

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ

До захисту допускається
Завідувач кафедри
_____ Скарга-Бандурова І.С.
« ____ » _____ 20__р.

МАГІСТЕРСЬКА РОБОТА

НА ТЕМУ:

Дослідження та реалізація комп'ютерної системи інтернет-аукціону

Освітньо-кваліфікаційний рівень “Магістр”
Спеціальність 123 – “Комп'ютерна інженерія” (освітня програма – “Комп'ютерні системи та мережі”)

Науковий керівник роботи:

_____ (підпис) _____ Д.О. Недзельський (ініціали, прізвище)

Консультант з охорони праці:

_____ (підпис) _____ Я.О. Критська (ініціали, прізвище)

Студент:

_____ (підпис) _____ К.А. Прищепін (ініціали, прізвище)

Група:

_____ КСМ-163м _____

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки
Кафедра Комп'ютерної інженерії
Освітньо-кваліфікаційний рівень магістр
Напрямок підготовки _____
(шифр і назва)
Спеціальність 123 – “Комп'ютерна інженерія”
(шифр і назва)

ЗАТВЕРДЖУЮ:

Завідувач кафедри _____
_____ І.С. Скарга-Бандурова
« _____ » _____ 2017 р.

**ЗАВДАННЯ
НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ**

Прищепіну Костянтину Андрійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження та реалізація комп'ютерної системи інтернет-аукціону

керівник проекту (роботи) к.т.н., доц. Недзельський Д.О.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від “ 18 ” 10 2017 року № 207/48

2. Строк подання студентом проекту (роботи) 12.01.2018

3. Вихідні дані до проекту (роботи) матеріали науково-дослідницької практики

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

Аналіз сервісів що надають послуги інтернет-аукціонів. Розробка структури бази даних MySQL. Функціональність проекту. Розробка і впровадження функціоналу проекту. Розробка допоміжного ПЗ. Проведення експериментів. Охорона праці

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслеників)
електронні плакати

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	Критська Яна Олександрівна		

7. Дата видачі завдання 18.10.2017

Керівник

(підпис)

Завдання прийняв до виконання

(підпис)**КАЛЕНДАРНИЙ ПЛАН**

№ п/п	Найменування етапів дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1	Аналіз стану питання у науковій літературі. Визначення вимог до роботи.	18.10.2017 – 16.11.2017	
2	Аналіз та вибір платформи розробки та архітектури додатку.	17.11.2017 – 22.11.2017	
3	Аналіз методів проектування системи інтернет-аукціону на основі UML та методів компонентного проектування.	23.11.2017 – 28.11.2017	
4	Розробка структури бази даних MySQL. Налагодження підключення до БД. Створення нащадка Active Record. Перевірка працездатності	29.11.2017 – 07.12.2017	
5	Розгортання проекту на основі FrameworkYii2. Функціональність.	08.12.2017 – 12.12.2017	
6	Розробка і впровадження функціоналу проекту.	13.12.2017 – 20.12.2017	
7	Розробка заходів з охорони праці.	21.12.2017 – 24.12.2017	
8	Оформлення пояснювальної записки і графічного матеріалу.	25.12.2017 – 06.01.2018	
9	Підготовка та подання магістерської роботи до захисту	07.01.2018 – 12.01.2018	

Студент

(підпис)

Прищепін К.А..

(прізвище та ініціали)

Науковий керівник

(підпис)

Недзельський Д.О.

(прізвище та ініціали)

АНОТАЦІЯ

Прищепін К.А. Дослідження та реалізація комп'ютерної системи інтернет-аукціону.

Розглянуто програмні та інструментальні засоби. В рамках проекту досліджені особливості архітектури систем, призначені для розробки інтернет-ресурсів. Запропоновано спосіб розробки, вибір якого заснований на дослідженні існуючих проектів і готових систем для реалізації. Розроблено архітектуру проекту, створена візуальна і функціональна програмні частини інтернет-аукціону на основі фреймворку Yii2.

Ключові слова: розробка, фреймворки, архітектура систем, аукціон

АНОТАЦИЯ

Прищепин К. А. Исследование и реализация компьютерной системы интернет-аукциона.

Рассмотрены программные и инструментальные средства. В рамках проекта исследованы особенности архитектуры систем, предназначенные для разработки интернет-ресурсов. Предложен способ разработки, выбор которого основан на исследовании существующих проектов и готовых систем для реализации. Разработана архитектура проекта, создана визуальная и функциональная программные части интернет-аукциона на основе фреймворка Yii2.

Ключевые слова: разработка, фреймворки, архитектура систем, аукцион.

ABSTRACT

Prischepin K. A. Research and implementation of the computer system for online auction.

Software and tools are considered. Within the framework of the project, the features of the system architecture, designed for the development of Internet resources, are investigated. A method of development is proposed, the choice of which is based on the study of existing projects and ready-made systems for implementation. The architecture of the project is developed, the visual and functional program parts of the Internet auction are created on the basis of the Yii2 framework.

Keywords: development, frameworks, system architecture, auction.

ЗМІСТ

ВСТУП.....	9
1 АНАЛІЗ МЕТОДІВ І ЗАСОБІВ ПОБУДОВИ ІНТЕРНЕТ АУКЦІОНІВ. ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕНЬ	10
1.1 Особливості розробки он-лайн аукціонів.....	10
1.2 Аналіз сервісів що надають послуги інтернет-аукціонів.....	11
1.3 Технічне завдання на розробку	11
1.3.1 Вимоги до дизайну	11
1.3.2 Типи користувачів	13
1.3.3 Профілі користувачів	14
1.3.4 Картка товару.....	15
1.3.5 Додавання товару і характеристики	16
1.3.6 Ціни на товар.....	16
1.3.7 Кошик	17
1.3.8 Основні розділи сайту	17
2 МЕТОДИ І ЗАСОБИ ПОБУДОВИ ІНТЕРНЕТ-АУКЦІОНУ	19
2.1 Засоби розробки серверної частини програми.....	19
2.1.1 Вибір платформи розробки та архітектури додатку	19
2.2 Загальна структура додатку Yii	25
2.3 Методи проектування системи інтернет-аукціону на основі UML та методів компонентного проектування.....	28
2.3.1 Проектування з UML-позначеннями	28
2.3.2 Компонентне програмування	33
2.4 Короткий опис засобів розробки	34
2.4.1 Редактор коду.....	34
2.4.2 Локальний веб-сервер	35
2.4.3 MySQL-клієнт	35
2.4.4 Система контролю версій	36
3 ПРАКТИЧНА РЕАЛІЗАЦІЯ ПРОЕКТУ	37
3.1 Розробка структури бази даних MySQL.....	37

	7
3.2 Підготовка бази даних.....	38
3.3 Налагодження підключення до БД.....	39
3.4 Створення нащадка Active Record.....	40
3.5 Створення Action	40
3.6 Створення View.....	42
3.7 Перевірка працездатності.....	42
3.8 Розгортання проекту на основі FrameworkYii2	43
3.8.1 Установка Yii2	43
3.8.2 Налаштування веб сервера	44
3.8.3 Рекомендовані настройки Apache	44
3.8.4 Рекомендовані настройки Nginx	45
3.8.5 Запуск програми	46
3.9 Функціональність.....	46
4 РОЗРОБКА І ВПРОВАДЖЕННЯ ФУНКЦІОНАЛУ	48
4.1 Реєстрація	48
4.2 Авторизація	53
Існує кілька способів створення реєстрації та авторизації користувачів. Насамперед необхідно:	53
4.3 Генерація коду за допомогою Gii.....	54
4.4 Генерація класу Active Record.....	55
4.5 Створення CRUD	55
4.6 Фільтрація.....	56
4.7 Оброблювач прайс-листів	57
4.8 Статична верстка CSS по макетах.....	58
4.8.1 Використання LESS.....	59
4.8.2 Розмітка. Система сітки.	60
4.8.3 Випадаюче меню	61
4.8.4 Групи форми введення і пошук.....	61
4.8.5 Media-запити	62
4.8.6 Анімація слайдера	64

5 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ	69
5.1 Загальні питання з охорони праці	69
5.1.1 Правові та організаційні основи охорони праці	70
_Тос504217595	
5.2 Аналіз стану умов праці	73
5.2.1 Вимоги до приміщень	73
5.2.2 Вимоги до організації місця праці	74
5.2.3 Навантаження та напруженість процесу праці	75
5.3 Виробнича санітарія	76
5.3.1 Аналіз небезпечних та шкідливих факторів при роботі на ПК.....	76
5.3.2 Пожежна безпека	78
5.3.3 Електробезпека	80
5.4 Гігієнічні вимоги до параметрів виробничого середовища.....	81
5.4.1 Мікроклімат	81
5.4.2 Освітлення.....	81
5.4.3 Шум та вібрація, електромагнітне випромінювання	85
5.4.4 Вентилювання.....	86
5.5 Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій.....	86
5.6 Охорона навколишнього природного середовища.....	87
5.6.1 Загальні дані з охорони навколишнього природного середовища	87
5.6.2 Вимоги до збору, пакування та розміщення відходів ІТ галузі	88
5.6.3 Визначення впливу та заходів щодо поводження з відходами ІТ галузі	89
5.7 Висновки до розділу	93
ВИСНОВКИ.....	94
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....	95

ВСТУП

Обґрунтування вибору теми досліджень. Нещодавно межа між звичайними користувачами і тими, хто створює інформацію в Інтернеті зникла: замість статичних сайтів з'явилась можливість виконувати покупки та продажі в режимі реального часу. Разом з тим, задачі розробки і підтримки засобів он-лайн торгівлі вимагають вирішення цілої низки питань, серед яких врахування типів користувачів, адміністрування, захист інформації тощо. Тому обґрунтованою є тема магістерської роботи, у якій вирішується науково-прикладне завдання розроблення моделей для реалізації систем аукціонів на базі веб-додатків.

Об'єкт дослідження – процеси побудови онлайн аукціонів реального часу.

Предмет дослідження – моделі та методи, що використовуються для розробки системи інтернет-аукціону.

Мета і завдання дослідження. Метою дослідження є підвищення ефективності роботи з даними з інтернету та підтримки он-лайн торгівлі за рахунок організації спеціалізованої платформи.

Для досягнення мети дослідження необхідно вирішити такі **завдання**:

- аналіз методів та засобів побудови інтернет-аукціонів;
- виявлення особливостей розробки інтернет-аукціонів;
- проектування інформаційної технології та розробка допоміжних програмних засобів.

Наукова новизна отриманих результатів:

Дістали подальшого розвитку методи побудови системи інтернет-аукціону за рахунок використання UML та методів компонентного проектування .

Особистий внесок здобувача полягає у розробленні нових моделей та інструментальних засобів, що дозволяють вирішити поставлені задачі. Усі основні результати отримані автором особисто.

Практичне значення отриманих результатів полягає у тому, що основні наукові положення дисертації реалізовані у виді розрахункових моделей та програмних засобів, які утворюють прикладну інформаційну технологію проектування автоматизованих систем підтримки он-лайн торгівлі.

Структура та обсяг роботи. Магістерська робота складається зі вступу, п'яти розділів, висновків, списку використаних джерел і додатків. Загальний обсяг роботи складає 114 сторінки, з яких основний текст на 95 сторінках, список використаних джерел із 8 найменувань, додатки. Робота містить 10 таблиць та 23 рисунків.

1 АНАЛІЗ МЕТОДІВ І ЗАСОБІВ ПОБУДОВИ ІНТЕРНЕТ АУКЦІОНІВ. ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕНЬ

У розділі проаналізовані особливості, програмні та інструментальні засоби для побудови інтернет-аукціону, моделі і методи вирішення задачі розробки засобів для побудови спеціалізованого програмного забезпечення, проведений початковий експеримент для виявлення проблем, поставлена наукова задача та обґрунтована методика досліджень.

1.1 Особливості розробки он-лайн аукціонів

Головною особливістю розробки сайту аукціону є робота з двома цільовими аудиторіями: Продавці та Замовники. На сайті аукціону повинна бути можливість гнучкого і швидкого пошуку по каталогу продавців. Товари повинні шукатися за різними параметрами. Ці параметри повинні залежати від галузі аукціону. Важливим додатковим фактором є зручна система масової заливки інформації Замовників і Продавців на портал аукціону. Більш просунутий варіант - це надання корпоративним клієнтам і партнерам можливості програмного доступу до каталогу через API, що прискорить процеси і знизить витрати.

Основні етапи розробки сайту аукціону:

1. Створення концепції проекту

Створюється клієнтом самостійно або у взаємодії з виконавцем. Включає в себе відповіді на наступні питання: хто буде цільовою аудиторією, що є продуктом, яка буде бізнес-модель сайту, основні сервіси сайту, ролі в системі і структура. Також може включати більш специфічні моменти, пов'язані з вашою предметною областю.

2. Створення технічного завдання

Створюється клієнтом самостійно або у взаємодії з виконавцем. Включає в себе відповіді на наступні питання: хто буде цільовою аудиторією, що є продуктом, яка буде бізнес-модель сайту, основні сервіси сайту, ролі в системі і структура.

3. Проект розробки по етапах

Після написання технічного завдання укладається договір на розробку движка проекту. Саме ТЗ є підставою для договору, який визначає терміни і вартість робіт.

4. Впровадження в експлуатацію

Саме від цього етапу залежить наскільки плавно вийде інтегрувати сайт бізнес-процеси.

Дана стадія характеризується цілим комплексом різних робіт: налаштування сервера, автоматичне створення резервних копій даних сайту, моніторинг параметрів сайту, зміна тестових налаштувань на бойові, seo-аудит, перевірка контенту на сайті, загальне тестування.

5. Супровід і розвиток

Сайт повинен постійно розвиватися і змінюватися в ногу з бізнесом. Це безперервний ітеративний процес поліпшень. Супровід включає в себе обслуговування сервера, створення нових модулів для сайту, додавання нового контенту на сайт, веб-аналітика, оптимізація реклами та пошукове просування.

1.2 Аналіз сервісів що надають послуги інтернет-аукціонів

Пошук в інтернеті готових рішень показує, що в лідерах для електронної комерції велика кількість безкоштовних систем управління контентом з відкритим вихідним кодом і тільки MODx в його варіаціях позиціонує себе як Framework.

Всі системи позиціонують себе як основа для інтернет-магазину і надають розробникові основний інструментарій функцій, таких як SEO оптимізація, зручний інтерфейс адміністративної панелі, готові шаблони дизайну а також величезний вибір плагінів і доповнень написані і поширювані через Інтернет набори практично під будь-яку задачу, значно спрощуючи її рішення.

Аналіз та пошукова видача на першій сторінці пошукової системи за запитом "Інтернет-аукціон України" показав, що аукціони використовують за свою основу наступні системи: WordPress, Joomla, MODx, а також індивідуальні рішення.

1.3 Технічне завдання на розробку

1.3.1 Вимоги до дизайну

Кросбраузерна і адаптивна верстка. Сайт повинен розтягуватися на будь-яку ширину сторінки.

Надані макети зроблені по сітці Bootstrap.

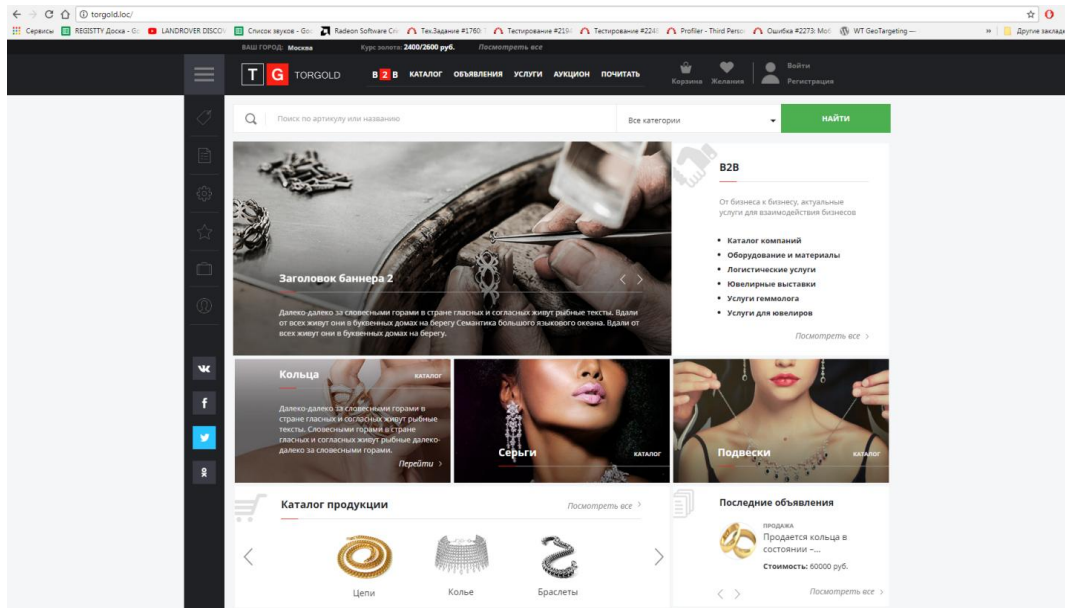


Рисунок 1.1 - макет сайта

Банер - якщо банерів даного типу в системі немає, вся область банеру не показується.

