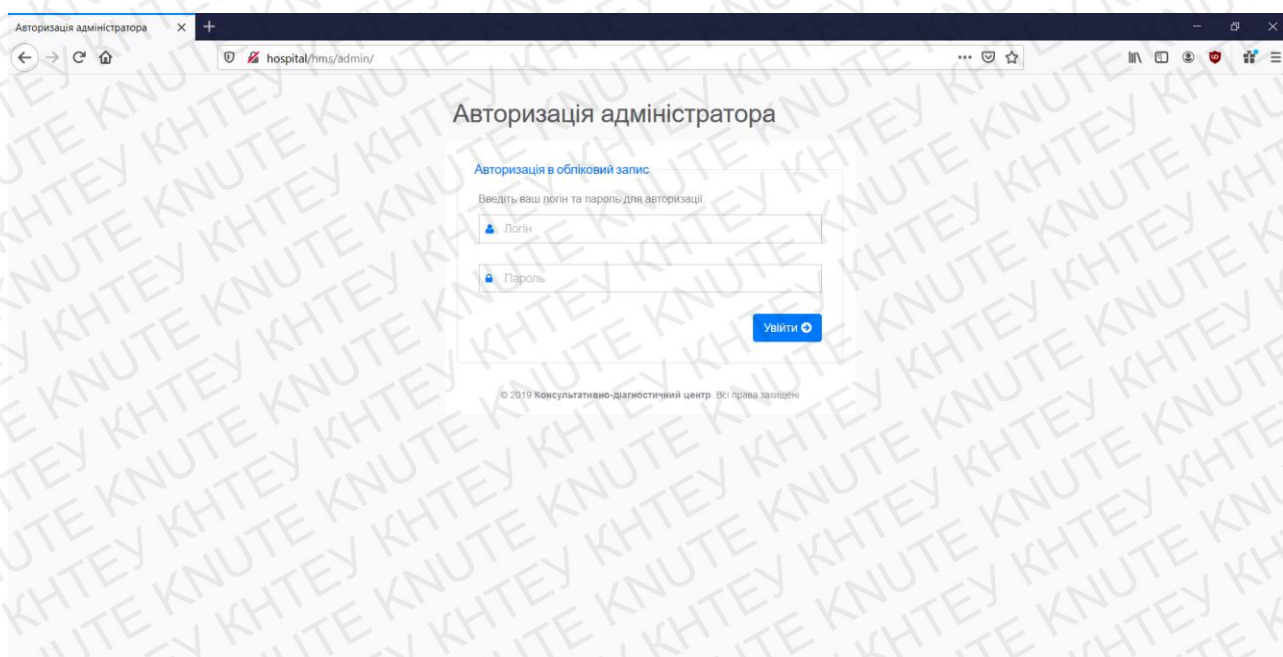


Рис. 3.1. Сторінка авторизації

Після авторизації йде направлення на панель керування. На цій сторінці можна переглянути інформацію про лікарів, користувачів системи, пацієнтів, повідомлення. Головну сторінку панелі керування наведено на рис. 3.2.

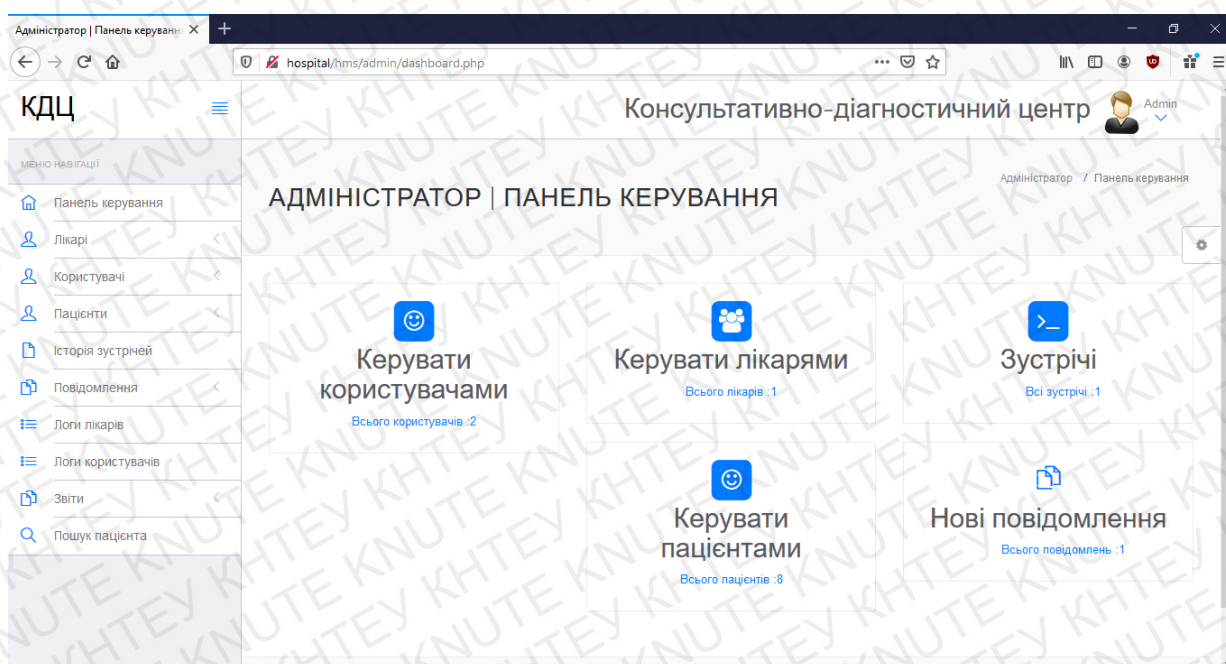


Рис. 3.2. Панель керування адміністратора

Для того щоб додати спеціалізацію лікаря, потрібно натиснути «лікарі» на панелі зліва, далі обрати «Спеціалізація лікаря», ввести потрібну спеціалізацію у відповідне поле, яке зображено на рис. 3.3., та натиснути кнопку «Підтвердити».

Рис. 3.3. Керування спеціалізаціями лікарів

Після натискання кнопки підтвердити, спрацьовує код

```
if(isset($_POST['submit']))
{
    $sql=mysqli_query($con,"insert into doctorSpecilization(specilization)
values('".$_POST['doctorspecilization']."");
    $_SESSION['msg']="Спеціалізацію успішно додано !";
}
```

Цей код перевіряє, якщо в метод POST було відправлено значення “submit”, то він створює SQL запит, який виконує підключення до нашої бази даних, та передає введену нами спеціалізацію в «doctorspecilization».

У меню «лікарі» є можливість додати нового лікаря, налаштувати спеціалізацію лікаря, і керувати вже існуючими обліковими записами лікарів.