Меню, розташоване праворуч від банера містить пункти блоку послуг, що пропонуються зареєстрованими компаніями

- 1) Каталог компаній
- 2) Устаткування і матеріали
- 3) Логістичні послуги
- 4) Ювелірні виставки
- 5) Послуги геммолога
- 6) Послуги для ювелірів

Механізм вибору поточного магазину - область містить випадаюче вікно, що дає можливість вибору магазину: двоетапний вибір міста, потім магазину.

Вхід в розділ користувача (вспливаюча форма авторизації). Якщо користувач авторизований, замість напису «авторизація» відображається його ПІБ або назва магазину, який представляє користувач.

Верхнє меню - містить посилання на загальну інформацію (оплата, доставка, гарантія, контакти). Меню може бути змінено за допомогою стандартних системних механізмів.

Система пошуку - в це поле введення можна ввести повне або часткове назва товару, після введення більше чотирьох символів, система починає пропонувати у вигляді списку товари за збігом підстрок поточного введення.

З запропонованого списку може здійснюватися перехід на картку товару. Можливий варіант пошуку, при якому у полі введення друкується назва товару + Enter, після чого на окремій сторінці система видає список товарів відповідних умовам, використовується посторінкове уявлення.

1.3.2 Типи користувачів

Тип користувача:

- Приватна особа
- Компанія

Система повинна надавати користувачам можливість зареєструватися та здійснювати діяльність як приватна особа, так і юридична, діючи від імені компанії-виробника.

Реєстрація користувача має на увазі під собою надання персональних даних з подальшим їх зберіганням і обробкою на сервері.

Надані дані проходять валідацію на коректність введення, поля відмічені зірочкою обов'язкові до заповнення повинні підсвічуватися, якщо були заповнені, користувачу при цьому виводиться відповідне повідомлення про тип помилки.

Користувач в процесі реєстрації заповнює наступні поля:

- Вид користувача
- Ім'я
- E-mail (логін)
- Пароль
- Підтвердження пароля
- Посилання на правила торговельного майданчика
- Підтвердження згоди

При реєстрації, за замовчуванням, користувач - приватна особа. Далі, при бажанні, в своєму офісі він може зареєструвати на порталі компанію. І в цьому випадку, всі здійснювані ним дії будуть виконуватися від імені компанії.

Приватна особа може:

- Переглядати кошик і оформляти замовлення.
- Відправляти і приймати повідомлення (система чату).
- Давати обмежену кількість "певних" оголошень.
- Створювати картку "послуга"
- Створювати картку "оголошення"

Приватна особа не може:

- Створювати картки товару.
- Створювати аукціонні пропозиції.
- Розміщувати на сайті контент (новини, статті)

Компанія може:

- Переглядати кошик і оформляти замовлення.
- Відправляти і приймати повідомлення (система чату).
- Давати обмежену кількість "певних" оголошень.
- Створювати картку "послуга"
- Створювати картку "оголошення"
- Створювати картки товару.
- Створювати аукціонні пропозиції.
- Розміщувати на сайті контент (новини, статті)

За замовчуванням користувач, зареєстрований від імені компанії має можливість використовувати наступний функціонал:

- Виставляти товари на продаж.
- Публікувати оголошення.
- Публікувати свої новини.
- Отримувати відгуки.
- Виставляти товар на аукціон
- Надавати послуги

Замовлення послуги здійснюється через форму, компанія отримує повідомлення, що відвідувач майданчики цікавиться даною послугою.

Користувач, що реєструється від імені компанії може створювати, змінювати, видаляти товарні позиції.

1.3.3 Профілі користувачів

Після проходження процедури реєстрації та авторизації на сайті користувач в залежності від обраного типу користувача має можливість додатково заповнити наступні типи полів:

Для приватної особи:

- Місто
- Адреса
- Номер телефону

- Додатковий номер телефону
- Skype

Для компанії:

- Тип компанії
- Місто
- Назва компанії
- Адреса сайту
- Торгівля в роздріб
- Торгівля оптом
- Є офіційним дилером
- Доставка кур'єром
- Відправлення в регіони
- Повна адреса
- ПІН
- Свідоцтво про реєстрацію
- Широта на карті, для відображення місцезнаходження
- Довгота на карті, для відображення місцезнаходження
- Номер телефону
- Додатковий номер телефону
- E-mail адреса
- Skype
- Вказати контактну особа
- Телефон контактної особи
- E-mail контактної особи
- Опис компанії
- Логотип

1.3.4 Картка товару

У картку товару заповнюються поля і вносяться наступні дані:

- Тип, вид товару (виріб, послуга)
- Артикул
- Назва товару
- Ціна
- Новинка

- Опис
- Характеристики товару
- Фото товару
- Рубрика товару
- Спеціальна ціна
- Оптова ціна

1.3.5 Додавання товару і характеристики

Товар на сайті компанія може створити наступними способами:

- 1) Вручну, поштучно з адміністративної панелі.
- 2) Скопіювавши в іншій компанії на сайті
- 3) Через завантаження компанією файлу типу XML
- 4) Через завантаження Excel-файлом

У товарів є супутні характеристики для виведення в картці і фільтрації. Для кожного розділу будуть створені свої характеристики.

Характеристики можуть бути вказані:

- 1) Зі списку
- 2) Додані вручну

1.3.6 Ціни на товар

Ціни на товар оновлюються або завантажуються файлом від компанії (по артикулу компанії)

- Через файл типу XML
- Вручну

Компанії можуть на окремі товари, групи товарів, усі товари встановлювати властивість "оптова ціна", а користувачі що представляють інші компанії - бачити оптові ціни. Для приватних осіб поле "оптова ціна" за замовчуванням приховано.

Всі види товарів:

- можна покласти в кошик.
- можна розмістити в оголошенні.
- можна виставити на аукціон.

1.3.7 Кошик

Розділ "Кошик" повинен містити перелік замовлених товарів з назвою, артикулом, фотографіями, цінами і загальною сумою до оплати, розміром знижки, інформацією про доставку, Блоки для введення персональної інформації, вибору способу доставки і оплати.

В адміністративній панелі сайту буде передбачено ведення історії покупок з можливістю присвоєння статусу, пошук товару по артикулу, пошук товару за назвою.

1.3.8 Основні розділи сайту

Послуги - Послуга пов'язана з однією рубрикою в каталозі послуг. У кожній послугі свої характеристики, фото і опис. Компанія вибирає зі списку яку вони надають послугу, вказує вартість і заповнює її характеристики.

Фотографія послуги - компанія або вибирає зі списку або завантажує на сайт власну.

Опис заповнюється на вибір:

- Автоматичний опис, створений системою
- Вказується компанією вручну.

Вказати певну послугу може тільки компанія.

Ціни та варіанти вибору виведення вартості послуги:

- 1) Фіксована ціна.
- 2) Ціна тільки ВІД
- 3) Ціна ВІД і ДО
- 4) Договірна

Оголошення - пов'язано з однією рубрикою в каталозі оголошень.

Поля, що вказуються при створенні оголошення

- Тип оголошення
- Категорія
- Регіон продавця
- Текст оголошення
- Стан товару
- Ціна
- Доставка

Приватна особа: може давати обмежену кількість оголошень.

Компанія: безкоштовно 50 оголошень. При додатковій платні - без обмежень. Оголошення можна буде платно виділяти кольором, піднімати в пошуку. А також розміщувати в блоці «Важливі оголошення»

Новини - можуть публікувати адміністрація сайту та компанії
Рубрикатора в новинах немає, замість нього поле "вид новини"

вид:

- Новини порталу
- Новини компаній
- Виставки
- Інше

Статті - Статтю публікує адміністратор сайту.

Рубрикатора статей немає, замість нього поле "вид статті"

вид:

- Це цікаво
- Новинки
- Законодавство

Аукціон - Виставляти товар на аукціон може:

- 1) Уповноважений (з правами) адміністратор сайту.
- 2) Компанія, якій дав дозвіл уповноважений (з правами) адміністратор сайту.

Рубрикатор - містить список рубрик, до яких прив'язуються товари і послуги. Перший рівень каталогу визначає загальний набір характеристик для всіх категорій товарів, усі нижчестоящі категорії автоматично працюють з першим рівнем

Розділ B2B - не містить окремих рубрикатор, компанія прив'язує себе до безлічі рубрик, методом фільтрації виводиться список компаній, які працюють з обраною рубрикою. Розділ містить специфічні послуги - перевезення, зберігання, митні послуги, юридичний супровід угод.

Компанія може прив'язуватися до певних розділів рубрикатора B2B.

«Послуги та обладнання» - для обладнання на кожній сторінки розділу «Послуги та обладнання» просто буде виділена жирним шрифтом посилання до каталогу у відповідний розділ ("Устаткування і матеріали").

2 МЕТОДИ І ЗАСОБИ ПОБУДОВИ ІНТЕРНЕТ-АУКЦІОНУ

У даному розділі представлені результати проведеного аналізу існуючих серверних мов програмування і сучасних фреймворків. Спираючись на отримані результати, проведено вибір платформи розробки і архітектури додатку. Представлені результати проектування бази даних та базової конфігурації програми.

2.1 Засоби розробки серверної частини програми

2.1.1 Вибір платформи розробки та архітектури додатку

Пристаюючи до розробки web-додатку необхідно, перш за все, враховувати той факт, що основне навантаження припадає на серверну частину. Тому, типовою є задача підібрати оптимальну мову серверного програмування, який дозволив би реалізувати функціонал програми. До таких мов відносять: PHP, Perl, Python, Java, Ruby, ASP.NET (C#, Visual Basic, Jscript і інші .NET мови). Важливою особливістю будь-якого серверної мови програмування є можливість безпосередньої взаємодії з системою управління базами даних (СУБД).

2.1.2 Порівняння популярних серверних мов програмування

На сьогоднішній день найпопулярнішими мовами web-програмування є PHP, Python, Ruby. Саме ці три скриптових серверних мови програмування найчастіше вибирають для розробки серверних додатків.

Компанія ТЮВЕ провела порівняння цих трьох мов згідно п'ятьма критеріями

Існуючі сайти:

	PHP	RUBY	PYTHON
SITES BUILT USING IT	<ul style="list-style-type: none"> WIKIPEDIA UDEMY FACEBOOK 	<ul style="list-style-type: none"> TWITTER HULU GROUPON 	<ul style="list-style-type: none"> YOUTUBE GOOGLE

Рисунок 2.1 - Сайти написані на відповідних мовах

- Кількість вакансій для розробників

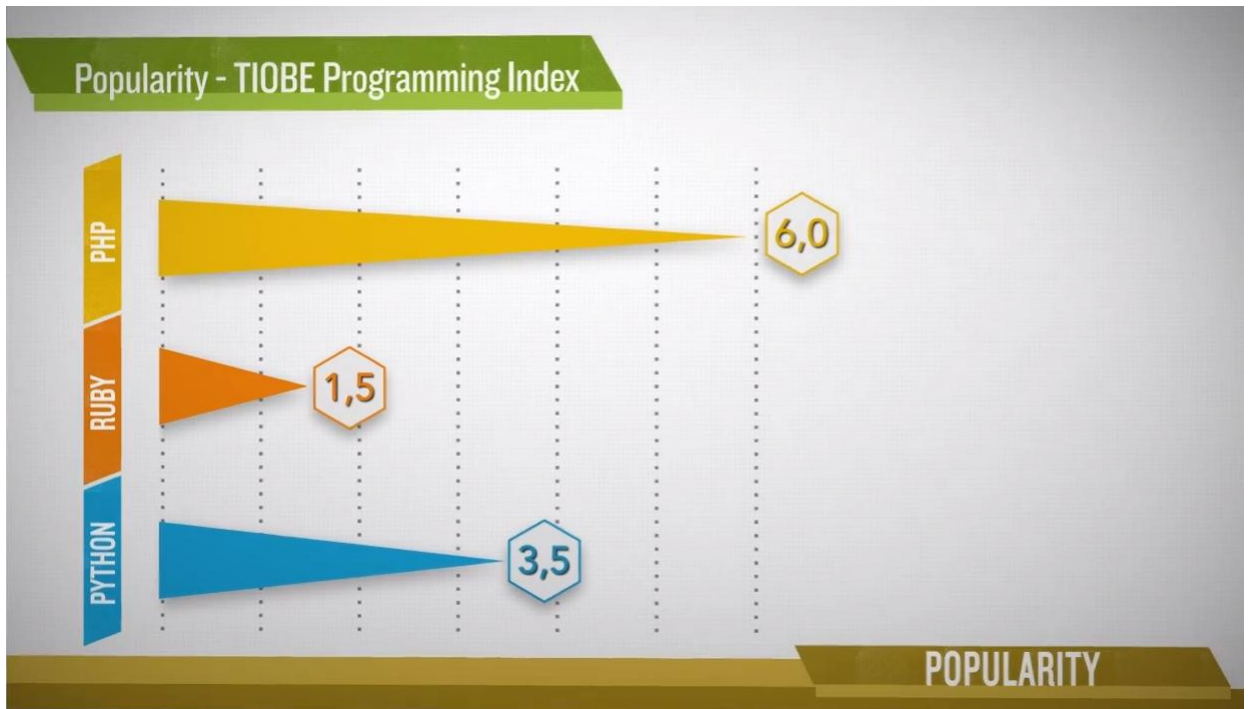


Рисунок 2.2 - Індекс вакансій

- Usability(зручність)