Рис. 3.4. Сторінка «Додати лікаря»

Після заповнення необхідних полів і натискання кнопки «підтвердити» спрацьовує умова:

```

if(isset($_POST['submit']))
{
    $docspecialization=$_POST['Doctorspecialization'];
    $docname=$_POST['docname'];
    $docaddress=$_POST['clinicaddress'];
    $docfees=$_POST['docfees'];
    $doccontactno=$_POST['doccontact'];
    $docemail=$_POST['docemail'];
    $password=md5($_POST['npass']);
    $sql=mysqli_query($con,"insert into
doctors(specilization,doctorName,address,docFees,contactno,docEmail,password)
values('$docspecialization','$docname','$docaddress','$docfees','$doccontactno','$doce
mail','$password')");
    if($sql)
    {
        echo "<script>alert('Інформація по лікарю успішно додана');</script>";
    }
}

```

```

echo "<script>>window.location.href ='manage-doctors.php'</script>";

}

}

?>

```

Система зчитує введені дані, і записує через метод POST у відповідні змінні, далі у змінну “sql” записується результат спрацювання `mysqli_query`, в яку передається інформація про підключення бази даних і потрібні поля для вводу, далі передаються введені значення. У разі успішного спрацювання, система відправляє повідомлення на екран про запис інформації.

Керування обліковими записами лікарів реалізовано на сторінці `/Manage-doctors.php`

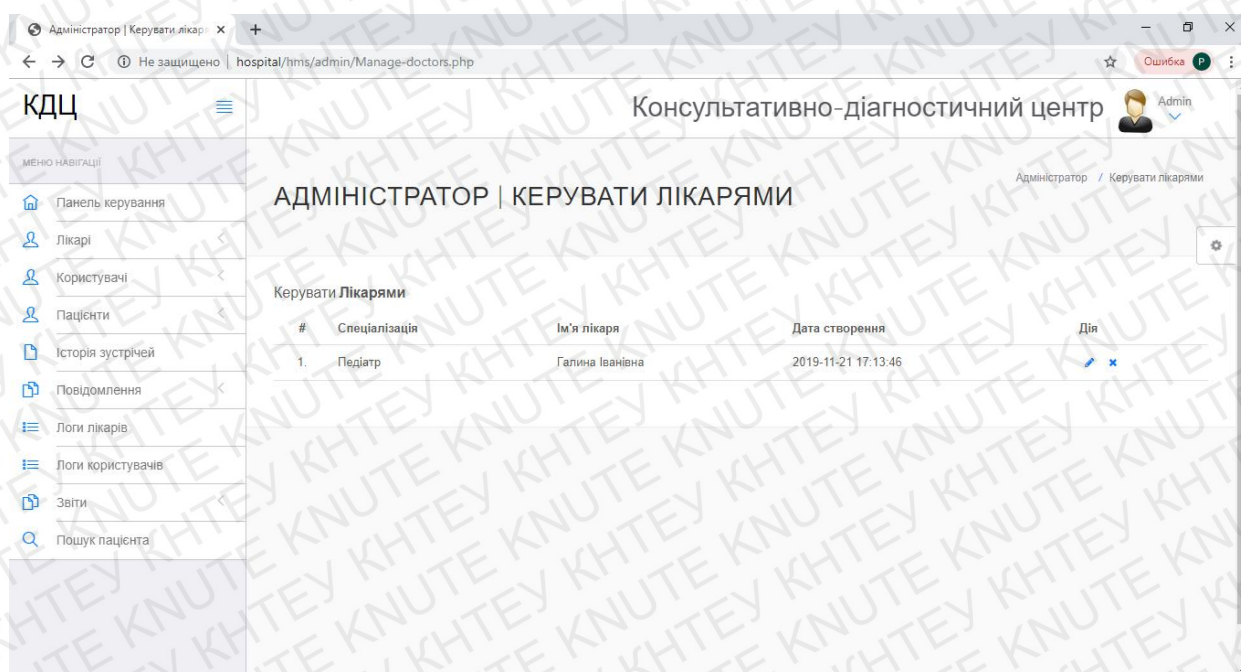


Рис. 3.5. Сторінка «Керувати лікарями»

Є можливість видалити, та редагувати запис. Для видалення потрібно натиснути на хрестик у колонці «дія», після чого спрацьовує код:

```

<a href="manage-doctors.php?id=?php echo $row['id']?>&del=delete"
onClick="return confirm('Ви впевнені що хочете видалити?')"class="btn btn-
transparent btn-xs tooltips" tooltip-placement="top" tooltip="Remove"><i class="fa
fa-times fa fa-white"></i></a>

```

Щоб захиститися від помилкового спрацювання, було додано уточнююче вікно, після підтвердження якого запис видаляється.

Для переходу до сторінки редагування облікового запису лікаря необхідно натиснути на «олівець» у колонці під назвою «дія». Спрацьовує даний код:

```
<div class="visible-md visible-lg hidden-sm hidden-xs">  
    <a href="edit-doctor.php?id=<?php echo $row['id'];?>" class="btn btn-transparent btn-xs" tooltip-placement="top" tooltip="Edit"><i class="fa fa-pencil"></i></a>
```

Який перенаправляє на сторінку /edit-doctor.php. Сторінку редагування даних лікаря наведено на рис 3.6.

Рис. 3.6. Сторінка редагування даних лікаря

Для збереження внесених змін потрібно натиснути кнопку «підтвердити».

3.4. Інтерфейс пацієнта

Для авторизації в системі під обліковим записом «пацієнта» потрібно на головній сторінці обрати «пацієнти» і ввести логін і пароль. Вікно авторизації наведено на рис 3.7.

Рис. 3.7. Сторінка авторизації пацієнта

При необхідності є можливість зареєструвати новий обліковий запис. Для цього потрібно натиснути «Створити обліковий запис». Вікно реєстрації наведено на рис 3.8.

Рис. 3.8. Реєстрація пацієнта

Після введення електронної адреси користувача спрацьовує функція, перевіряє в існуючій базі даних на повтор.

```
<script>
```

```
function userAvailability() {
```

```
$("#loaderIcon").show();
```

```
jQuery.ajax({
```

```

url: "check_availability.php",
data:'email='+$("#email").val(),
type: "POST",
success:function(data){
$("#user-availability-status1").html(data);
$("#loaderIcon").hide();
},
error:function (){}
});
}
</script>

```

Якщо електронна адреса існує в базі, користувач отримає повідомлення, що адреса зайнята.