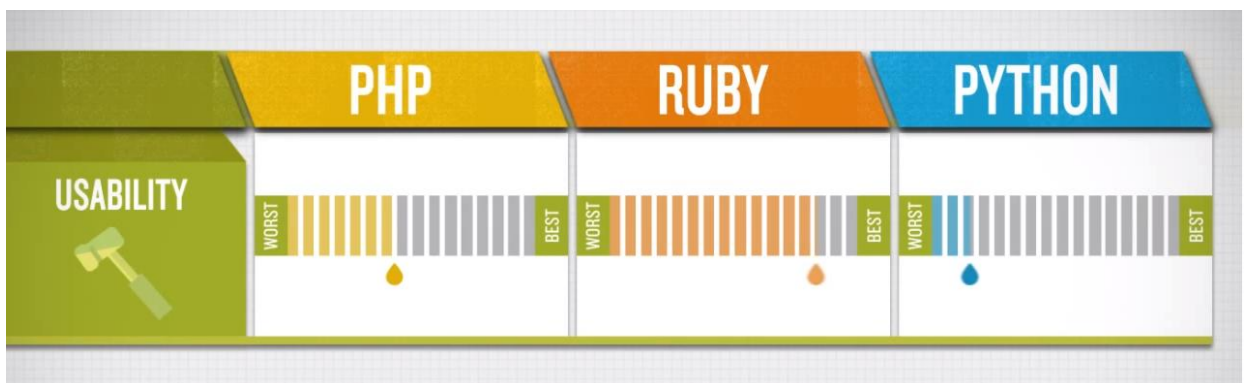


Рисунок 2.3 - Зручність використання

- Поріг входження

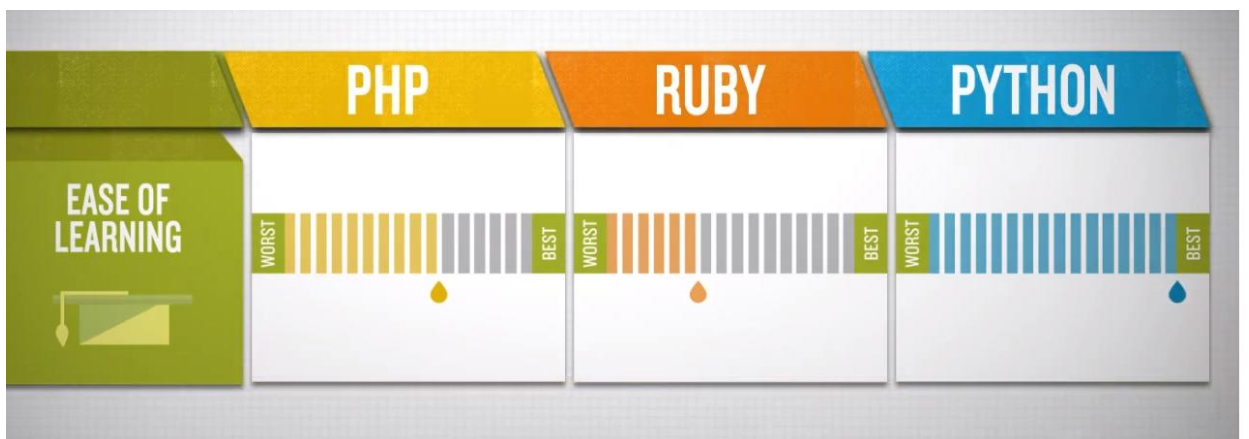


Рисунок 2.4 - Простота вивчення

- Кількість розробників

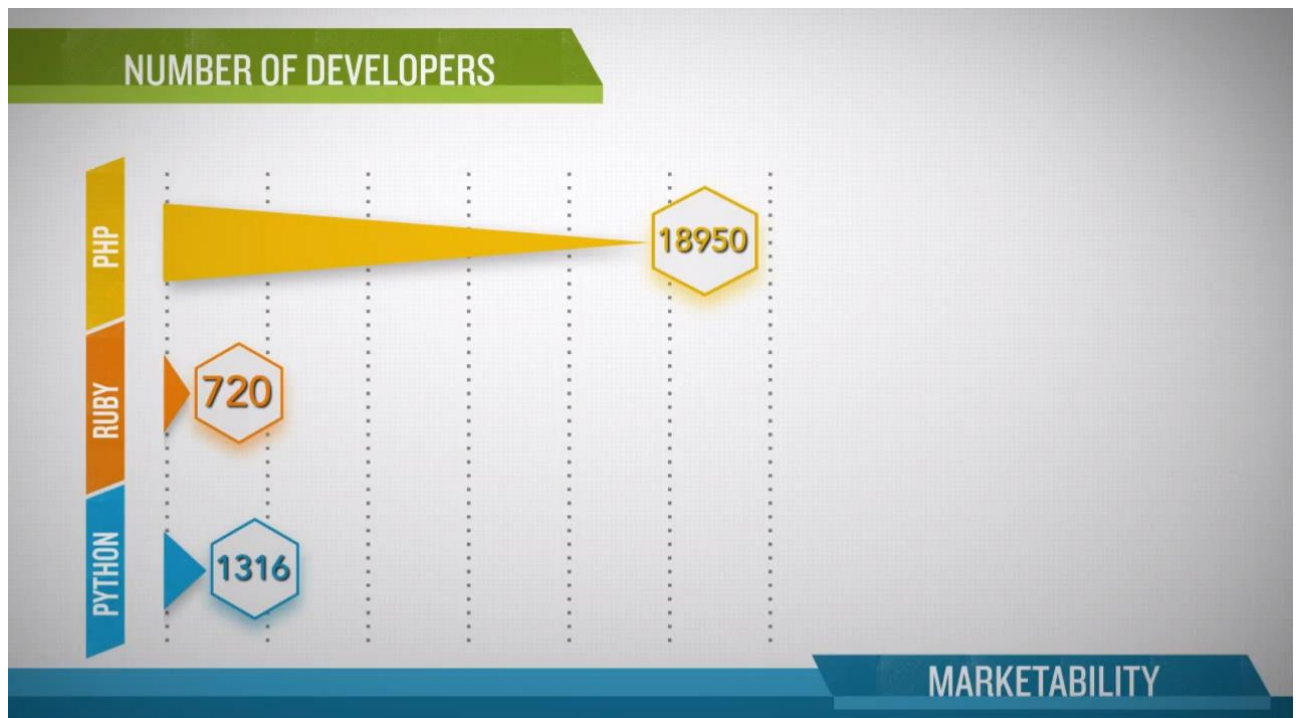


Рисунок 2.5 - Індекс кількості розробників

Згідно з проведеним дослідженням, PHP значно випереджає своїх конкурентів. Незважаючи на те що за останній рік Python набирає все більшої популярності, слід відзначитися що на PHP написано найбільшу кількість сайтів, у цієї мови одне з найбільших спільнот розробників. Завдяки цьому PHP постійно розвивається, з'являється все більше корисних базових розширень. Також PHP може похвалитися великим вибором користувача фреймворків і систем управління контентом (CMS).

2.1.3 Призначення Frameworks та CMS

Після вибору платформи необхідно визначити архітектуру проекту. Сьогодні все менше проектів розробляються без використання на базовому функціоналі PHP. Розширення і надбудови значно розширюють можливості платформи, що дозволяє збільшити швидкість і якість розробки.

Найчастіше використовуються системи управління контентом (CMS), або фреймворки. Основний плюс систем управління контентом - для їх використання досить базових знань мови програмування на якому вони написані, дуже низький поріг входження. Вони засновані за принципом «конструктора». До базового каркасу підключаються доступні модулі. Найчастіше можливість «кастомізації» модулів

мінімальна. Також у проєктів на CMS спостерігаються проблеми з безпекою. «Дірки» в вихідному коді зустрічаються практично по всіх CMS, що часто приводить до злому сайту. Найчастіше, системи управління контентом використовують для розробки невеликих проєктів не розрахованих на велику кількість відвідувачів. Найпопулярніші PHP CMS - це Wordpress, Drupal, Joomla.

На відміну від CMS, фреймворки є не "конструктором» написаному на PHP, а скоріше надбудовою для нього.

Фреймворк визначається як безліч конкретних і абстрактних класів, а також визначень способів їх взаємодій. Конкретні класи зазвичай реалізують відносини між класами. Абстрактні класи є точками розширення, в яких каркаси можуть бути використані або адаптовані.

Точка Розширення - це та частина фреймворка, для якої не приведена реалізація. Відповідно каркас концептуальної моделі складається з концептуальних класів, а каркас програмної системи з класів мови програмування загального призначення.

Процес створення фреймворка складається в Вибори підмножини завдань проблеми і їх реалізації. В ході реалізації спільних методів вирішення завдань знаходяться в конкретних класах, а змінні елементи вносяться в точки Розширення.

Перш за все важливо зрозуміти, що фреймворк полегшує життя тільки тим програмістам, які на хорошому рівні знають мову програмування який лежить в основі фреймворка. Фреймворк реалізує тільки основні функції, наприклад роботу з СУБД, завантаження файлів, висновки і обробка стандартних html-елементів та інше. Тобто найпоширеніші функції, більшість методів все одно доведеться писати самому. Фреймворк робить за програміста його роботу як CMS, він тільки допомагає і прискорює процес розробки. З плюсів використання фреймворка можна виділити:

- дозволяє розробнику не відволікатися на дрібниці, а зосередитися на логіці проєкту;
- значно прискорює розробку;
- дозволяє реалізувати проєкти будь-якої складності;
- всі проєкти будуть мати одну і ту ж структуру файлів, що дозволяє легко інтегрувати раніше написані скрипти в нові розробки;
- всі фреймворки мають розширюваність, тобто підтримують підключення користувальницьких плагінів і модулів.

Іншими словами фреймворк дозволяє створити каркас додатка, навколо якого буде проводиться подальша розробка.

2.1.4 Порівняння PHP Frameworks

Згідно зі щорічним рейтингом «SitePoint framework popularity survey» [<http://www.sitepoint.com/best-php-framework-2015-sitepoint-survey-results/>] в 2015 році в п'ятірку лідерів по використанню увійшли PHP-фреймворки Laravel [<http://laravel.com/>], Symfony2 [<https://symfony.com/>], Nette [<http://nette.org/>], Codeigniter [<http://www.codeigniter.com/>] та Yii 2 [<http://www.yiiframework.com/>].

2.1.4.1 Laravel

Laravel пропонує одну з найпотужніших реалізацій Active Record в PHP. Laravel дотримується паттерна MVC. Можлива гнучка настройка маршрутизації. Підтримується кешування запитів в СУБД. Вбудовані шаблони коду, реалізовані на движку Blade. Засоби тестування коду. Вбудований обробник подій. Валідація даних. Оброблювач черг. Підтримка міграцій. Є підтримка консольних команд. Ви можете налаштовувати оточення для різних версій проекту. Установка і оновлення ядра фреймворку і модулів через composer.

2.1.4.2 Symfony 2

Додатки в Symfony 2 мають модульну структуру, модулі називаються бандл (bundle). У Symfony існує дуже багато консольних команд. Symfony дотримується паттерна MVC. Налаштування маршрутизації відбувається при установці каркаса додатки. Робота з базою даних здійснена через Doctrine. Doctrine є повноцінною ORM, але побудована вона за іншим принципом ніж ActiveRecord. Як шаблонізатора Symfony використовує Twig - один з кращий PHP-шаблонізатор.

2.1.4.3 Nette

Особливості Nette:

- підтримка HTML5;
- підтримка PHP 5.3;
- вбудовані в HTML макроси шаблонізатора;
- технологія Context-Aware Escaping;
- конфігурації в форматі NEON, на основі яких генерується PHP код;
- власна реалізація базового класу для об'єктів Nette \ Object;
- робота з подіями;
- функції зворотного виклику (callbacks);

- новий потоковий протокол `safe: //` для атомарного доступу до файлової системи.

2.1.4.4 CodeIgniter

Особливості CodeIgniter:

- використовується модель MVC (Модель-Відображення-Контролер);
- підтримується безліч баз даних (MySQL, PostgreSQL, MSSQL, SQLite, Oracle);
- докладна документація з прикладами;
- дуже швидкий в роботі.

2.1.4.5 Yii 2

В Yii 2 робота з базою даних реалізована за допомогою Active Record і DAO. Вбудована підтримка інтерналізації. Yii 2 дотримується паттерна MVC. Гнучка настройка маршрутизації. Кешування запитів до СУБД, сторінок і окремих ділянок коду. Підтримка всіх функцій PHP 5.4. Робота з namespaces. Обробка помилок і виключень. Вбудоване тестування коду. Підтримка міграцій. Величезне співтовариство. Гарна документація. Є робота з консольними додатками. Є можливість генерації шаблонів. Є можливість налаштовувати оточення для різних версій проекту. Установка і оновлення ядра фреймворку і модулів через composer.

Всі наведені вище фреймворки об'єднані в таблиці 2.1.

Таблиця 2.1 Порівняння PHP-фреймворків

Критерій	Laravel	Symfony 2	Nette	CodeIgniter	Yii 2
Опис	MVC, підтримка модулів, дуже багато методів «з коробки», ActiveRecord	MVC, Twig, Doctrine, бандли	Швидкий, вбудовані макроси в шаблонизатор, технологія Context-Aware Escaping	MVC, швидкий.	MVC, ActiveRecord, DAO, велика база модулів. кешування даних.
Необхідний рівень знань.	PHP, ООП	PHP5, ООП, консоль, ORM	PHP5, ООП.	PHP5, ООП.	PHP, ООП
Проекти	Всі типи	Великі	Невеликі, середні	Невеликі, середні	Всі типи
PHP 5.4	Підтримує	Піддерживает	Немає	Немає	Підтримує
Жорстка структура каталогів	Немає	Так	Так	Так	Немає

Продовження таблиці 2.1

Критерій	Laravel	Symfony 2	Nette	CodeIgniter	Yii 2
Офіційна підтримка інтернаціоналізації	Присутня	Присутня	Немає	Присутня	Присутня
Повна підтримка ORM	Присутня	Присутня	Немає	Присутня	Присутня
Документація і приклади	Гарна	Гарна	Нормальна	Гарна	Гарна
Unit-тести для вихідного коду фреймворка	Так	Так	Немає	Немає	Так

Перевага Yii2 над іншими фреймворками полягає в ефективності, широких можливостях і якості документації. Yii2 успадкував всі переваги минулої версії, і значно розширив функціонал відповідно до сучасних вимог, з урахуванням дослідження функціоналу найбільш популярних фреймворків. Це результат роботи цілої спільноти розробників з усього світу.

2.2 Загальна структура додатку Yii

Всі сторінки додатка використовують суміжний хедер (шапка сайту) і футер (підвал). В "шапці" знаходиться головне меню, за допомогою якого користувач переміщається по сайту. В "підвалі" - копірайт і загальна інформація.

У самій нижній частині вікна при роботі в режимі розробки будуть доступні системні повідомлення Yii - журнал, налагоджувальна інформація, повідомлення про помилки, запити до бази даних та інші. Відображенням цієї інформації керує вбудований відладчик, він записує і відображає інформацію про хід виконання програми.



Рисунок 2.6 - Вбудований відладчик Yii

У доповнення до веб-додатку є консольний скрипт з назвою `yii`, який знаходиться в базовій директорії додатку. Цей скрипт може бути використаний для виконання фонових завдань і послуг у програмі.

В `Yii` реалізована схема проектування модель-вид-контролер (MVC), яка відповідає структурі директорій додатку. В директорії `models` знаходяться Моделі, в `views` розташовані Види, а в каталозі `controllers` все Контролери додатку.

Рисунок 2.7 демонструє внутрішній устрій додатку.

У кожній програмі `Yii` є точка входу в додаток, `web / index.php` це єдиний PHP-скрипт доступний для виконання з `Web`. Він приймає вхідний запит і створює екземпляр додатку. Додаток обробляє вхідні запити за допомогою компонентів і відправляє запит контролеру. Віджети використовуються в Видах для побудови динамічних інтерфейсів сайту.

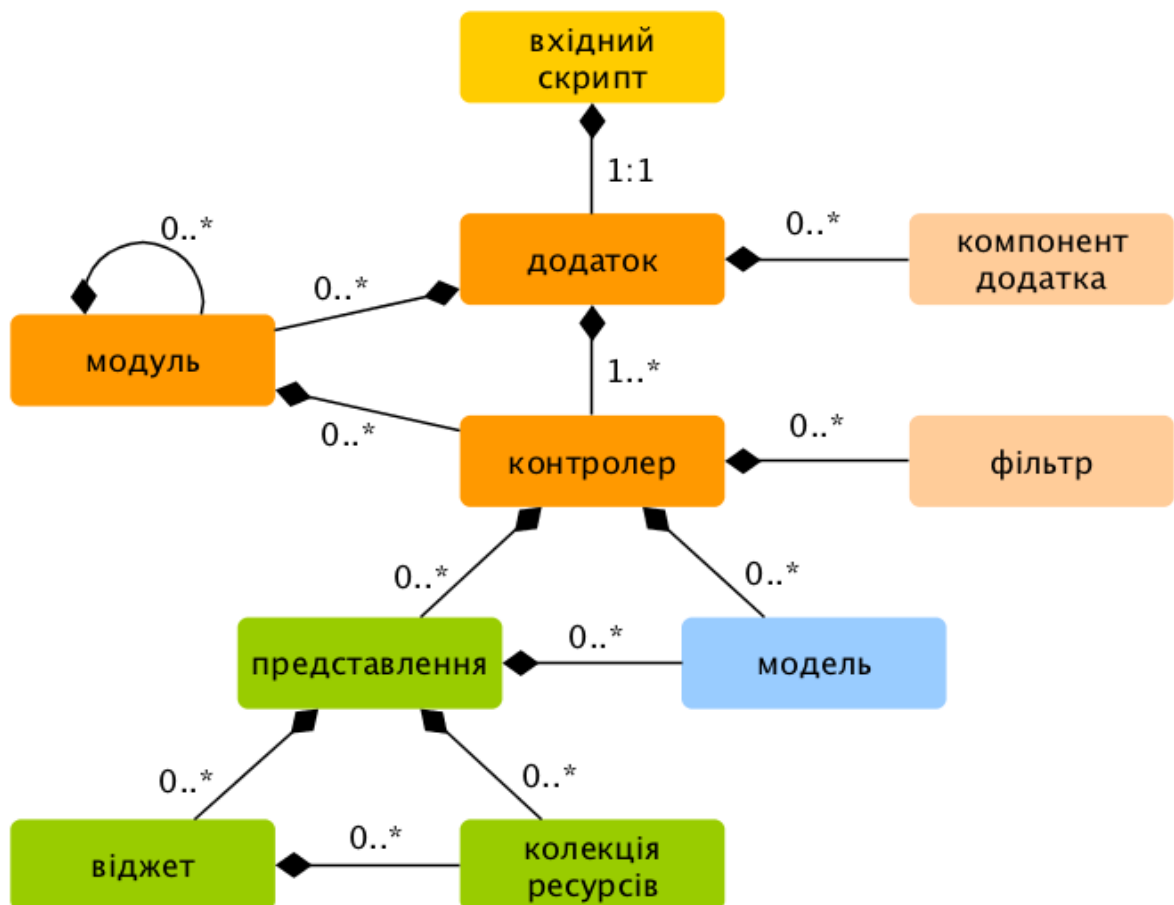


Рисунок 2.7 - Внутрішній устрій `Yii` додатку

На зображенні 2.8 показано як додаток обробляє запит.

1. Користувач звертається до точки входу web / index.php.
2. Скрипт завантажує конфігурацію configuration і створює екземпляр додатку для подальшої обробки запиту.
3. Додаток визначає маршрут запиту за допомогою компонента додатку - запит.
4. Додаток створює екземпляр контролера для виконання запиту.
5. Контролер, в свою чергу, створює дію і накладає на нього фільтри.
6. Якщо хоча б один фільтр дає збій, виконання програми зупиняється.
7. Якщо всі фільтри пройдені - додаток виконується.
8. Дія завантажує модель даних. Найімовірніше з бази даних.
9. Дія генерує вид, відображаючи в ньому дані (в т.ч. і отримані з моделі).
10. Створений вигляд програми передається як компонент "відповідь".
11. Компонент "відповідь" відправляє готовий результат роботи програми браузеру користувача.