При введення паролів функція «valid» перевіряє чи співпадають введені паролі.

```

<script type="text/javascript">
function valid()
{
if(document.registration.password.value!=
document.registration.password_again.value)
{
alert("Паролі не співпадають !!");
document.registration.password_again.focus();
return false;
}
return true;
}
</script>

```

Якщо паролі різні, система відправить помилку. Приклад наведено на рис 3.9.

ustration.php

Підтвердіть дійствие на странице hospital

Паролі не співпадають !!

OK

Стать

☐ Жіноча ☒ Чоловіча

Введіть дані вашого облікового запису:

Email доступний для реєстрації .

☒ Я погоджуюсь

Рис. 3.9. Підтвердження пароля

Після успішної авторизації відкривається доступ до панелі керування пацієнта. Сторінку «Користувач» наведено на рис. 3.10.

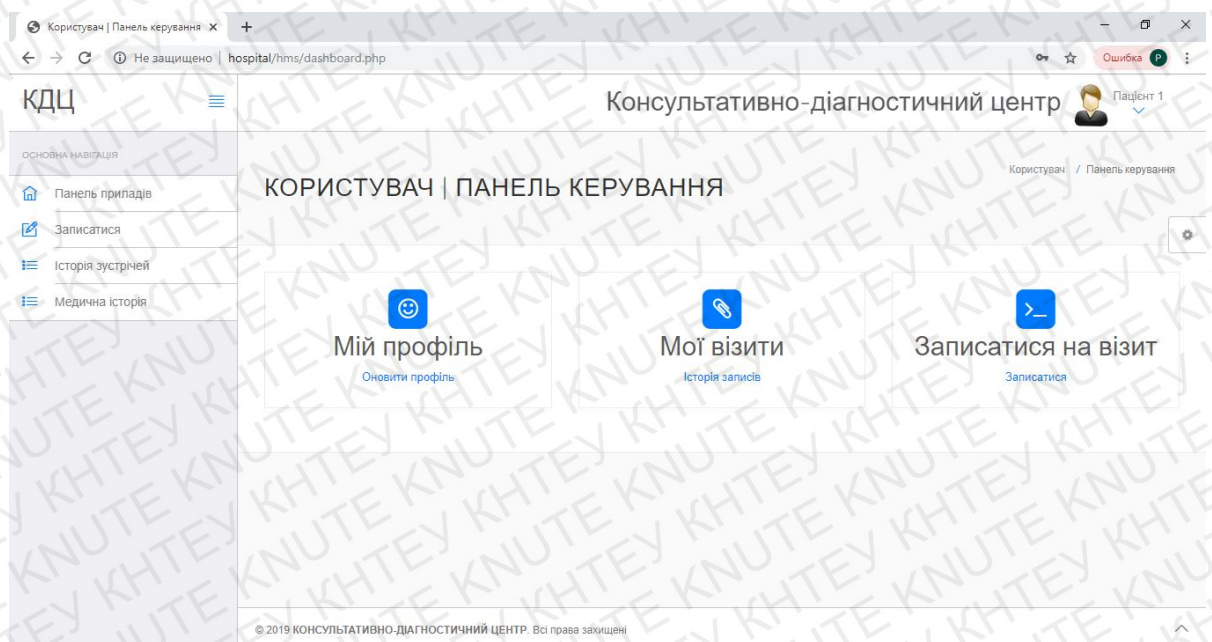


Рис. 3.10. Панель керування користувача

У панелі керування пацієнта є можливість записатися до лікаря. Для цього потрібно натиснути «Записатися» у вікні «Записатися на візит». Далі

користувача перенаправляє на сторінку /book-appointment.php, яку зображено на рис. 3.11.

The screenshot shows a web browser window with the address bar displaying 'hospital/hms/book-appointment.php'. The page title is 'КОРИСТУВАЧ | ЗАПИСАТИСЯ'. On the left, there is a sidebar menu with four items: 'Панель приладів', 'Записатися', 'Історія зустрічей', and 'Медична історія'. The main content area is titled 'Записатися' and contains a form with the following fields: 'Спеціалізація Лікаря' (with a dropdown menu showing 'Обрати спеціалізацію'), 'Лікар' (with a dropdown menu showing 'Обрати Лікаря'), 'Вартість' (a text input field), 'Дата' (a date picker), and 'Час' (a time picker showing '2:00 AM' with an example 'приклад : 10:00 PM'). At the bottom of the form is a blue button labeled 'Підтвердити'.

Рис. 3.11. Сторінка запису на візит

```
<script>
function getdoctor(val) {
    $.ajax({
        type: "POST",
        url: "get_doctor.php",
        data:'specilizationid='+val,
        success: function(data){
            $("#doctor").html(data);
        }
    });
}
</script>
```

Даний скрипт виводить з бази даних список всіх доступних лікарів по обраній спеціалізації. В залежності від обраного лікаря система дістає з бази даних вартість послуг за допомогою функції «getfee»

```
<script>
```

```

function getfee(val) {
    $.ajax({
        type: "POST",
        url: "get_doctor.php",
        data:'doctor='+val,
        success: function(data){
            $("#fees").html(data);
        }
    });
}
</script>

```

3.5. Інтерфейс лікаря

Для авторизації лікаря на головній сторінці у вікні «Лікар» натиснути «Натисніть тут». Вікно авторизації наведено на рис 3.12.

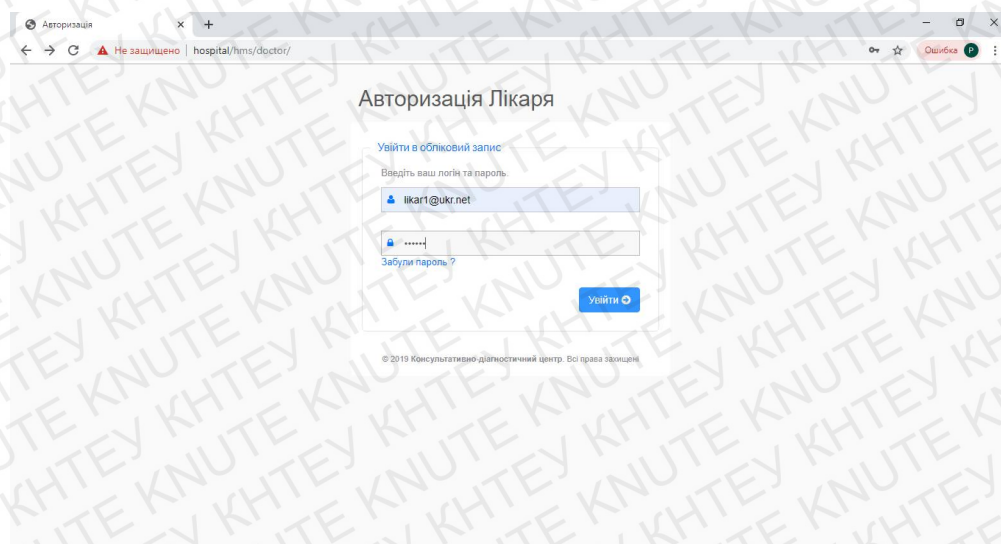


Рис. 3.12. Реєстрація пацієнта

У лікарів відсутня можливість самостійно реєструвати обліковий запис. Дана функція існує лише у панелі керування адміністратора.