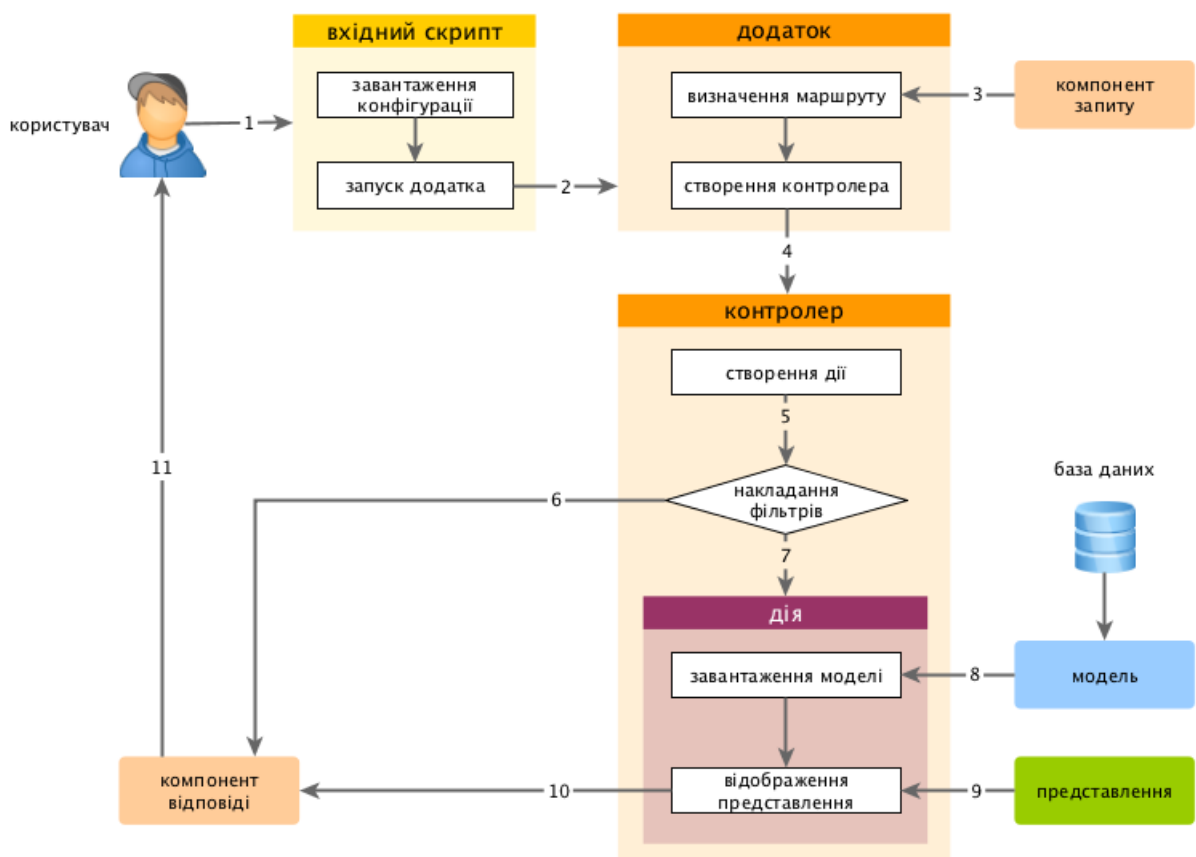


Рисунок 2.8 - Життєвий цикл призначеного для користувача запиту

2.3 Загальна конфігурація додатку

Загальна конфігурація додатку включає в себе проектування бази даних і настройку підключення до СУБД, визначення регіональних параметрів, настройка логування серверних процесів, підключення та налаштування необхідних модулів (RBAC, модуль інтерналізації). Також повинні бути налаштовані компоненти програми (кешування, authmanager та інше).

На рис. 2.5 представлена загальна схема конфігурації програми

Після установки програми повинні бути розроблені моделі для взаємодії з сутностями бази даних і відповідні контролери для рендеринга відображень і обробки інформації користувачей.

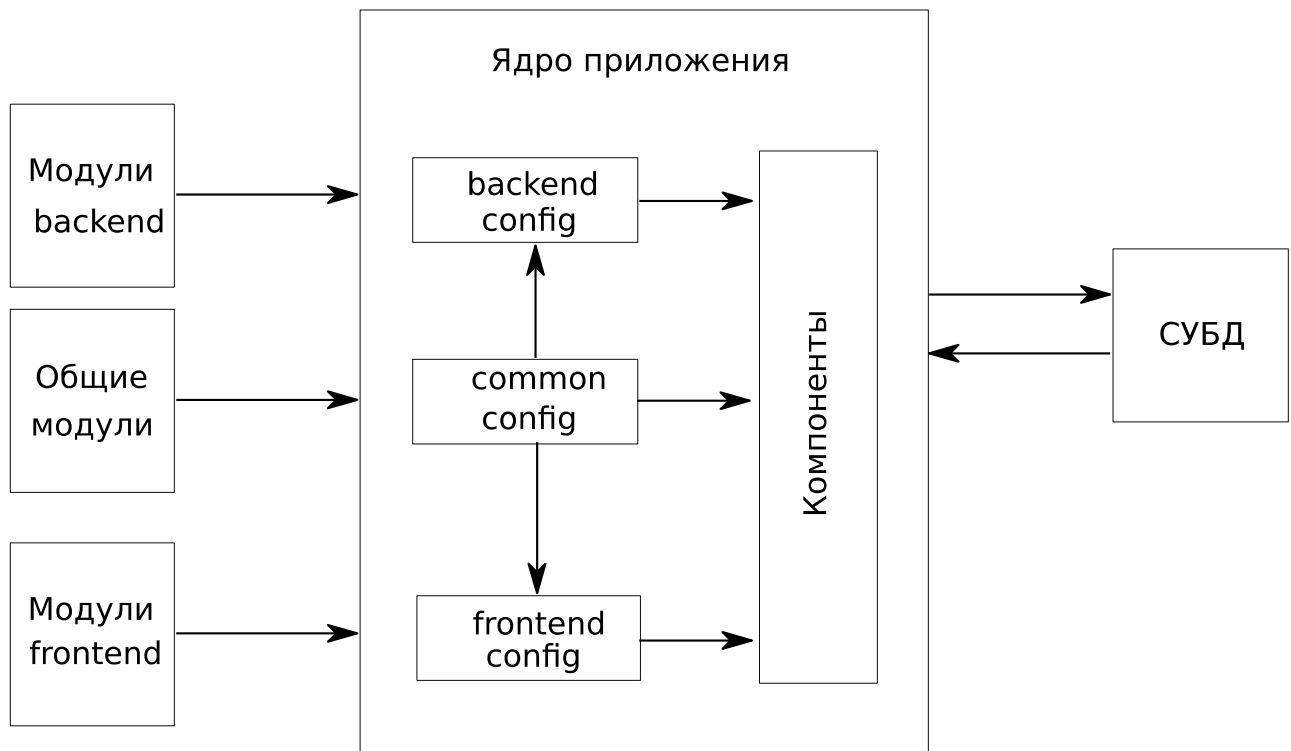


Рисунок 2.9 Схема конфігурації програми

2.3 Методи проектування системи інтернет-аукціону на основі UML та методів компонентного проектування

2.3.1 Проектування з UML-позначеннями

UML - це уніфікована мова моделювання для надання стандартизованої позначки для опису моделей, орієнтованих на об'єкт. Проте для того, щоб ефективно застосовувати позначення UML, воно повинно використовуватися разом із методом Object-Oriented

Analysis and Design. Об'єктно-орієнтований аналіз та дизайн (OOAD) описує спільноту методологій для розробки програмного забезпечення на основі бізнес-компонентів. Методологія визначає життєвий цикл розробки системи, що визначає завдання та результати в об'єктно-орієнтованому проекті. Використовуючи комбінацію UML-позначень та Object-Oriented Analysis and Design, ми можемо скоротити життєвий цикл розвитку системи, легко підтримувати систему та покращувати багаторазове використання модулів. Програмування на основі компонентів також призводить до збереження та багаторазового використання коду.

Система аукціону з підтримкою Інтернету використовує чотири діаграми UML та три компоненти, які можуть бути багаторазово використані на рівні специфікації.

Розробка системи онлайн-аукціону вимагає прийняття рішень та вибору технологій для підтримки цих рішень.

Система аукціону була розроблена, використовуючи Case Case, Sequence, Class і компонентні діаграми, запропоновані UML та інструментом Rational Rose.

Діаграма з використанням - це візуалізація користування, тобто взаємодія між системою аукціону та користувачами.

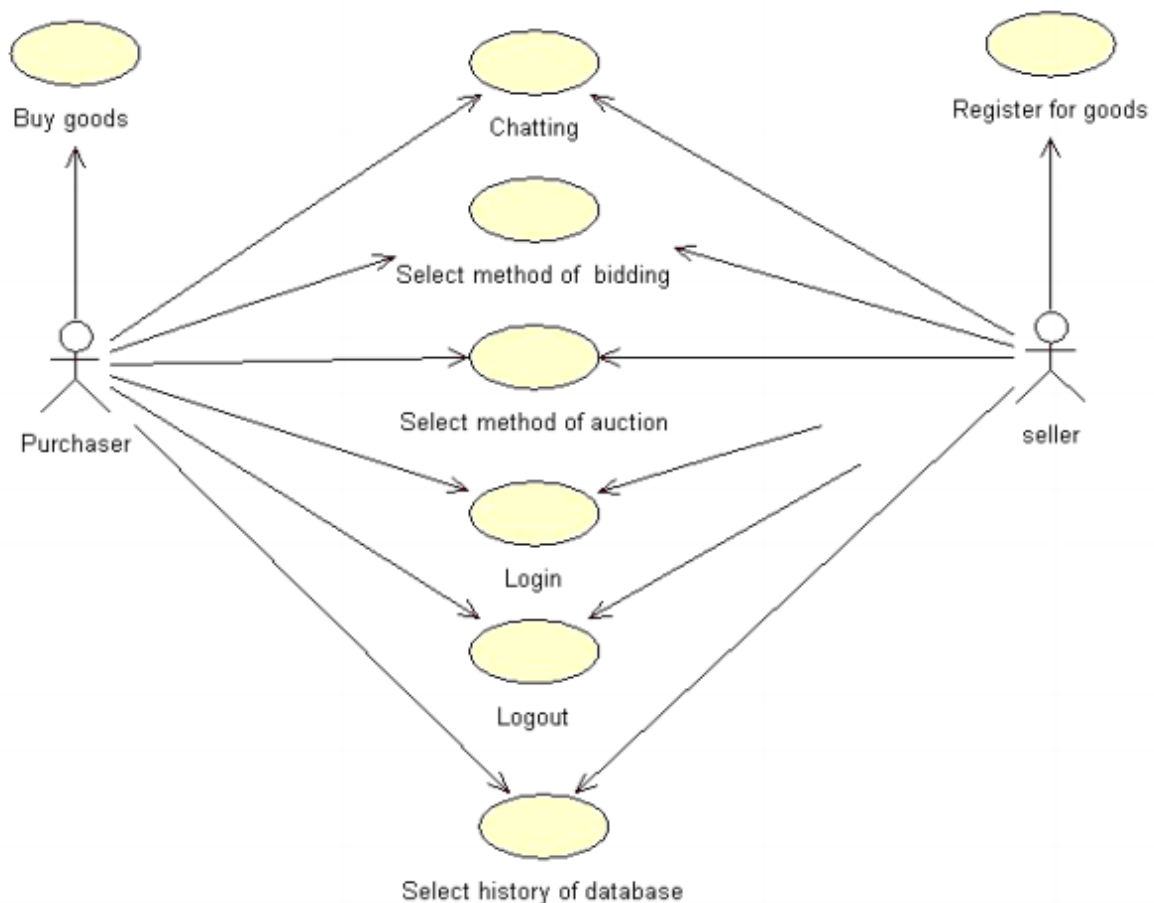


Рисунок 2.10 - Діаграма «Використання справи»

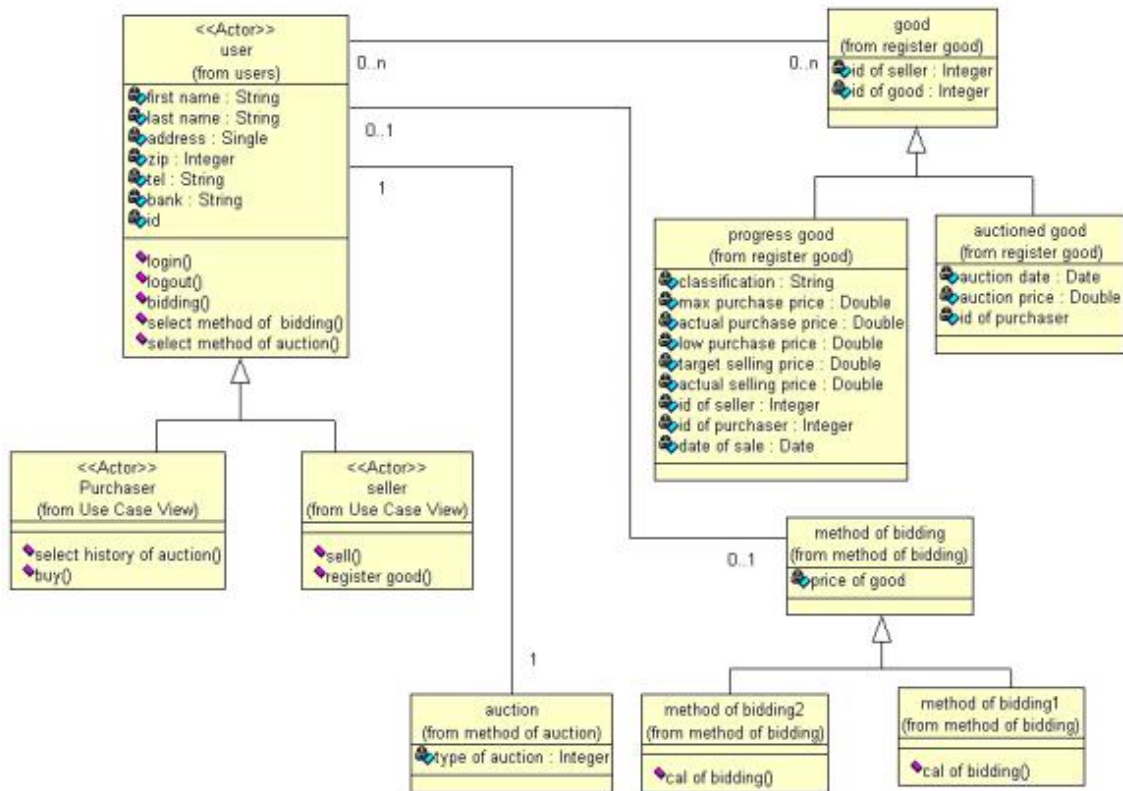


Рисунок 2.12 - Діаграма класу для системи аукціону.

Абстрактний клас (наприклад, клас користувача) абстрагує загальні характеристики (атрибут, операція, ім'я, адреса, номер телефону тощо) про актора. Ми використовуємо Бетонні класи (Покупець, Продавець), наслідуючи атрибути та операції від користувача абстрактного класу.

Діаграма послідовності відображає загальний потік керування в об'єктно-орієнтованій програмі. Як правило, це фіксує поведінку одного випадку використання.

У цій реалізації було створено ті організації, які використовуються на аукціоні як компоненти, що використовують EJB (Enterprise Java Beans). З діаграми ясно, що можна змінювати відповідні частини / компоненти, не впливаючи на всю систему.

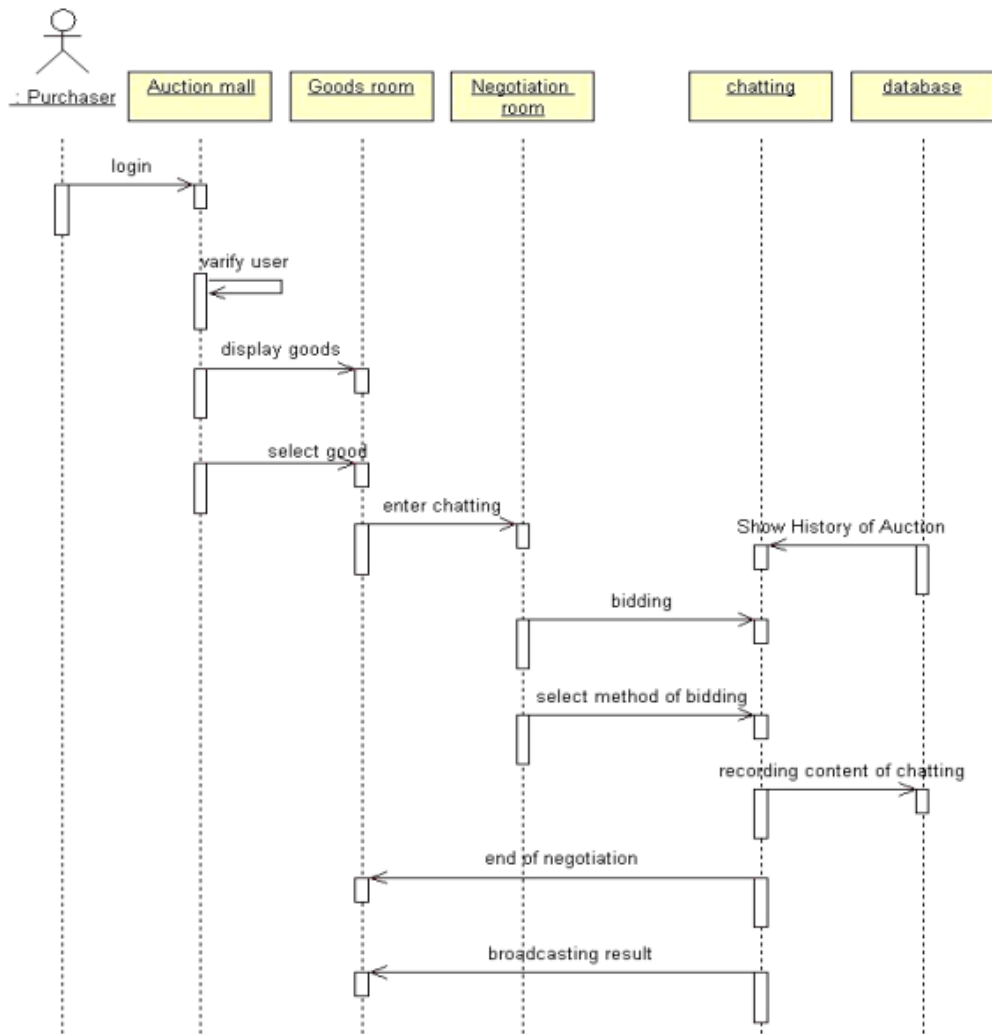


Рисунок 2.13 - Діаграма послідовності.