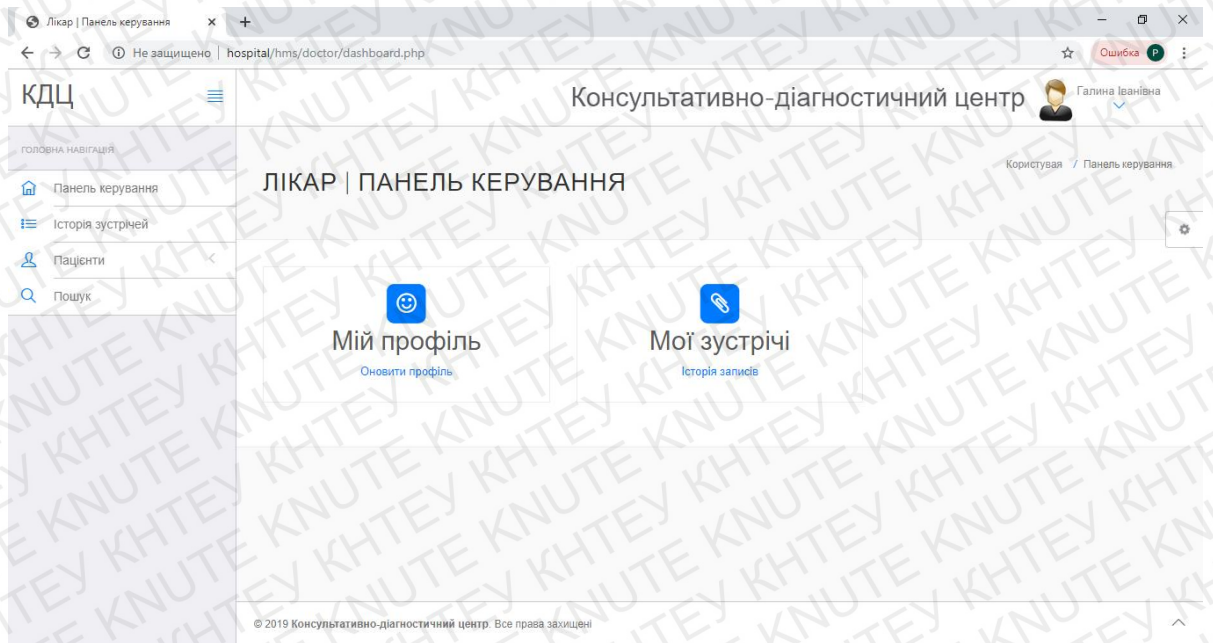


Рис. 3.13. Панель керування лікаря

У панелі керування лікаря є можливість редагувати профіль, та керувати зустрічами з пацієнтами.

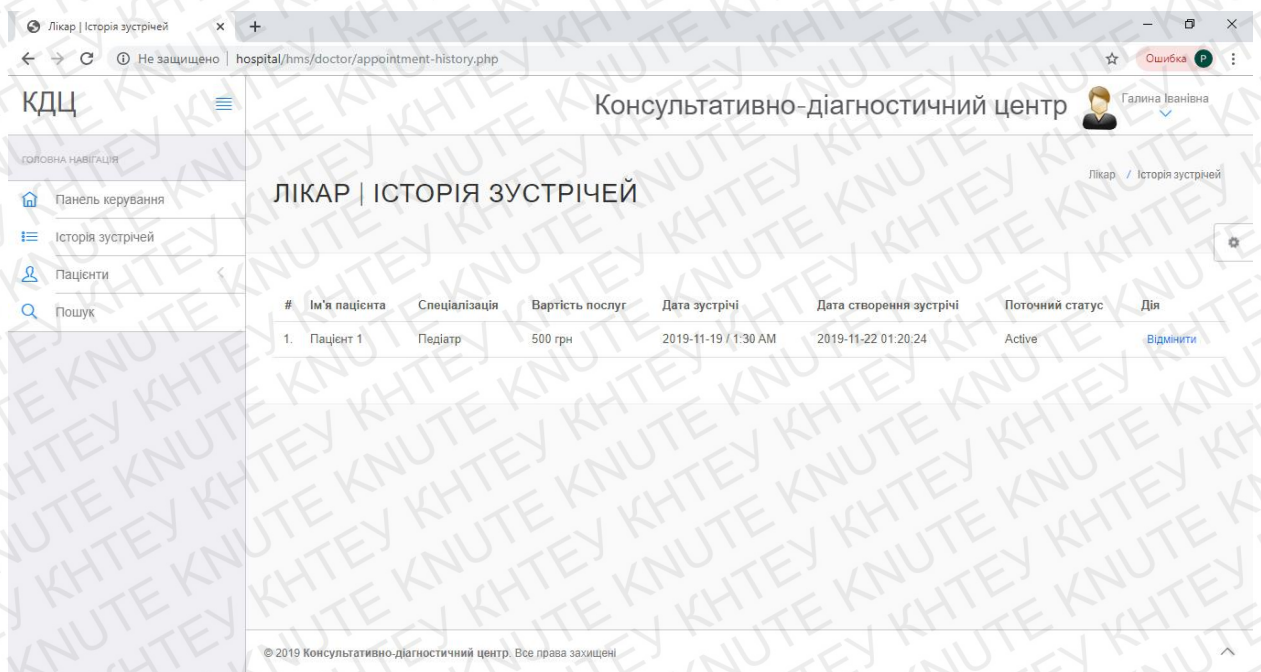


Рис 3.14. Історія зустрічей

Вивід інформації про зустрічі реалізований за допомогою даного коду:

```
<td class="center"><?php echo $cnt;?>.</td>
class="hidden-xs"><?php echo $row['fname'];?></td>
<td>
```

```

<td><?php echo $row['doctorSpecialization'];?></td>

<td><?php echo $row['consultancyFees'];?></td>

<td><?php echo $row['appointmentDate'];?> / <?php echo
$row['appointmentTime'];?>

</td>

<td><?php echo $row['postingDate'];?></td>

<td>

```

У лікаря є можливість відмінити зустріч, після чого у адміністратора з'явиться помітка про те, що зустріч була відмінена саме лікарем.