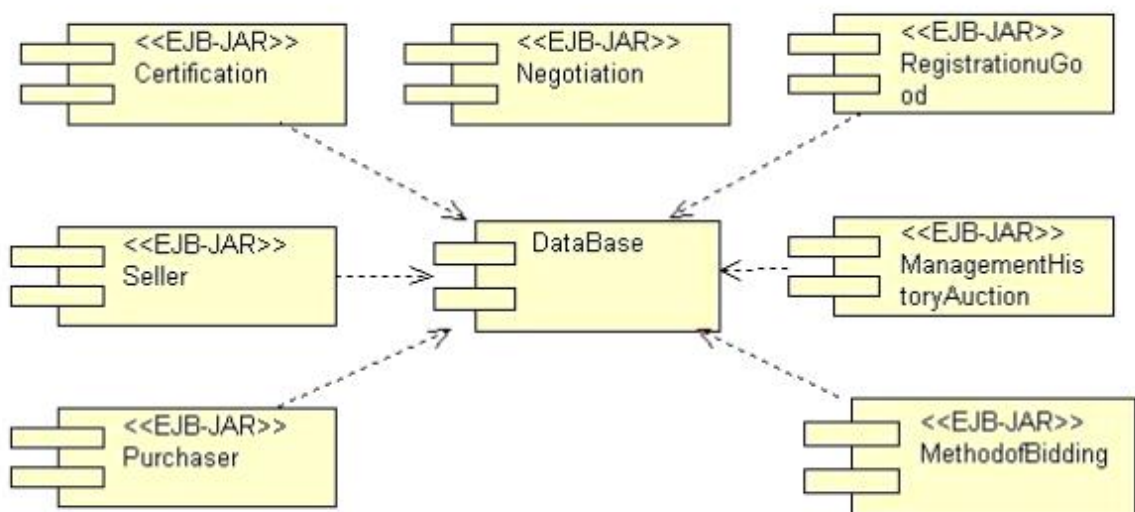


Рисунок 2.14 - Інтерактивна взаємодія між програмними компонентами системи аукціону

- Компонент сертифікації використовується для перевірки користувача, який намагається ввійти в систему.
- Продавець вводить продукти в систему за допомогою
- Реєстрація Good компонента. У цей час продавець вводить дату і час закінчення аукціону, включаючи початкові та кінцеві ціни продуктів.
- Покупець та постачальник керують інформацією, пов'язаною з аукціонами покупця та продавця, а також їх приватною інформацією.
- Компонент переговорів керує аукціоном. Якщо учасник торгів прибув на момент проведення аукціону, то закриття або якщо покупець, який пропонує найвищу ціну існує, то аукціон буде закритий. Коли аукціон закривається, запис даних аукціону переходить на компонент «Управління історією».
- Компонент ManagementAustoryAuction показує попередній аукціонний аукціон аукціоніста, який проводить поточний аукціон.
- Компонент DataBase зберігає релевантні дані про поточний аукціон (наприклад, ціну продуктів і вмісту) окремо в базі даних.
- Згідно з трьома способами встановлення ставок (Speed, Medium, Leisure), покупець вирішує наступну ставку після підтвердження кінцевої ціни, запропонованої далеко від компонента DataBase, використовуючи компонент MethodofBidding.

2.3.2 Компонентне програмування

Компонентне програмування дозволяє швидко розгортати програмне забезпечення, шляхом повторного використання збірних компонентів, які є незалежними виконуваними одиницями. Індивідуальні компоненти можуть бути виконані на замовлення, щоб відповідати новим вимогам і можуть бути перероблені в різних композиціях. Багаторазові можливості та підтримка є двома основними перевагами компонентного програмування.

Компоненти - це універсальні функціональні можливості, що дозволяють розробникам концептувати програмне забезпечення як взаємозв'язувані блоки. Темі досліджень у цій галузі включають в себе розробку структур інтеграції компонентів для призначення архітектури, яка дозволяє гнучкий склад сторонніх компонентів у програмах, повторне використання веб-дизайну по відношенню до повторного використання коду та багаторазовість компонентів для якісного програмного забезпечення.

Було впроваджено аукціонну систему, використовуючи компонентне програмне забезпечення для простого обслуговування, а також зручне повторне використання цих компонентів.

2.4 Короткий опис засобів розробки

2.4.1 Редактор коду

Редактор коду Sublime Text 3 має ряд корисних інструментів для розробки:

1) Виділення стовпців і множинна правка - виділення стовпців цілком або розстановка кілька покажчиків по тексту, що робить можливим миттєву правку. Покажчики поводяться, ніби кожен з них - єдиний 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 / кінець рядка і т. Д., Впливає на всі покажчики незалежно і відразу, дозволяючи правити складний за своєю структурою текст швидко, без використання макрокоманд або регулярних виразів.

2) Автодоповнення - 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.

3) Підсвічування синтаксису і висока контрастність - темний фон Sublime Text призначений для збільшення контрастності тексту. Основні елементи синтаксису виділені різними кольорами, які краще поєднуються з темним тлом, ніж зі світлим.

4) Підтримка систем збірки - Sublime Text дозволяє користувачеві збирати програми і запускати їх без необхідності перемикатися на командний рядок. Користувач також може налаштувати свою систему збирання та включити автоматичну збірку програми кожного разу при збереженні коду.

5) Заготовки (сніппети) - збереження фрагментів часто використовуваного коду, ключові слова для їх запуску.

6) Перехід по файлах - навігаційний інструмент, який дозволяє користувачам переміщатися між файлами, а також всередині них, за допомогою нечіткого пошуку.

7) Реалізована функція автоматичного збереження, яка допомагає користувачам не втратити виконану роботу.

8) Настроюються комбінації клавіш і інструмент навігації дозволяють призначити свої комбінації клавіш для меню і панелей інструментів.

9) Є можливість автоматизації за допомогою макросів і повтору останніх дій.

10) Команди редагування, включаючи редагування відступів, форматування параграфів і об'єднання рядків.

2.4.2 Локальний веб-сервер

Open Server Panel - це портативна серверна платформа і програмне середовище, створена спеціально для веб-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 веб-проектів, а так само для надання веб-сервісів в локальних мережах.

Склад програмного комплексу OSPanel 5.2.8;

- Apache 2.2.34 / 2.4.29;
- Nginx 1.11.7 / 1.12.2;
- MySQL 5.1.73 / 5.5.58 / 5.6.38 / 5.7.20;
- FTP FileZilla 0.9.59;
- PHPMyAdmin 4.7.3;
- PHP 7.2.00; (Xdebug 2.5.5, PDFlib 9.1.1p3, MongoDB 1.3.4, Redis 3.1.5)

2.4.3 MySQL-клієнт

MySQL-клієнт: HeidiSQL має ряд можливостей:

- Підключення до декількох серверів в одному вікні
- Підключення до серверів за допомогою командного рядка
- Підключення через тунель SSH, SSL
- Створення та редагування таблиць, уявлень, збережених процедур, тригерів і заплановані події.
- Створення дамів SQL
- Експорт з одного сервера або бази даних безпосередньо на інший сервер або база даних
- Управління користувачами і привілеями
- Імпорт текстових файлів

- Експорт в рядкові таблиці: CSV, HTML, XML, SQL, LaTeX, вікі-розмітки та PHP масиви
- Перегляд і редагування таблиць, даних за допомогою зручної сітки
- Масова зміна таблиць (зміни типу, сортування і т.д.)
- Пакетна вставка ASCII або бінарних файлів в таблиці
- Онлайн редактор запитів з підсвічуванням синтаксису і автозавершенням коду
- Моніторинг та закриття клієнтських-процесів

2.4.4 Система контролю версій

Система контролю версій: Git і візуальна оболонка SmartGit

Git підтримує швидкий поділ і злиття версій, включає інструменти для візуалізації та навігації по нелінійної історії розробки. Git надає кожному розробнику локальну копію всієї історії розробки, зміни копіюються з одного сховища в інше.

3 ПРАКТИЧНА РЕАЛІЗАЦІЯ ПРОЕКТУ

3.1 Розробка структури бази даних MySQL

Першим етапом розробки проекту є проектування і створення бази даних. Основні таблиці, які будуть використовуватися в базі даних проекту:

`admin` - Зберігає дані про всіх користувачів, які мають доступ до адміністративної панелі.

`advert` - Для розділу оголошень зберігає безпосередньо текст оголошень, їх `id`.

`attribute` - Зберігає характеристики товару - вага, карати, кількість одиниць. Використовується для фільтрації пошуку. Для заповнення прайс-листів, картки товару.

`auction` - Зберігаються аукціони. Дата початку і закінчення, `id` товару, `id` магазину або продавця.

`auction_bet` - Зберігаються зроблені на конкретний аукціон ставки.

`cart` - Кошик, містить інформацію про замовлення, дату, час.

`cart_item` - Містить інформацію про конкретні замовлення. Ціна, продавець, `id` замовлення.

`company` - Містить інформацію, картку товару, інформація про продавця, `id` компанії.

`company_abbr` - Абревіатура компанії для юридичних осіб.

`country` - Список країн.

`delivery` - Список способів доставки.

`delivery_settings` - Містить умови доставки, строки вартість.

`file` - Містить інформацію про завантажуються користувачами файлах, логотип компанії, Рисунок товарів.

`message` - Налаштування листів, що відправляються користувачами. Текст листа, список подій за якими відправляється лист, налаштовується в адміністративній панелі.

`news` - Зберігає безпосередньо текст новини, їх `id`, `id` автора.

`order` - Інформацію про замовлення, на етапі замовлення, до оформлення замовлення сам замовлення лежить в `cart_item`, після переходять сюди і в `order_item`.

`payment` - Список платіжних систем.

`payment_settings` - Налаштування для продавця які платежі він приймає.

`product` - картка товару.

product_group_size - id рубрики, до якої прив'язана картка товару за розмірами.

product_price - Прив'язка id товару + ціна, зберігає в собі кілька цін, роздрібна, оптова, спеціальна, знижки.

property - Характеристики товару. Вага, довжина, розмір.

property_value - Значення характеристики. Наприклад, розмір кільця: 16.

service - Інформація про послуги, як про товари.

user - Інформація про зареєстрованих користувачів.

Имя	Срок	Размер	Создано	Обновлено	Движок	Комментарий	Тип
admin	1	48.0 KB	2016-12-13 10:43:04		InnoDB	Администратор	Table
advert	5	96.0 KB	2016-12-13 10:43:04		InnoDB	Объявления	Table
article	3	32.0 KB	2016-12-13 10:43:04		InnoDB	Статьи	Table
attribute	8	16.0 KB	2016-12-13 10:43:04		InnoDB	Атрибуты	Table
attribute_ctg_value_advert	13	64.0 KB	2016-12-13 10:43:05		InnoDB	Рубрикатор объявлений	Table
attribute_ctg_value_article	3	64.0 KB	2016-12-13 10:43:06		InnoDB	Рубрикатор новостей	Table
attribute_ctg_value_city	2	16.0 KB	2016-12-13 10:43:06		InnoDB	Города	Table
attribute_ctg_value_company	35	64.0 KB	2016-12-13 10:43:06		InnoDB	Рубрикатор Б2В	Table
attribute_ctg_value_news	1	64.0 KB	2016-12-13 10:43:06		InnoDB	Рубрикатор новостей	Table
attribute_ctg_value_product	34	64.0 KB	2016-12-13 10:43:06		InnoDB	Рубрикатор товаров	Table
attribute_ctg_value_service	24	64.0 KB	2016-12-13 10:43:07		InnoDB	Рубрикатор услуг	Table
attribute_tag_value	0	32.0 KB	2016-12-13 10:43:08		InnoDB	Теги	Table
auction	11	96.0 KB	2016-12-13 10:43:08		InnoDB	Аукцион	Table
auction_bet	4	48.0 KB	2016-12-13 10:43:43		InnoDB	Ставки аукциона	Table
big_task	28	16.0 KB	2016-12-13 10:43:43		InnoDB	Тени	Table
cart	1	48.0 KB	2016-12-13 10:43:43		InnoDB	Корзина	Table
cart_item	5	64.0 KB	2016-12-13 10:43:43		InnoDB	Корзина	Table
company	6	80.0 KB	2016-12-13 10:43:43		InnoDB	Компании	Table
company_addr	2	16.0 KB	2016-12-13 10:43:43		InnoDB	Формы деятельности компаний	Table
company_attribute_link	2	32.0 KB	2016-12-13 10:43:43		InnoDB	Компании	Table
company_review	3	48.0 KB	2016-12-13 10:43:44		InnoDB	Компании	Table
country	2	16.0 KB	2016-12-13 10:43:44		InnoDB	Table	Table
delivery	2	16.0 KB	2016-12-13 10:43:44		InnoDB	Table	Table
delivery_settings	2	48.0 KB	2016-12-13 10:43:44		InnoDB	Список действий	Table
file	46	32.0 KB	2016-12-13 10:43:44		InnoDB	Настройка доставки	Table
file_object_link	45	32.0 KB	2016-12-13 10:43:44		InnoDB	Table	Table
file_tag_link	27	32.0 KB	2016-12-13 10:43:44		InnoDB	Table	Table
gallery	18	16.0 KB	2016-12-13 10:43:44		InnoDB	Table	Table
group	1	16.0 KB	2016-12-13 10:43:44		InnoDB	Table	Table
group_attribute	1	32.0 KB	2016-12-13 10:43:45		InnoDB	Table	Table
import_processing	6	32.0 KB	2016-12-13 10:43:45		InnoDB	Table	Table
import_processing_info	0	48.0 KB	2016-12-13 10:43:45		InnoDB	Table	Table
message	23	48.0 KB	2016-12-13 10:43:45		InnoDB	Table	Table
message_channel	3	32.0 KB	2016-12-13 10:43:45		InnoDB	Table	Table
message_channel_rule	42	64.0 KB	2016-12-13 10:43:45		InnoDB	Table	Table
message_instance	14	96.0 KB	2016-12-13 10:43:45		InnoDB	Table	Table
message_gltcholder	83	48.0 KB	2016-12-13 10:43:45		InnoDB	Table	Table
message_link	23	32.0 KB	2016-12-13 10:43:45		InnoDB	Table	Table
message_task_report	6	32.0 KB	2016-12-13 10:43:45		InnoDB	Table	Table
message_translate	27	64.0 KB	2016-12-13 10:43:46		InnoDB	Table	Table
migration	26	31.1 KB	2016-12-13 10:43:46	2016-12-13 10:43:46	MyISAM	Table	Table
news	4	32.0 KB	2016-12-13 10:43:46		InnoDB	Table	Table
offer_column_description	n	91.9 KB	2016-12-13 10:43:46		InnoDB	статусы	Table

Рисунок 3.1 - Структура бази даних

Таблиця auction_bet відповідає безпосередньо за зберігання даних про товари, виставлених на аукціон і має стовпці

id

auction_id - Ідентифікаційний номер аукціону

user_id - Ідентифікаційний номер користувача, який створив аукціон

price - Первісна вартість, оголошена на торги

status - поточний статус аукціону

create_utime - Дата додавання товару на аукціон

update_utime - Дата поновлення товару

3.2 Підготовка бази даних

Для початку, створюємо базу даних MySQL, в якій будемо зберігати і отримувати з неї дані.

```

CREATE TABLE `country` (
  `code` CHAR(2) NOT NULL PRIMARY KEY,
  `name` CHAR(52) NOT NULL,
  `population` INT(11) NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

INSERT INTO `country` VALUES ('AU','Australia',24016400);
INSERT INTO `country` VALUES ('BR','Brazil',205722000);
INSERT INTO `country` VALUES ('CA','Canada',35985751);
INSERT INTO `country` VALUES ('CN','China',1375210000);
INSERT INTO `country` VALUES ('DE','Germany',81459000);
INSERT INTO `country` VALUES ('FR','France',64513242);
INSERT INTO `country` VALUES ('GB','United Kingdom',65097000);
INSERT INTO `country` VALUES ('IN','India',1285400000);
INSERT INTO `country` VALUES ('RU','Russia',146519759);
INSERT INTO `country` VALUES ('US','United States',322976000);

```

3.3 Налаштування підключення до БД

Необхідно переконатись, що встановлені PHP-розширення PDO і драйвер PDO для використовуваної бази даних. Це базова вимога в разі використання додатком реляційної бази даних. Після того, як вони встановлені, в файлі `config / db.php` змінюються параметри на вірні для використовуваної бази даних. За замовчуванням цей файл містить наступне:

```

<? Php
return [
    'Class' => 'yii \ db \ Connection',
    'Dsn' => 'mysql: host = localhost; dbname = yii2basic',
    'Username' => 'root',
    'Password' => '',
    'Charset' => 'utf8',
];

```

Файл `config / db.php` - типовий конфігураційний інструмент, який базується на файлах. Даний конфігураційний файл визначає параметри, необхідні для створення і ініціалізації примірника `yii \ db \ Connection`, через який можливо робити SQL-запити до створеної бази даних.

Підключення до бази даних, налаштоване вище, доступно в коді програми через вираз `Yii::$app->db`.

3.4 Створення нащадка `Active Record`

Щоб представляти і отримувати дані з таблиці `country`, створюється клас - нащадок `Active Record`, під назвою `Country`, і зберігається в файлі `models / Country.php`.

```
<? Php
namespace app \ models;
use yii \ db \ ActiveRecord;
class Country extends ActiveRecord
?>
```

Клас `Country` успадковується від `yii \ db \ ActiveRecord`.

З кодом, наведеним вище, `Yii` зв'яже ім'я таблиці з ім'ям класу.

Використовуючи клас `Country`, можна легко маніпулювати даними в таблиці `country`, як показано в цих фрагментах:

```
use app \ models \ Country;
// отримуємо всі рядки з таблиці "country" і сортуємо їх по "name"
$ Countries = Country :: find () -> orderBy ( 'name') -> all ();
// отримуємо рядок з первинним ключем "UA"
$ Country = Country :: findOne ( 'UA');
// відобразить "Ukraine"
echo $ country-> name;
// міняємо ім'я країни на "Україна" і зберігаємо в базу даних
$ Country-> name = 'Україна';
$ Country-> save ();
```

3.5 Створення `Action`

Для того, щоб показати дані по країнам кінцевим користувачам, необхідно створити новий `action`. Замість розміщення нового `action`'а в контролері `site`, матиме більше сенсу створити новий контролер спеціально для всіх дій, що відносяться до даних по країнам. Називаємо новий контролер `CountryController`, і створюємо `action index` всередині нього, як показано наижче.

Зберігаємо код у файлі `controllers / CountryController.php`.

Action index викликає Country :: find (). Даний метод Active Record буде запит до БД і витягує всі дані з таблиці country. Щоб обмежити кількість країн, що повертаються кожним запитом, запит розбивається на сторінки за допомогою об'єкта yii \ data \ Pagination. Об'єкт Pagination служить двом цілям:

Встановлює пункти offset і limit для SQL інструкції, представленої запитом, щоб вона повертала тільки одну сторінку даних за раз (в нашому випадку максимум 5 рядків на сторінку).

```
<?php

namespace app\controllers;

use yii\web\Controller;
use yii\data\Pagination;
use app\models\Country;

class CountryController extends Controller
{
    public function actionIndex()
    {
        $query = Country::find();

        $pagination = new Pagination([
            'defaultPageSize' => 5,
            'totalCount' => $query->count(),
        ]);

        $countries = $query->orderBy('name')
            ->offset($pagination->offset)
            ->limit($pagination->limit)
            ->all();

        return $this->render('index', [
            'countries' => $countries,
            'pagination' => $pagination,
        ]);
    }
}
```

Він використовується у view для відображення пагінатора, що складається з набору кнопок з номерами сторінок, це буде роз'яснено в наступному підрозділі.

В кінці коду action index виводить view з ім'ям index, і передає в нього дані по країнам разом с інформацією про пагінацію.

3.6 Створення View

Створюється піддиректорія з ім'ям country всередині директорії views. Ця папка буде використовуватися для зберігання всіх view, виведених контролером country. Всередині директорії views / country створюється файл з ім'ям index.php, що містить наступний код:

```
<?php
use yii\helpers\Html;
use yii\widgets\LinkPager;
?>
<h1>Countries</h1>
<ul>
<?php foreach ($countries as $country): ?>
    <li>
        <? = Html::encode("{ $country->name} ( { $country->code} )" ) ?>:
        <? = $country->population ?>
    </li>
<?php endforeach; ?>
</ul>

<? = LinkPager::widget(['pagination' => $pagination]) ?>
```

View має дві частини щодо відображення даних по країнам. У першій частині надані дані по країнам виводяться як невпорядкований HTML-список. У другій частині виводиться віджет yii \ widgets \ LinkPager, використовуючи інформацію про пагінацію, передану з action у view. Віджет LinkPager відображає набір посторінкових кнопок. Клік по кожній із них оновить дані по країнам у відповідній сторінці.

3.7 Перевірка працездатності

Щоб побачити, як працює весь вищенаведений код, переходимо за наступним посиланням в браузері:

```
http://torgold.loc/index.php? r = country% 2Findex
```

На початку відображається сторінка, що показує п'ять країн. Під країнами присутній пагігатор з чотирма кнопками. Якщо клікнути на кнопку "2", здійснюється перехід на сторінку, яка буде показувати інші п'ять країн з бази даних: друга сторінка записів. Також, звертаємо увагу на те, що URL в браузері теж змінилося на

```
http://torgold.loc/index.php? r = country% 2Findex & page = 2
```

За кадром, `Pagination` надає всю необхідну функціональність для посторінкового розбивки набору даних:

На початку `Pagination` показує першу сторінку, яка відображає `SELECT` запит країн з параметрами `LIMIT 5 OFFSET 0`. Як результат, перші п'ять країн будуть отримані і відображені.

Віджет `LinkPager` виводить кнопки сторінок використовуючи URL'и, створені `Pagination`. Ці URL'и міститимуть параметр запити `page`, який представляє різні номери сторінок.

Якщо зробити клік по кнопці "2", спрацює і буде опрацьовано новий запит для маршруту `country / index`. Таким чином новий запит країн матиме параметри `LIMIT 5 OFFSET 5` і поверне наступні п'ять країн для відображення.

3.8 Розгортання проекту на основі FrameworkYii2

3.8.1 Установка Yii2

Установка Yii з архіву складається з трьох кроків:

1. Завантажується архів з `yiiframework.com`;
2. Розпаковується скачаний архів в папку, доступну з Web.
3. У файлі `config / web.php` додається секретний ключ в значення `cookieValidationKey`.

Після установки фреймворк буде доступний за наступним URL:

```
http://torgold.loc/basic/web/index.php
```

Тут мається на увазі, що фреймворк було встановлено в директорію basic в кореневій директорії веб-сервера і сервер працює локально (localhost).

Для коректної роботи фреймворка необхідна установка PHP, відповідна його мінімальним вимогам. Основна вимога - PHP версії 5.4 і вище. Якщо фреймворк працює з базою даних, необхідно встановити розширення PHP PDO і відповідний драйвер (наприклад, pdo_mysql для MySQL).

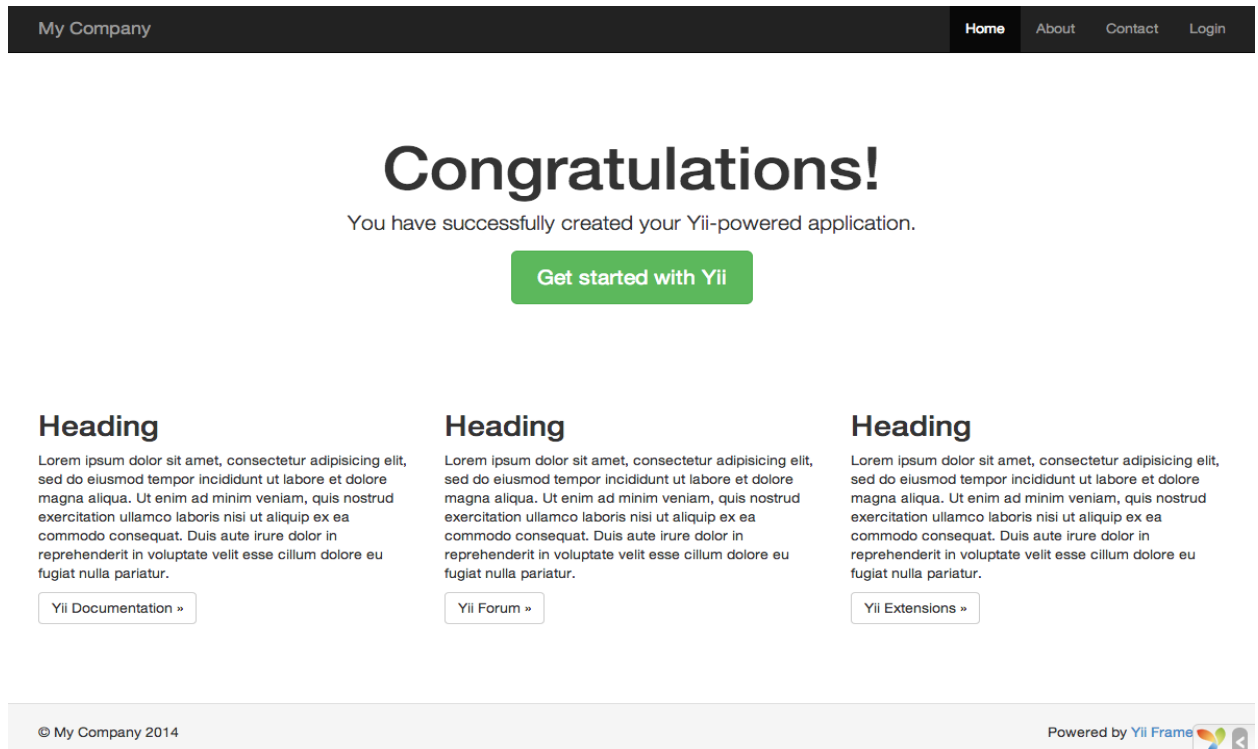


Рисунок 3.2 – Перша сторінка розгорнутого проекту

3.8.2 Налаштування веб сервера

Фреймворк, встановлений за інструкціями, буде працювати відразу як з Apache, так і з Nginx під Windows і Linux з встановленим PHP 5.4 і вище. Yii 2.0 також сумісний з HHVM. Проте, в деяких випадках поведінка при роботі з HHVM відрізняється від звичайного PHP.

Щоб змінити вигляд URL фреймворку на робочому сервері з

`http://www.torgold.loc/basic/web/index.php` на більш зручний `http://www.torgold.loc/index.php` необхідно змінити кореневу директорію в налаштуваннях веб-сервера так, щоб та вказувала на basic / web.

3.8.3 Рекомендовані настройки Apache

Додаємо наступні рядки в файл `httpd.conf` Apache в конфігураційний файл віртуального хоста. Замінюємо `path/to/basic/web` на коректний шлях до `basic / web`.

```
# Встановлюємо кореневої директорією "basic / web"
DocumentRoot "path / to / basic / web"
<Directory "path / to / basic / web">
    RewriteEngine on
    # Якщо запитувана в URL директорія або файл існують звертаємося до
них безпосередньо
    RewriteCond% {REQUEST_FILENAME}! -f
    RewriteCond% {REQUEST_FILENAME}! -d
    # Якщо немає - перенаправляємо запит на index.php
    RewriteRule. index.php
</ Directory>
```

3.8.4 Рекомендовані настройки Nginx

```
server {
    charset utf-8;
    client_max_body_size 128M;
    listen 80; ## listen for ipv4
    #listen [::]: 80 default_server ipv6only = on; ## слухаємо ipv6
    server_name mysite.test;
    root / path / to / basic / web;
    index index.php;
    access_log /path/to/project/log/access.log;
    error_log /path/to/project/log/error.log;
    location / {
        # Перенаправляти всі запити до неіснуючих тек і файлів на index.php
        try_files $ uri $ uri / /index.php?$args;
    }

    # Розкоментуйте рядки нижче, щоб уникнути обробки Yii звернень до
неіснуючих статичних файлів
    #location ~ \. (js | css | png | jpg | gif | swf | ico | pdf | mov |
fla | zip | rar) $ {
        # Try_files $ uri = 404;
    #}
    #error_page 404 /404.html;
    location ~ \ .php $ {
        include fastcgi.conf;
        fastcgi_pass 127.0.0.1:9000;
```

```

        #fastcgi_pass unix: /var/run/php5-fpm.sock;
    }
    location ~ /\.(ht|svn|git) {
        deny all;
    }
}

```

Використовуючи дану конфігурацію встановлюємо `cgi.fix_pathinfo = 0` в `php.ini` щоб запобігти зайвим системним викликам `stat ()`.

Використовуючи HTTPS необхідно ставити `fastcgi_param HTTPS on`; щоб Yii міг коректно визначати захищене з'єднання.

3.8.5 Запуск програми

Після установки Yii він буде доступний або по URL `http://torgold.loc/basic/web/index.php`, або по `http://torgold.loc/index.php`, в залежності від налаштувань Web-сервера.

3.9 Функціональність

Встановлений шаблон простого додатка складається з чотирьох сторінок: домашня сторінка, відображається при переході по URL

`http://torgold.loc/index.php`

Сторінка "About" ("Про нас") - містить загальну інформацію про розробника

На сторінці "Contact" знаходиться форма зворотного зв'язку, на якій користувач може звернутися до розробника по e-mail.

На сторінці "Login" відображається форма авторизації.

Ці сторінки використовують суміжний хедер (шапка сайту) і футер (підвал). В "шапці" знаходиться головне меню, за допомогою якого користувач переміщається по сайту. В "підвалі" - копірайт і загальна інформація.

У самій нижній частині вікна будуть відображатися системні повідомлення Yii - журнал, зневадження, повідомлення про помилки, запити до бази даних та інше. Висновком цієї інформації керує вбудований відладчик, він записує і відображає інформацію про хід виконання програми.

На додаток до фреймворку є консольний скрипт з назвою Yii, який знаходиться в базовій директорії фреймворку. Цей скрипт може бути використаний для виконання фонових завдань і послуг у програмі.

4 РОЗРОБКА І ВПРОВАДЖЕННЯ ФУНКЦІОНАЛУ

4.1 Реєстрація

При натисканні на посилання "Реєстрація" користувач потрапляє на сторінку registration.php, на якій буде перебувати форма для введення всіх необхідних даних при реєстрації. Вихідний код процедури реєстрації містить наступний HTML код:

```
<form action = "verification.php" method = "POST">
<table>
<tr>
<td> Логін <font color = "red"> * </ font>:</ td>
<td><input type = "text" size = "20" name = "login"></ td>
</tTr>
<tr>
<td> Пароль <font color = "red"> * </ font>:</ td>
<td><input type = "password" size = "20" maxlength = "20" name =
"password"></ td>
</tr>
<tr>
<td> Підтвердження пароля <font color = "red"> * </ font>:</ td>
<td><input type = "password" size = "20" maxlength = "20" name =
"password2"></td>
</tr>
<tr>
<td> Е-mail <font color = "red"> * </ font>:</ td>
<td><input type = "text" size = "20" name = "email"></ td>
</tr>
<tr>
<td> Ім'я: </ td>
<td><input type = "text" size = "20" name = "name"></ td>
</tr>
<tr>
<td> Прізвище: </ td>
<td><input type = "text" size = "20" name = "lastname"></ td>
</tr>
<tr>
<td></ td>
```

```

        <td colspan = "2"><input type = "submit" value = "Зареєструватися" name
= "submit">
    </td>
</tr>
</table>
</form>

<br> Поля із позначкою <font color = "red"> * </ font> обов'язкові для
заповнення.