3.6. Висновок до розділу 3

Проаналізувавши та узагальнивши інформацію по розробці інформаційної системи, було створено інформаційну систему для консультативно-діагностичного центру з власною системою керування вмістом, яка реалізована переважно на мові PHP, частково з використанням Javascript. Було обрано вільну систему керування реляційними базами даних MySQL, яку використовують для створення динамічних інформаційних систем, оскільки має чудову підтримку з боку різноманітних мов програмування. Реалізовані наступні функції зі сторони пацієнта: зареєструвати свій обліковий запис, змінити пароль, записатися на зустріч з лікарем, обравши необхідну спеціалізацію, по якій система підбирає лікаря, відмінити зустріч при необхідності, зі сторони лікаря: переглянути поточні зустрічі, при необхідності є можливість відмінити, додати нового пацієнта, робота вже з існуючими, редагування власного профілю, зміна паролем. Зі сторони адміністратора реалізовано такі функції – керування користувачами, керування лікарями, реєстрація нових лікарів та додавання спеціалізацій, перегляд зустрічей, керування існуючими пацієнтами, перегляд повідомлень. Складання звітів, перегляд логів користувачів та лікарів, пошук пацієнта по контактному номеру.

ВИСНОВОК

В ході виконання випускної кваліфікаційної роботи було проведено системний аналіз об'єкту автоматизації та постановку задачі на проектування, а саме: досліджено роботу консультативно-діагностичного центру, та його організаційну структуру.

У процесі дослідження виявлено функції, що потребували автоматизації для подальшої реалізації у системі.

В результаті системного аналізу було розроблено інформаційну систему на основі власної системи керування інформацією, з використанням бази даних MySQL. Програмний код додатку можливо доповнювати додатковими функціями, він написаний з коментарями, та інтуїтивно зрозумілий.

Розроблена інформаційна система дозволяє повноцінно функціонувати консультативно-діагностичному центру, реалізована взаємодія пацієнтів з лікарями, зручне керування інформацією зі сторони адміністратора.

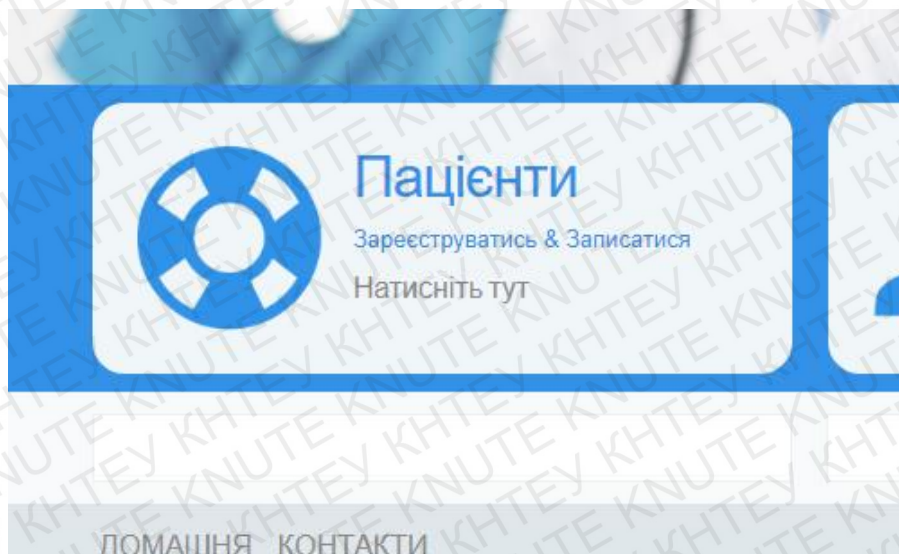
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Вебер Д. Технология Java в подлиннике :Пер. с англ. /Д. Вебер – СПб:ВНУ – Санкт-Петербург, 2000.- 1104 с..
2. Вонтинг Ларс Бо. Oracle Enterprise Manager 101 : пер. з англ. / Ларс Бо Вонтинг, Дирк Щепанек – К. : «Лори», 2005. – 480 с.
3. Гаврилова Т.А. Базы знаний интеллектуальных систем / Т.А. Гаврилова, В.Ф. Хорошевский / СПб: Питер, 2000.-384 с.
4. Ржеуцька С.Ю. - Базы данных. Язык SQL. / К. Вологда: ВоГТУ – 2012.
5. Гольцман В. MySQL 5.0. Библиотека программиста. (Интерфес с PHP, Perl, Java) / В. Гольцман - СПб: БХВ-Петербург, 2005. – 252с.
6. Грабер М. " SQL. Справочное руководство" / М. Грабер – М.: "Лори", 2000, - 291 с.
7. Давыдов С.В. IntelliJ IDEA. Профессиональное программирование на Java. / С.В. Давыдов, А.А. Ефимов – СПб.: БХВ – Петербург, 2005. – 800с.
8. Дейт К. Дж., Введение в системы баз данных, 8-е издание: Пер. с англ. / К. Дж. Дейт – К.; М.; СПб.: Издательский дом «Вильямс», 2005. – 1328 с.
9. Ебби Майкл. Oracle 9i: Первое знакомство : пер. з англ. / Майкл Ебби, Майкл Кори, Йен Амбросон – К. : «Лори», 2003. – 506 с.
10. Каучмен Джейсон С. OCA Oracle9i Associate DBA. Подготовка администраторов баз данных: пер. с англ. / Джейсон С. Каучмен, Судхир Н. Марисети – М.: Издательство «Лори», 2006. – 680с.
11. Кренке Д. Теория и практика построения баз данных. / Д. Кренке – СПб.: Питер, 2005. – 864 с.
12. Кузнецов М. Самоучитель MySQL 5 / М. Кузнецов, И. Самдянов – СПб.: БХВ-Петербург, 2007.- 560.
13. Маклаков С.В. BPro и Erwin CASE- средства разработки информационных систем / С.В. Маклаков - М.; Диалог-МИФИ, 2001.-306с.
14. Горбунов-Посадов М. М. Расширяемые программы. – М.: Полиптих, 1999. – 336 с. – ISBN 5-901-11801-4

15. Інформаційна система Байкальського державного університету економіки і права [Електронний ресурс]. – Режим доступу: <http://www.isea.ru/edu-portal/> (дата звернення 06.03.2014)
16. Інформаційна система «Електронний університет» Хмельницького національного університету [Електронний ресурс]. – Режим доступу: <http://isu1.tup.km.ua/isu/> (дата звернення 06.03.2014)
17. Гаврилова Т.А. Базы знаний интеллектуальных систем / Т.А. Гаврилова, В.Ф. Хорошевский / СПб: Питер, 2000.-384 с.
18. Ржеуцька С.Ю. - Базы даних. Мова SQL. / К. Вологда: ВоГТУ – 2012.
19. Тестування [Електронний ресурс]. – Режим доступу: <http://www.yiiframework.com/doc/guide/1.1/uk/test.overview>
20. Гольцман В. MySQL 5.0. Библиотека программиста. (Интерфес с PHP, Perl, Java) / В. Гольцман - СПб: БХВ-Перербург, 2005. – 252с.
21. Т. Карпова. Базы данных. Модели, разработка, реализация. СПб.: Питер, 2001. - 304 с.

Керівництво користувача (пацієнта)

Для початку роботи необхідно перейти у меню авторизації



Створити новий обліковий запис. Для цього потрібно ввести персональні дані, та дані для входу в обліковий запис

Реєстрація пацієнта

Авторизація

Введіть ваші персональні дані:

Іванов Іван Іванович

вул. Ревуцького 1, кв.52

Київ

Стать

☐ Жіноча ☒ Чоловіча

Введіть дані вашого облікового запису:

☒ ivanov@ukr.net
Email доступний для реєстрації

☒ Я погоджуюсь

Вже маєте обліковий запис? [Авторизуватися](#)

[Підтвердити](#)

Далі перейти у меню авторизації та ввести логін і пароль

Авторизація

Вхід в обліковий запис

Будь ласка введіть свій логін і пароль

 ivanov@ukr.net



[Забули пароль ?](#)

Увійти 

Ви ще не маєте обліковий запис? [Створили обліковий запис](#)

© 2019 Консультативно-діагностичний центр. Усі права захищені

Для запису на прийом до лікаря потрібно натиснути «записатися» у вікні «записатися на візит»

КДЦ

ОСНОВНА НАВІГАЦІЯ

 Панель приладів

 Записатися

 Історія зустрічей

 Медична історія

Консультативно-діагностичний центр

Іванов Іван Іванович

Користувач / Панель керування

КОРИСТУВАЧ | ПАНЕЛЬ КЕРУВАННЯ

 Мій профіль
Оновити профіль

 Мої візити
Історія записів

 Записатися на візит
Записатися

© 2019 КОНСУЛЬТАТИВНО-ДІАГНОСТИЧНИЙ ЦЕНТР. Всі права захищені

Обираємо зі списку «Спеціалізація лікаря» необхідну, після чого у строці «Лікар» обрати необхідного лікаря, якого система підбирає відповідно до спеціалізації, та обрати дату і час візиту

Записатися

Спеціалізація Лікаря

Педіатр

Лікар

Галина Іванівна

Вартість

500 грн

Дата

2019-11-26

Час

10:45 AM

приклад : 10:00 PM

Підтвердити

Після чого можна переглянути свої записи, для цього потрібно обрати пункт меню в основній навігації «Історія зустрічей»

КДЦ

Консультативно-діагностичний центр

Іванов Іван Іванович

Користувач / Історія зустрічей

ОСНОВНА НАВИГАЦІЯ

Панель приладів

Записатися

Історія зустрічей

Медична історія

КОРИСТУВАЧ | ІСТОРІЯ ЗУСТРІЧЕЙ

#	Ім'я Лікаря	Спеціалізація	Вартість	Дата зустрічі	Дата створення зустрічі	Поточний статус	Дія
1.	Галина Іванівна	Педіатр	500 грн	2019-11-26 / 10:45 AM	2019-11-26 04:49:26	Активний	Відмінити

© 2019 КОНСУЛЬТАТИВНО-ДІАГНОСТИЧНИЙ ЦЕНТР. Всі права захищені

Якщо по деяким обставинам немає можливості прийти на обраний час, запис можна відмінити, натиснувши «відмінити».

Для зміни паролю потрібно у випадаючому меню обрати «Змінити пароль»

Консультативно-діагностичний центр



Іванов Іван Іванович

Мій профіль

Змінити пароль

Завершити сеанс



У поле «Поточний пароль» потрібно ввести старий пароль, у поля «Новий пароль» та «Підтвердити пароль» необхідно ввести новий пароль.

КДЦ

Консультативно-діагностичн

ОСНОВНА НАВІГАЦІЯ

 Панель приладів

 Записатися

 Історія зустрічей

 Медична історія

КОРИСТУВАЧ | ЗМІНИТИ ПАРОЛЬ

Змінити пароль

Поточний пароль

Введіть поточний пароль

Новий пароль

Новий пароль

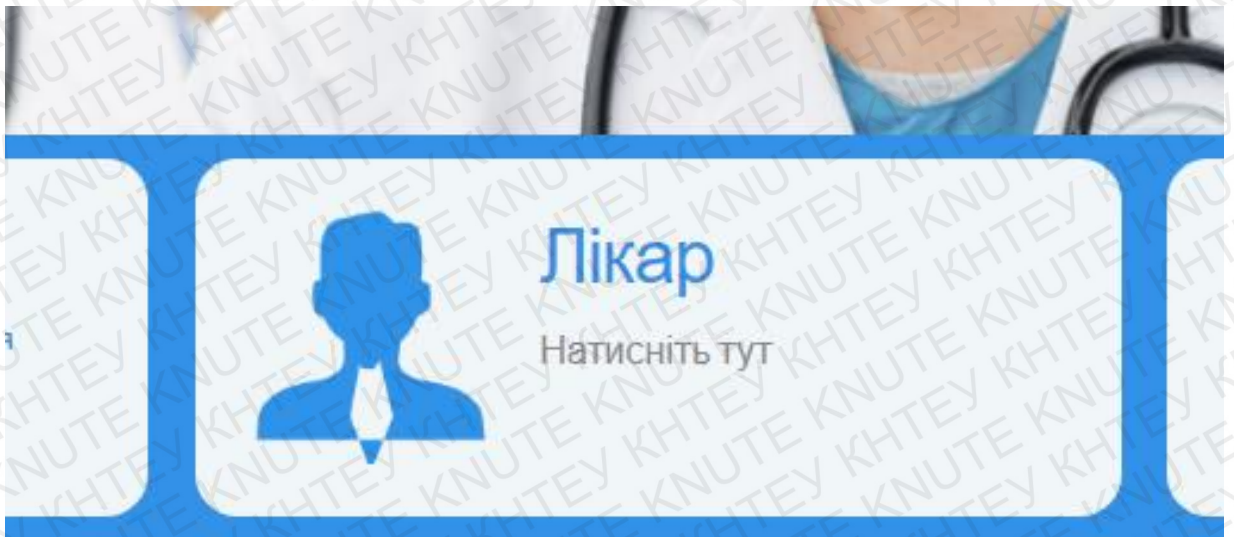
Підтвердити пароль

Підтвердьте пароль

Підтвердити

Керівництво користувача (лікаря)

Для початку роботи необхідно перейти у меню авторизації лікаря.