```

У цій формі користувач вводить логін, пароль, e-mail і при бажанні ім'я і прізвище, після чого всі ці дані відправляються оброблювачу verification.php. У ньому буде відбуватися перевірка даних і в разі успіху дані заносяться в базу даних. Після цього користувач буде вважатися зареєстрованим. Розглянемо, як влаштований обробник verification.php

```

<?php
include_once ( "bd.php");
if (isset ( $ _ POST [ 'submit'])) {
    if (empty ( $ _ POST [ 'login'])) {
        echo '<br><font color = "red"><img border = "0" src = "error.gif"
alt = "Введіть логін"> Введіть логін! </ font>';
    }
    elseif (! preg_match ( "/ ^ \ w {3} $ /", $ _ POST [ 'login'])) {
        echo '<br><font color = "red"><img border = "0" src = "error.gif" alt =
"У поле" Логін "введені неприпустимі символи!">В поле "Логін" введені
неприпустимі символи! Тільки букви, цифри і підкреслення! </ Font> ';
    }
    elseif (empty ( $ _ POST [ 'password'])) {
        echo '<br><font color = "red"><img border = "0" src = "error.gif" alt =
"Введіть пароль!"> Введіть пароль! </ font>';
    }
    elseif (! preg_match ( "/ \ A ( \ w) {6,20} \ Z /", $ _ POST [
'password'])) {
        echo '<br><font color = "red"><img border = "0" src = "error.gif" alt =
"Пароль занадто короткий!">Пароль занадто короткий! Пароль повинен бути не
менше 6 символів! </ Font> ';
    }
    elseif (empty ( $ _ POST [ 'password2'])) {
        echo '<br><fontcolor = "red"><imgborder = "0" src = "error.gif" alt =
"Введіть підтвердження пароля!"> Введіть підтвердження пароля! </ font>';
    }
}

```



```

elseif ($_POST ['password'] != $_POST ['password2']) {
    echo '<br><fontcolor = "red"><imgborder = "0" src = "error.gif" alt =
"Введені паролі не збігаються!"> Введені паролі не збігаються! </ font>';
}
elseif (empty ($_POST ['email'])) {
    echo '<br><font color = "red"><img border = "0" src = "error.gif" alt =
"Введіть E-mail!"> Введіть E-mail! </ Font> ';
}
elseif (! preg_match ( "/ ^ [a-zA-Z0-9 _ \. \ -] + @ ([a-zA-Z0-9 \ -] +
\.) + [a-zA-Z] {2, 6} $ / ", $_POST ['email'])) {
    echo '<br><font color = "red"><img border = "0" src = "error.gif" alt =
"E-mail має неприпустимо формат!"> E-mail має неприпустимо формат! Наприклад,
name@gmail.com! </ Font> ';
}
else {
    $ Login = $_POST ['login'];
    $ Password = $_POST ['password'];
    $ MdPassword = md5 ($ password);
    $ Password2 = $_POST ['password2'];
    $ Email = $_POST ['email'];
    $ Rdate = date ( "d-m-Y в H: i");
    $ Name = $_POST ['name'];
    $ Lastname = $_POST ['lastname'];
    $ Query = ( "SELECT id FROM users WHERE login = '$ login'");
    $ Sql = mysql_query ($ query) or die (mysql_error ());
    if (mysql_num_rows ($ sql) > 0) {
        echo '<font color = "red"><img border = "0" src = "error.gif" alt =
"Вибачте, такий користувач зареєстрований!"> Вибачте, такий користувач
зареєстрований! </ font>';
    }
    else {
        $ Query2 = ( "SELECT id FROM users WHERE email = '$ email'");
        $ Sql = mysql_query ($ query2) or die (mysql_error ());
        if (mysql_num_rows ($ sql) > 0) {
            echo '<font color = "red"><img border = "0" src = "error.gif" alt =
"Користувач з таким e-mail зареєстрований!"> Користувач з таким e-mail вже
зареєстрований! </ font> ';
        }
        else {
            $ Query = "INSERT INTO users (login, password, email, reg_date,
name_user, lastname)

```

```

VALUES ( '$ login', '$ mdPassword', '$ email', '$ rdate', '$ name', '$
lastname') ";
$ Result = mysql_query ($ query) or die (mysql_error ()) ;;
echo '<font color = "green"><img border = "0" src = "ok.gif" alt = "Ви
успішно зареєструвалися!"> Ви успішно зареєструвалися! </ font><br><a href =
"index .php "> На головну </a> ' ;
}
}
}
}
?>

```

На самому початку підключається файл `bd.php`, що містить дані про підключення до бази даних.

```

mysql_connect ("localhost", "user", "password");
mysql_select_db ("reg");

```

```

CREATE TABLE `users` (
  `id` INT NOT NULL AUTO_INCREMENT ,
  `login` VARCHAR( 20 ) NOT NULL ,
  `password` VARCHAR( 35 ) NOT NULL ,
  `email` VARCHAR( 50 ) NOT NULL ,
  `reg_date` VARCHAR( 32 ) NOT NULL ,
  `name_user` VARCHAR( 32 ) NOT NULL ,
  `lastname` VARCHAR( 32 ) NOT NULL ,
  PRIMARY KEY ( `id` )
);

```

Далі здійснюється перевірка введених даних, якщо вони не коректні, то видається помилка. Якщо все проходить вдало, то перетворюємо глобальні змінні в звичайні, а так само шифруємо пароль. Потім йде перевірка, чи немає у нас користувача з таким же логіном і e-mail, якщо є, то виводиться помилка. Якщо ці перевірки були пройдені успішно, заносимо нового користувача в базу даних і повідомляємо про успішну реєстрацію.

Тепер, коли користувач зареєструвався йому потрібно увійти на сайт. При натисканні кнопки входу відправляються дані файлу `login.php`, який перевіряє чи є такий користувач:

```

<?php
if (isset ($ _ POST [ 'login'])) {
    $ Login = $ _POST [ 'login'];
    if ($ login == '') {

```

```

        unset ($ login);
        exit ( "Будь ласка введіть логін!");
    }
}
if (isset ($ _ POST [ 'password'])) {
    $ Password = $ _POST [ 'password'];
    if ($ password == '') {
        unset ($ password);
        exit ( "Введіть пароль");
    }
}
$ Login = stripslashes ($ login);
$ Login = htmlspecialchars ($ login);
$ Password = stripslashes ($ password);
$ Password = htmlspecialchars ($ password);

$ Login = trim ($ login);
$ Password = trim ($ password);
$ Password = md5 ($ password); // шифруємо пароль
$ User = mysql_query ( "SELECT id FROM users WHERE login = '$ login' AND
password = '$ password'");
$ Id_user = mysql_fetch_array ($ user);
if (empty ($ id_user [ 'id'])) {
    exit ( "Вибачте, введений вами логін або пароль невірний.");
}
else {
    $ _SESSION [ 'password'] = $ password;
    $ _SESSION [ 'login'] = $ login;
    $ _SESSION [ 'id'] = $ id_user [ 'id'];
}
echo "<meta http-equiv = 'Refresh' content = '0; URL = index.php'>";
?>

```

Файл login.php перевіряє користувача на існування і запускає для нього сесію, так як весь вхід побудований саме на них. Ці змінні будуть «висіти» в браузері до тих пір поки не буде закритий (на цьому заснований вихід).

Тепер щоб сесії працювали потрібно в файлі bd.php дописати рядок session_start (); Для зручності використання змінних сесії будемо привласнювати їх звичайним змінним:

```
$ Login = $ _SESSION [ 'login'];
```

```
$ Password = $_SESSION [ 'password'];
$ Id_user = $_SESSION [ 'id'];
```

Тепер щоб увійшов користувач міг бачити зміст сайту для зареєстрованих користувачів, створюється така умова:

```
if (empty ($ login) and empty ($ password)) {
echo "Вам необхідно зареєструватися, щоб бачити цей контент";
}
else {
echo "Привіт, ". $ login. " | <a href='exit.php'> Вихід </a><br> Контент
для зареєстрованих користувачів";
}
```

Наостанок створюється файл `exit.php`, який дозволить користувачеві вийти з сайту. Все що для цього потрібно це знищити змінні в сесіях:

```
unset ($ _ SESSION [ 'password']);
unset ($ _ SESSION [ 'login']);
unset ($ _ SESSION [ 'id']);
```

4.2 Авторизація

Існує кілька способів створення реєстрації та авторизації користувачів. Насамперед необхідно:

- Створити форму реєстрації на HTML.
- Отримати дані з форми в скрипті-обробнику.
- Перевірити отримані дані, і якщо вони є некоректними, то зробити перенаправлення назад на форму реєстрації.
- Якщо дані коректні, то записати їх в базу даних.

Інформація про авторизацію повинна десь зберігатися. Найпростіший варіант - це зберігання інформації в сесії (або в cookie).

Для цього необхідно створити форму авторизації користувача на HTML, куди користувач повинен буде ввести свій логін і пароль.

В скрипті-обробнику прийняти дані від користувача. Перевірити вірність введених даних, і якщо логін і пароль співпадають з існуючим користувачем в базі даних, то запишете в cookie або сесію інформацію з логіном і шифрованим паролем.

Якщо логін і/або пароль введені невірно, то робити перенаправлення назад на форму авторизації.

Фактично, тепер потрібно цю інформацію перевіряти на кожній сторінці сайту і звіряти її аналогічно перевірці форми авторизації. Тобто зчитується з cookie(сесії) логін і пароль, і перевіряється. Якщо вони вірні, то відображається одна сторінка (для зареєстрованих користувачів), а якщо невірні, то відображається інша (для гостей).

Створюємо кнопку "Вихід". При натисканні на цю кнопку, стираються cookie, або сесія. Таким чином, користувач автоматично виходить з сайту.

4.3 Генерація коду за допомогою Gii

Gii використовується для автоматичної генерації коду, що реалізує деякі загальні функції сайту. Для досягнення цієї мети потрібно ввести необхідну інформацію відповідно до інструкцій, відображеними на веб-сторінках Gii. Gii представлений в Yii як модуль. Активується Gii, налаштуванням його у властивості modules. Залежно від того, яким чином було створено програму, є можливість перевірки наявності наступного коду в конфігураційному файлі config/web.php:

```
$ Config = [...];

if (YII_ENV_DEV) {
    $ Config [ 'bootstrap' ] [] = 'gii';
    $ Config [ 'modules' ] [ 'gii' ] = [
        'Class' => 'yii \ gii \ Module',
    ];
}
```

Наведена вище конфігурація показує, що перебуваючи в режимі розробки, програма має включати в себе модуль з ім'ям gii, який реалізує клас yii \ gii \ Module.

При огляді вхідного скрипта web / index.php фреймворку, можна відмітити наступний рядок, яка встановлює константу YII_ENV_DEV в значення true.

```
defined ( 'YII_ENV') or define ( 'YII_ENV', 'dev');
```

Завдяки цьому рядку ваше додаток знаходиться в режимі розробки, і Gii вже активований відповідно до описаної вище конфігурацією. Тепер можна отримати доступ до Gii за наступною адресою:

```

http: //torgold.loc/index.php? r = gii

'Gii' => [
    'Class' => 'yii \ gii \ Module',
    'AllowedIPs' =>[ '127.0.0.1', ':: 1', '192.168.0. *',
'192.168.178.20'] // регулюйте відповідно до своїх потреб
],

```

4.4 Генерація класу Active Record

Щоб використовувати Gii для генерації класу Active Record, потрібно вибрати "Генератор моделі" натиснувши на посилання на головній сторінці Gii і заповнити форму:

Ім'я таблиці: country

Клас моделі: Country

Потім натиснути на кнопку "Попередній перегляд". Буде видно, що в моделі models / Country.php перерахований результат створюваних файлів класів. Для перегляду вмісту класу потрібно натиснути на ім'я файлу.

Якщо вже був створений такий же файл і необхідно перезаписати його, потрібно натиснути на кнопку "Diff" поруч з ім'ям файлу, щоб побачити відмінності між генеруються кодом і існуючою версією.

Для перезапису існуючого файлу встановлюється прапорець поруч з "overwrite" і натискається кнопку "Generate". Для створення нового файлу можна просто натиснути "Generate".

Після цього виводиться підтвердження, яке вказує на те, що код був успішно згенерований. Якщо файл існував до цього, також виводиться повідомлення про те, що файл був перезаписаний новим кодом.

4.5 Створення CRUD

CRUD розшифровується як Create, Read, Update і Delete, що надає чотири основні функції, що виконуються над даними на більшості веб-сайтів. Щоб створити функціональність CRUD використовуючи Gii, вибирається "CRUD Генератор" натисканням на посилання на головній сторінці Gii. Для таблиці «country» заповнюється отримана форма наступним чином:

```

ModelClass: app \ models \ Country
Search Model Class: app \ models \ CountrySearch
Controller Class: app \ controllers \ CountryController

```

Після натискання на кнопку "Preview" відображається список файлів, які будуть створені.

Якщо файли controllers / CountryController.php і views / country / index.php вже були створені потрібно встановити прапор "overwrite", щоб замінити їх.

4.6 Фільтрація

Основне завдання фільтра - звузити видачу, скоротити обсяг релевантних результатів. На відміну від розширеного пошуку, який також дозволяє ввести уточнюючі параметри і отримати відразу точну вибірку, фільтр робить це крок за кроком і вчасно дає зрозуміти, які параметри зводять можливу видачу до нуля. Фільтрація простіша для сприйняття. Одне із завдань сайту - бути продавцем-консультантом. Люди приходять в магазин, в якому спочатку шукають кільця (категорія), потім матеріал виробу (фільтрація по параметру), а потім вже відповідні розміри (фільтрація по наступного параметра). Чим більше вибір, тим більша кількість фільтрів може бути використано для пошуку відповідного варіанту.

Фільтром потрібно користуватися після видачі, тому що поки не отримали хоч якихось результатів, ми не можемо знати, що буде після уточнення.

Можливо, в магазині всього 20 виробів, і користувачеві немає необхідності фільтрувати їх. Людський мозок влаштований так, що йому простіше розпізнати, чим згадати. Необхідно пам'ятати про базовий принцип, на якому будується робота з фільтрами на сайті - спочатку вміст, потім фільтрація. Щоб щось відфільтрувати - потрібно щось спочатку вивести.

Тому, при реалізації фільтрації, не варто послідовно показувати сторінку категорій в каталозі товарів, потім сторінки підкатегорій та інше, щоб щось в загальні вивести. Створюється загальний висновок на сторінці каталогу і надається користувачеві можливість для послідовного звуження вибірки.

Можна відразу видавати результат вже впорядкований за будь-якою ознакою. На першій сторінці відображаються найбільш популярні товари, новинки. Як показує аналітика - як мінімум з цікавості користувачі будуть активно кликати в перші пункти навіть не відсортованої видачі. За першими результатами клацають просто тому, що вони

перші. Далі, після уточнення запиту фільтром ми звужуємо видачу, залишаючи тільки підходящі варіанти.

Як добре організованої системи фільтрації можна взяти за приклад Яндекс.Маркет, де для кожної категорії товарів запропоновані свої фільтри. Однак варто врахувати, що на Маркеті представлений дуже широкий асортимент, тому велика кількість докладних фільтрів виправданий захід.

Інший приклад адаптації видачі під потребу клієнта - «консультант». Він проводить роботу за відвідувача вашого сайту, об'єднуючи параметри товару в групи, відповідні конкретному завданні. Наприклад, видача товарів може бути зроблена на основі того, для якої мети купується виріб. Ще одна важлива деталь - гарний фільтр повинен звужувати вибірку в реальному часі.

Наприклад, якщо був пересунутий повзунок, звуживши ціновий діапазон, видача повинна змінитися відразу ж. Вибрали якийсь параметр - вибірка звужується без додаткових оновлень і натискань. Звичайно ж, в хорошому фільтрі завжди є можливість скинути поточні умови.

Розглянемо введення чисел, діапазонів. Наприклад, ціни. Класичний варіант рішення - повзунок для миші і поле введення для клавіатури.

Рухається повзунок - змінюється число в поле введення, і навпаки, змінюється число - пересувається повзунок. Однак в такому варіанті дію і зворотний зв'язок відірвані одна від одної. Повзунок ми рухаємо в одному місці, а число змінюється десь поруч. Рекомендується об'єднати поле введення і повзунок. Бібліотека jQuery дозволяє швидко впровадити таке рішення.

У разі, коли ми задаємо діапазон цін, можна відразу на повзунку показувати кількість доступних варіантів. Це допомагає відразу побачити, що всі доступні варіанти знаходяться в заданому діапазоні цін.

4.7 Оброблювач прайс-листів

Оброблювач прайс-листів - це інструмент для завантаження товарів і цін в інтернет-магазин з XLS-файлів.

Додавання товарів через XLS-файли дуже зручно - за одну хвилину їх можна додати декілька тисяч. Причому, до них можуть довантажити описання, Рисунок та інші поля автоматично, так, що залишиться тільки сформувати структуру каталогу і магазин буде готовий до роботи.

Вимоги до XLS-файлу

Оброблювач прайс-листів підтримує XLS-файл формату Книга Excel з розширенням * .xls і * .xlsx.

Прайс-лист будується за принципом: один товар - один рядок. Те ж саме стосується категорій. Кожна категорія має перебувати на новому рядку. Допускаються порожні рядки, обробник буде їх ігнорувати.

Наступне правило: один стовпець - один параметр товару. Допускається пропускати осередку або стовпці у товарів, для категорій ігноруються всі осередки крім назви самої категорії.

Щоб розмістити параметр товару, необхідно помістити його в відповідному полі. При вказівці цін можна використовувати формули в Excel. При вказівці повного опису можна використовувати HTML.

4.8 Статична верстка CSS по макетах

CSS (Cascading Style Sheets - каскадні таблиці стилів) - одна з базових технологій, що дозволяє створювати зовнішній вигляд веб-сторінок зміст яких не пов'язано з дизайном. Змінюючи всього один рядок можна змінити властивість, відображення, розміри і колір для потрібного елемента в усьому проекті.

CSS-код - це список інструкцій для браузера, - як і де відобразити елементи веб-сторінки, написаний особливим чином. Під «елементами» зазвичай маються на увазі теги XHTML / HTML та їх вміст.

Основна ідея CSS в тому, щоб відокремити дизайн документа від його вмісту. CSS відповідає за оформлення і зовнішній вигляд, а XHTML / HTML - за зміст і логічну структуру документа.

Вихідний код Bootstrap включає CSS, JavaScript і шрифти, разом з вихідним Less, JavaScript . Також містить документацію.

```
bootstrap/
├─ less/
├─ js/
├─ fonts/
├─ dist/
│   ├─ css/
│   └─ js/
├─ fonts/
└─ docs/
    └─ examples/
```

Рисунок 4.1 - Структура Bootstrap

less /, js /, шрифти / з іконками. Папка dist / включає в себе всі перераховані скомпільовані файли, зазначені в розділі вище. Папка docs / містить документацію та приклади / використання Bootstrap. Крім того, знаходиться інформація про ліцензії та розвитку попередніх версій.