Ввести свій логін та пароль і натиснути кнопку «Увійти».

Авторизація Лікаря

[Увійти в обліковий запис](#)

Введіть ваш логін та пароль.

 likar1@ukr.net



[Забули пароль ?](#)

Увійти 

© 2019 Консультативно-діагностичний центр. Всі права захищені

Для перегляду запланованих зустрічей необхідно натиснути «Історія записів» у вікні «Мої зустрічі»

ЛІКАР | ПАНЕЛЬ КЕРУВАННЯ



Мій профіль

[Оновити профіль](#)



Мої зустрічі

[[Історія записів](#)]

При необхідності є можливість відмінити запис, відповідна помітка буде відображатись у панелі керування адміністратора

ЛІКАР | ІСТОРІЯ ЗУСТРІЧЕЙ

Лікар / [Історія зустрічей](#)



#	Ім'я пацієнта	Спеціалізація	Вартість послуг	Дата зустрічі	Дата створення зустрічі	Поточний статус	Дія
1.	Пацієнт 1	Педіатр	500 грн	2019-11-19 / 1:30 AM	2019-11-22 01:20:24	Active	Відмінити
2.	Іванов Іван Іванович	Педіатр	500 грн	2019-11-26 / 10:45 AM	2019-11-26 04:49:26	Active	Відмінити

Для того щоб додати дані пацієнта у базу даних потрібно у меню навігації натиснути на пункт меню «Пацієнти» і обрати «Додати пацієнта»

ГОЛОВНА НАВІГАЦІЯ



Панель керування



Історія зустрічей



Пацієнти

Додати пацієнта

Керування пацієнтами



Пошук

Для внесення запису в базу, потрібно заповнити дані пацієнта

Додати пацієнта

Ім'я пацієнта

Іванов Іван Іванович

Контактний номер пацієнта

0636336363

Email пацієнта

ivanov@ukr.net

Email available for Registration

Стать



Жіночий



Чоловічий

Адреса пацієнта

вул. Ревуцького 1, м. Київ

Вік пацієнта

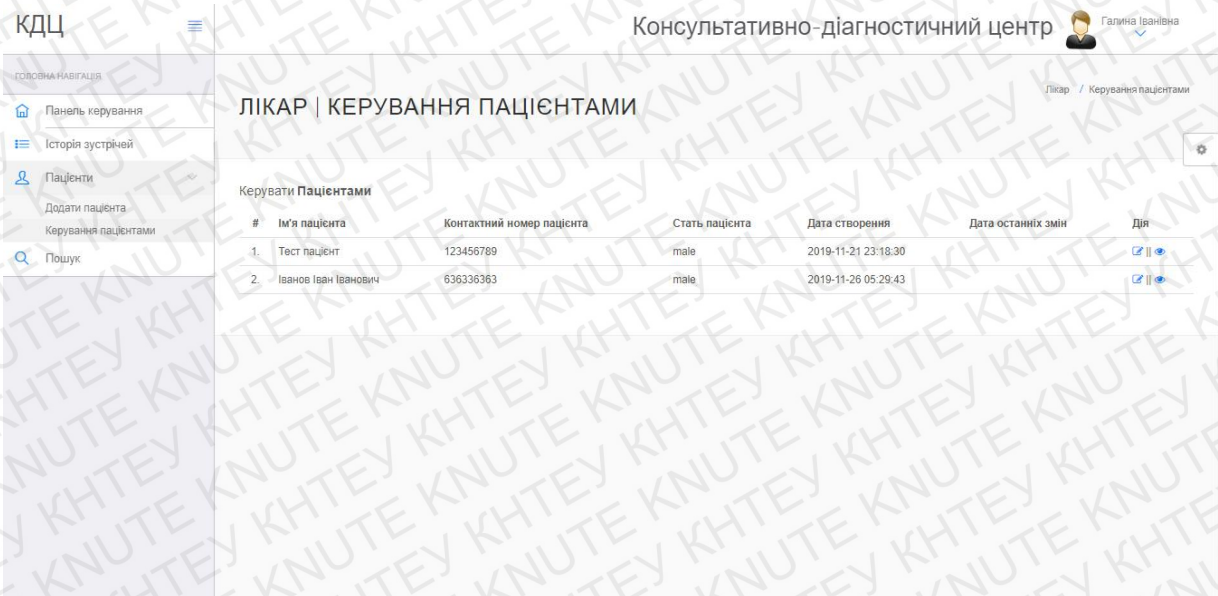
25

Медична історія

Пацієнт має підвищений артеріальний тиск

[Додати](#)

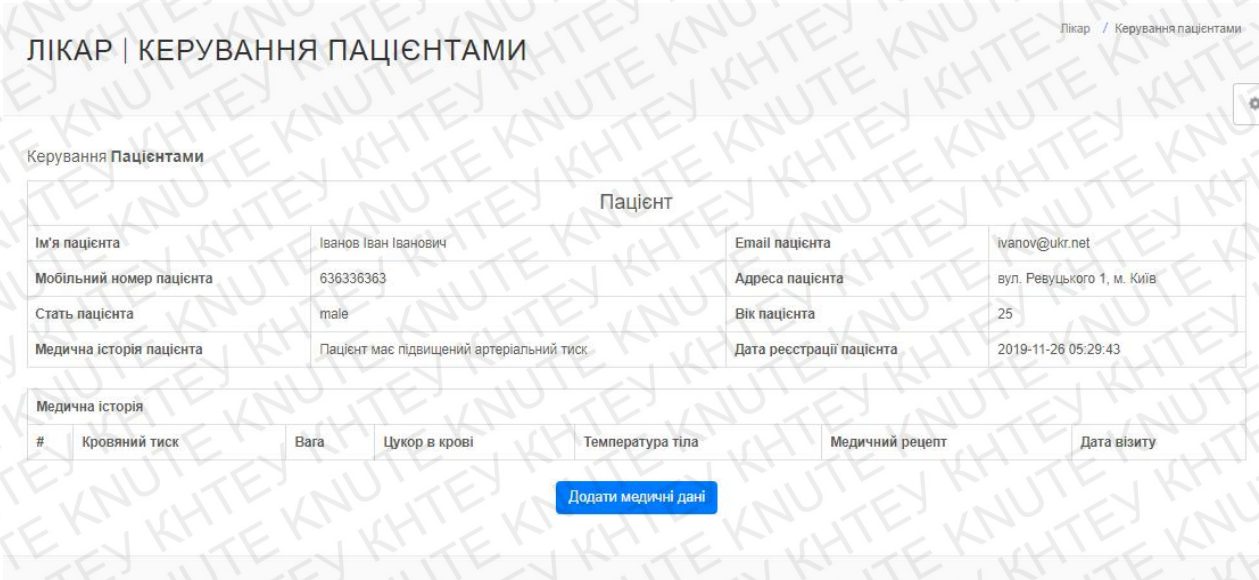
Для перегляду існуючих в базі пацієнтів потрібно у меню навігації обрати пункт меню «Пацієнти» и натиснути на «Керування пацієнтами»



Для того щоб додати дані обстеження потрібно натиснути на піктограму ока у колонці «Дія»



Після чого натиснути на кнопку «Додати медичні дані»



У створеному вікні потрібно ввести дані обстеження, та натиснути кнопку «Підтвердити»

Додати медичну історію



Кров'яний тиск :

142/90

Цукор в крові :

4,5

Вага :

93

Температура тіла :

36,7

Рецепт :

Зменшити вживання солі та уникнути вживання алкоголю. Прийти на повторне обстеження через 2 тижні.

Закрити

Підтвердити

Київський національний торговельно-економічний університет

Кафедра програмної інженерії та кібербезпеки

РЕФЕРАТ

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Розробка інформаційної системи Консультативно-діагностичного центру»

Студента 2м курсу, 5 групи,
спеціальності
«Інженерія програмного
забезпечення»
спеціалізації
«Інженерія програмного
забезпечення»

підпис студента

Кінзерського
Костянтина Юрійовича

Науковий керівник
к.т.н., доцент

підпис керівника

Рзаєва С.Л.

КИЇВ – 2019

АНОТАЦІЯ

До випускної кваліфікаційної роботи Кінзерського Костянтина Юрійовича на тему «Розробка інформаційної системи консультативно-діагностичного центру»

У випускній кваліфікаційній роботі розглядається розробка клієнт-серверної інформаційної системи в який входить мова програмування PHP, фреймворку Bootstrap, та база даних MySQL .

Ключові слова: інформаційна система, клієнт-серверна архітектура, мова програмування, PHP, javascript, база даних MySQL.

ANNOTATION

Prior to the final qualification work of Kinzerskyi Kostiantyn Yuriyovuch on the topic " Development of information system of consulting and diagnostic center"

The final qualification examines the development of a client-server information system that includes PHP programming language, Bootstrap framework, and MySQL database.

Keywords: information system, client-server architecture, programming language, PHP, javascript, MySQL database.

ЗАГАЛЬНА ХАРАКТЕРИСТИКА РОБОТИ

Дана випускна кваліфікаційна робота присвячена інформаційній системі Консультативно-діагностичного центру. Було проведено аналіз сучасних технологій які дозволяють створити ефективну інформаційну систему. Для розробки інформаційної системи представлено в даній роботі були використані такі технології розробки: мова програмування PHP, javascript, база даних MySQL, мова гіпертекстової розмітки HTML, каскадні таблиці стилів CSS, фреймворк Bootstrap.

Актуальність обраної теми полягає в тому, що в наш час створення інформаційної системи відіграють значну роль в успішності будь-якого

підприємства. Термін "інформаційна система" належить до класу програмних продуктів, що автоматизують ведення бізнесу. Система називається інформаційною, якщо вона підтримує інформаційне забезпечення бізнесу.

Інформаційна система – це сукупність засобів збору, зберігання, передачі, оброблення інформації для досягнення поставленої мети у процесі управління.

Для створення власної інформаційної системи неможливо обійтися без використання баз даних. Важливою особливістю СУБД є забезпечення виконання запитів до бази даних. Ще однією важливою особливістю більшості сучасних СУБД є забезпечення режиму мультидоступу.

Основні функції СУБД - безпосереднє керування даними у зовнішній пам'яті. Ця функція включає збереження та ведення структурної інформації (даних), перетворення даних за запитом на структурну інформацію для користувача. При цьому повинні забезпечуватися: простота використання системи; можливість автономного функціонування при порушеннях мережі або при адміністративних потребах; високий ступінь ефективності.

Мета дослідження – розробити інформаційну систему для консультативно-діагностичного центру.

Об'єкт дослідження – діяльність консультативно-діагностичного центру.

Предмет дослідження – розробка інформаційної системи для консультативно-діагностичного центру.

Задачі дослідження визначити функціональні можливості майбутньої інформаційної системи, вимоги до інтерфейсу.

Основні задачі в рамках цієї випускної-кваліфікаційної роботи:

- визначити вхідну і вихідну інформацію;
- визначити основні інструменти для розробки програмного продукту;
- визначити основні функції системи.

Методи дослідження – Розроблено інформаційну систему для підвищення ефективності роботи консультативно-діагностичного центру.

Наукова новизна полягає у використанні нових та актуальних методів розробки інформаційної системи, які дозволяють оптимізувати роботу Консультативно-діагностичного центру, реалізують зручну реєстрацію в системі

користувачів(пацієнтів), запис на зустріч до лікаря по обраним критеріям, лікарі перед прийомом мають можливість ознайомитись з медичною історією та доповнити її після зустрічі, вносити в базу нових пацієнтів, та результат обстеження, адміністратор інформаційної системи має повний доступ до інформації про пацієнтів, лікарів, має можливість додавати в систему нові спеціалізації, керувати та додавати нових лікарів, переглядати всі дії користувачів.

Практична цінність дослідження полягає у використанні інформаційної системи у консультативно-діагностичному центрі.

Структура та обсяг роботи складається з вступу, трьох розділів, висновків, списку використаних джерел та додатку. Містить 20 рисунків, 11 таблиць. Ключові слова: інформаційна система, мова програмування, PHP, javascript, база даних MySQL.

ОСНОВНИЙ ЗМІСТ ДИПЛОМНОЇ РОБОТИ

У **вступі** визначено актуальність теми, об'єкт, мету предмет, задачі та методи, що використовувались у ході дослідження.

Перший розділ «Теоретичні відомості розробки та побудови інформаційної системи» присвячений визначенню основних задач системи, принципам функціонування консультативно-діагностичного центру.

Другий розділ «Архітектура та моделювання програмного забезпечення» присвячений визначенню архітектури та моделюванню інформаційної системи, визначено програмне середовище розробки, організаційну структуру інформаційної системи, розглянуто існуючі застосування для вирішення завдання.

У **третьому розділі** «Створення інформаційної системи консультативно-діагностичного центру» розглянуто основні принципи створення та проектування бази даних, реалізацію інтерфейсу пацієнта, лікаря та адміністратора системи за допомогою стеку технологій: мови програмування PHP, javascript, бази даних MySQL, реалізація графічної частини на основі HTML, CSS з використанням фреймворку Bootstrap.