Згідно з наданими макетами для верстки буде використовуватися спеціальний фреймворк - Bootstrap, що включає в себе CSS і JavaScript рішення.

У Bootstrap використовуються деякі елементи HTML і властивості CSS, які вимагають використання HTML5 doctype.

Фреймворк Bootstrap 3 розрахований на роботу з мобільними пристроями. Замість того щоб додавати додаткові шаблони для мобільних пристроїв, вони вбудовані в ядро фреймворка. Макети для мобільних пристроїв зберігаються в бібліотеці, а не окремими файлами.

Щоб забезпечити належний рендеринг і зміну розміру натисканням, додається метатег viewport в тег <head>.

```
<Meta name = "viewport" content = "width = device-width, initial-scale = 1">
```

4.8.1 Використання LESS

Даний спосіб має на увазі, що всі змінні Bootstrap зберігаються в .less файлах. Розробник працює з цими змінними і в разі потреби вручну або автоматично компілює їх в CSS файли, а сам HTML підключає тільки скомпільовані CSS файли. Саме цей варіант і буде розглядатися, як самий гнучкий.

Існує велика кількість способів скомпілювати LESS файли і Bootstrap залишає це на розсуд розробника. Сам Bootstrap використовує для компіляції Grunt, також передбачена можливість використовувати плагін для продуктів JetBrains. Такими рішеннями є програми WinLess для Windows, SimpLESS для Mac або Koala для Linux.

Всі ці програми роблять приблизно одне й те саме: отримують на вхід папку з LESS файлами і слухають зміни в них. Як тільки були внесені зміни в будь-який файл - тут же він компілюється в зазначений CSS файл.

Таким чином відпадає необхідність запускати компіляцію вручну після кожної зміни. Після зміни і збереження LESS файлу відразу ж стають видні зміни на сайті в уже скомпільованому, стислому вигляді.

4.8.2 Розмітка. Система сітки.

Основна мета застосування Bootstrap - зверстати веб-проект, адаптований під різні мобільні пристрої. Адаптивність в даному випадку означає, що макет веб-документа повинен підлаштовуватися під пристрій, який його відображає - ПК, планшет, смартфон.

Візуальне оформлення окремих елементів макета веб-сторінки також включено в фреймворк.

Система розмітки використовуються для створення макетів сторінок за допомогою ряду рядків і стовпців, в яких розміщується зміст сторінки. Принцип роботи системи розмітки Bootstrap:

.row повинні бути поміщені в .container (fixed-width) або .container-fluid (full-width) для правильного вирівнювання та заповнення.

Використовувати рядки для створення горизонтальних групи стовпців.

Вміст має бути розміщено в стовпцях, і тільки колонки можуть бути безпосередніми нащадками рядків.

Стандартні класи розмітки як .row і .col-xs-4 доступні для швидкого прийняття розмітки макетів. LessMixins також може бути використаний для більш семантичних схем.

Стовпці створюють жолоби за допомогою padding. Тоді padding компенсується в рядках для першого і останнього стовпця за допомогою негативного margin на .row.

Стовпці розмітки створюються із зазначенням кількості дванадцяти доступних стовпців, які ви хочете охопити. Наприклад, три рівні колонки використовували б три .col-xs-4.

```
<head>
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap.min.css">
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap-theme.min.css">
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/js/bootstrap.min.js"></script>
</head>
```

Рисунок 4.2 - Підключення Bootstrap

Відповідно до створеної структури Yii2 Framework основні файли макета розташовуються в папці \ vendor \ vae \ themes \ frontend \ views \ layouts, а css-файли \ frontend \ web \ assets.

Для сторінок, які не вимагають зміни положення блоків, зміни структури макета для мобільних пристроїв була використана стандартна для Bootstrap система сітки.

```

<div class="row static-page">
  <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12">
    <div class="white-background"></div>
    <div class="row">
      <div class="col-lg-1 col-md-2 col-sm-2 col-xs-12">
        <a class="btn btn-default block-h4-btn-back" href="/">Назад</a>
      </div>
      <div class="col-lg-11 col-md-10 col-sm-10 col-xs-12">
        <h4 class="block-h4"><?= $this->title; ?></h4>
      </div>
      <div class="col-lg-12 col-md-12 col-sm-12 col-xs-12 static-page-content">
        <h1><?= !empty($seo['title_h1']) ?$seo['title_h1'] :$this->title?></h1>

```

Рисунок 4.3 - Використання системи розмітки

4.8.3 Випадаюче меню

Кнопки в випадаючому меню можуть мати будь-який розмір. Створено меню, елементи якого піднімаються вгору, додавши `.dropdown` до батьківського контейнера.

```

<! - Split button ->
<Div class = "btn-group">
  <Button type = "button" class = "btn btn-danger"> Дія </ button>
  <Button type = "button" class = "btn btn-danger dropdown-toggle"
data-toggle = "dropdown">
  <Span class = "caret"></ span>
  <Span class = "sr-only"> Меню з перемиканням </ span>
</ Button>
  <Ul class = "dropdown-menu" role = "menu">
    <Li><a href="#"> Каталог </a></ li>
    <Li><a href="#"> Оголошення </a></ li>
    <Li><a href="#"> Послуги </a></ li>
    <Li class = "divider"></ li>
    <Li><a href="#"> Аукціон </a></ li>
  </ Ul>
</ div>

```

4.8.4 Групи форми введення і пошук

Для того щоб створити пошук, знадобляться наступні компоненти Bootstrap: інлайнова форма, згруповані контролю і кнопка.

У тезі header створимо інлайнову форму, вирівняну по правому краю. Поля такої форми обов'язково повинні мати клас «`form-control`» і `label`.

У форму ми поміщаємо компонент «згруповані контролю». Угрупування контролів дозволяє прибрати відступ між введенням тексту і кнопкою і як би злити їх в єдиний

елемент. Він представляє з себе `div` з класом `input-group` і полями, а кнопка такого компонента поміщається в блок з класом `input-group-btn`.

Label для поля пошуку приховуємо класом `sr-only`. Це потрібно для спеціальних пристроїв читання з екрана. Кнопці додається клас `btn-primary`, що означає, що це первинна кнопка даної форми.

4.8.5 Media-запити

Для мобільних пристроїв були створені і підключені окремі CSS-файли `mobile-320.css` і `mobile-768.css` які викликаються за допомогою медіа-запитів виду:

```
@media screen and (min-width: 768px) and (max-width: 1224px) {
  .mobile-768.best-goods-widget {
    display: block;
    width: 100%;
  }
}
```

Рисунок 4.4 - Медіа-запит

Медіа-запити використовуються в Less-файлах, щоб створити визначальну контрольну точку в системі розмітки.

Такого роду запити дозволили приховати або видозмінити окремі блоки, адаптувавши дизайн для мобільних пристроїв. Таким чином, наприклад, були приховані бічні відступи, адаптовано верхнє меню сайту, пошук, основні блоки товарних пропозицій, змінено розташування блоків в нижній частині сайту (футері).

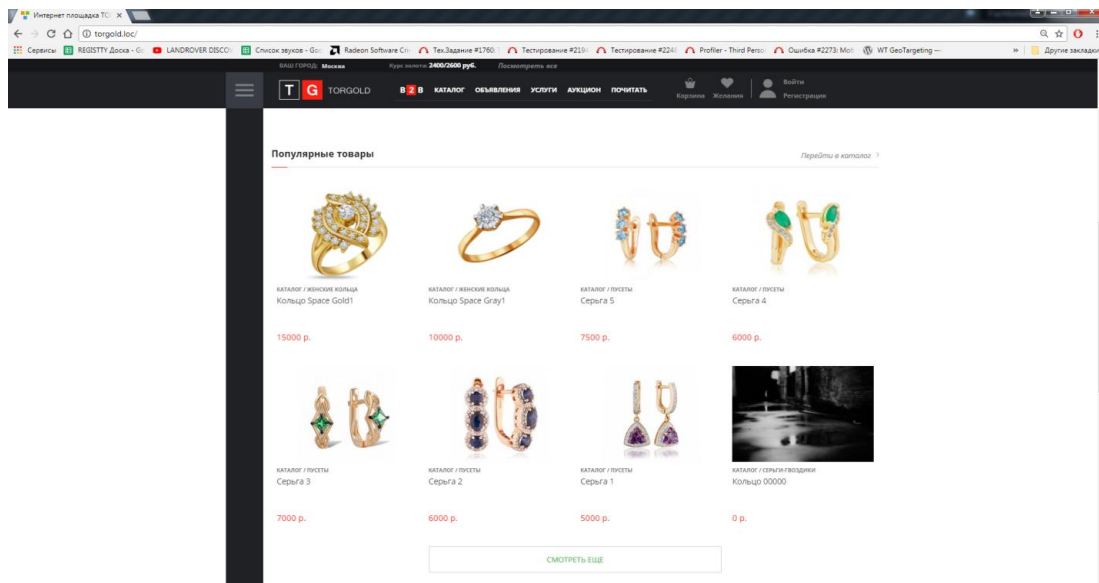


Рисунок 4.5 – Скріншот сайту для комп'ютера

Для того, щоб створені колонки склалися на невеликих пристроях використовувалися дуже маленькі xs або середні md-класи розмітки пристрої додаючи .col-xs- * .col-md- * до стовпців.

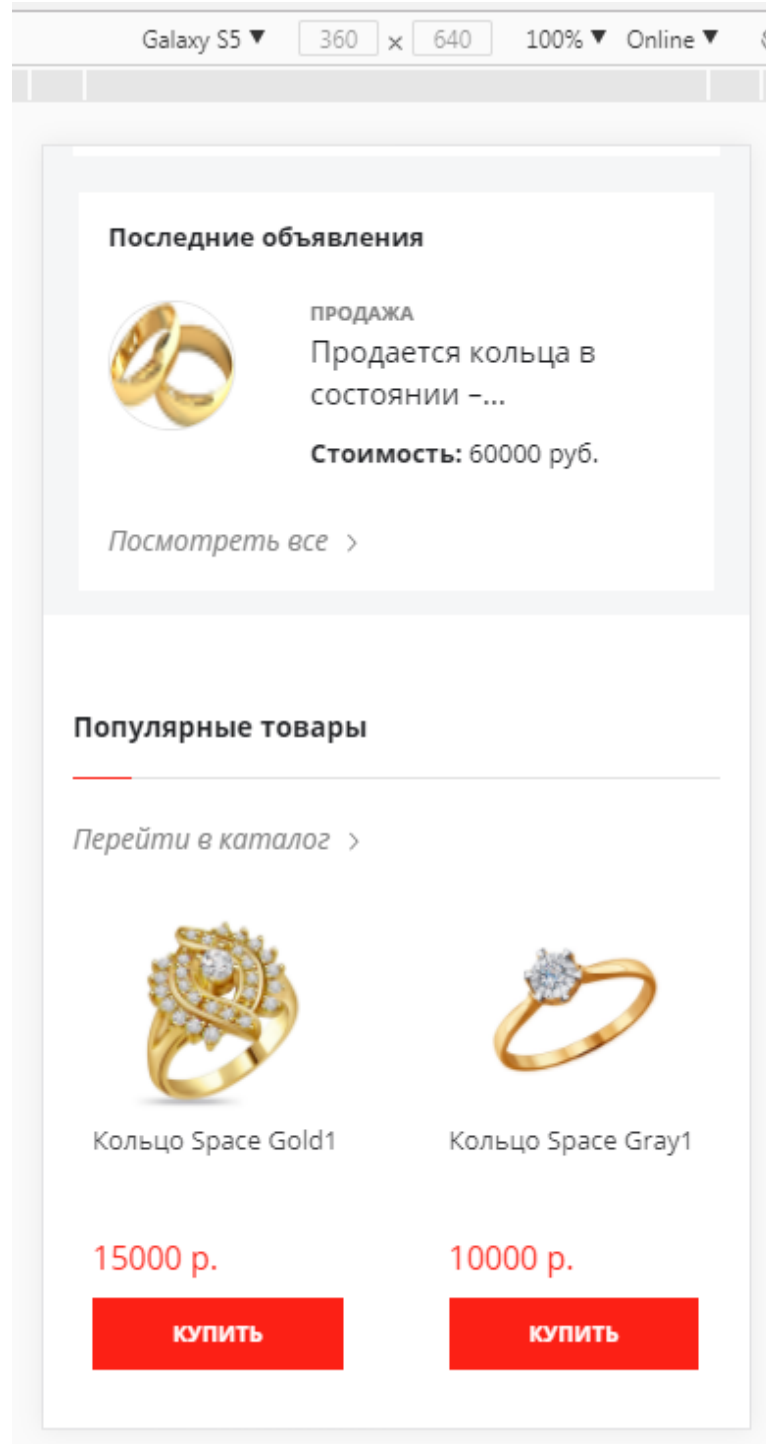


Рисунок 4.6 – Скріншот мобільної версії сайту

4.8.6 Анімація слайдера

Слайдери, як правило робляться з використанням Javascript. Зазвичай для цього задіють бібліотеку JQuery. Хоча CSS і не дозволяє створити повнофункціональний слайдер з купою можливостей, таких як анімація за таймером, перегортання пальцем на тач-екрані та інше, у багатьох випадках такі просунуті можливості і не потрібні.

Слайдер набагато простіше зверстати на чистому HTML і CSS, ніж задіяти для цього можливості Javascript. По-перше, код стилів слайдера вийде набагато простіше і зрозуміліше, ніж код слайдера на Javascript. По-друге, та частина користувачів, яка використовує аддон NoScript, який видаляє весь Javascript код, теж зможе скористатися вашим слайдером.

Створимо каркас для слайдера:

```
<divclass = 'sliderA'>
  <input type = "radio" name = "slider1" id = "slider1_1" checked =
"checked">
  <label for = "slider1_1"></ label>
  <div></ div>
  <input type = "radio" name = "slider1" id = "slider1_2">
  <label for = "slider1_2"></ label>
  <div></ div>
  <input type = "radio" name = "slider1" id = "slider1_3">
  <label for = "slider1_3"></ label>
  <div></ div>
  <input type = "radio" name = "slider1" id = "slider1_4">
  <label for = "slider1_4"></ label>
  <div></ div>
</ div>
```

Контейнер слайдера - div з класом sliderA. Кожен кадр слайдера описується трьома тегами:

Радіокнопкою (inputtype = "radio"), що відповідає за стан даного кадру (видно / не видно)

Міткою label, що відповідає за відображення кнопки переходу на даний слайдер.

Тегом div в якому знаходиться вміст кадру.

Radio-кнопки всередині одного слайдера повинні мати однакове ім'я name. Також, всі radio-кнопки повинні мати унікальні (різні) ідентифікатори id. Мітка for повинна містити значення id відповідної radio-кнопки.

Стили для контейнера:

```
.sliderA {
```

```

width: 400px;
height: 250px;
border: 1px solid # 888;
position: relative;
text-align: center;
}

```

Були задані фіксовані розміри контейнера. Розміри встановлені рівними розміру зображень, які будуть використовуватися. Властивість `position: relative` необхідна, щоб задати нову точку відліку позицій вкладених контейнерів. Властивість `text-align: center` додано для вирівнювання по центру смугу міток-кнопок, які будуть позиціонуватися як `inline-елементи`.

Відключення відображення радіо-кнопок:

```

.sliderA> input {
  display: none;
}

```

Оформлення мітки-кнопки:

```

.sliderA> input + label {
  display: inline-block;
  width: 10px;
  height: 10px;
  border-radius: 8px;
  background: rgba (90, 90, 90, 0.8);
  border: 2px solid rgba (190, 190, 190, 0.8);
  cursor: pointer;
  z-index: 100;
  position: relative;
  margin-right: 4px;
  top: 90%;
  transition: background 0.8s ease-out 0s;
}

```

```

.sliderA> input + label: hover {
  background: rgba (250, 250, 250, 0.8);
}
.sliderA> input: checked + label {
  background: rgba (220, 220, 220, 0.8);
}

```

Встановлена властивість `display: inline-block`, дозволяє розмістити мітки одним рядком і вирівняти їх по центру як рядкові елементи, і в той же час поставити їм фіксовані розміри як блокам. Потім задається ширина, висота, а також радіус заокруглення кутів,

колір фону і колір кордону. Все це призводить до того, що мітки будуть відображатися у вигляді маленьких круглих кнопок.

Властивість `cursor` дозволяє задати зовнішній вигляд курсору миші при наведенні на кнопку. `Pointer`, задає вид покажчика миші: покажчик буде такою ж, як при наведенні на посилання.

Властивість `z-index` необхідно для того, щоб мітки-кнопки лежали поверх кадрів слайдера. Для цього ж встановлено властивість `position: relative` - інакше `z-index` не працюватиме.

Властивість `margin-right` додає відступи між кнопками, а `top: 90%` зрушує їх в нижню частину контейнера.

Також поставлено колір фону для натиснутою кнопки і кнопки, на яку наведена миша. Властивість `transition` визначає анімацію для зміни фону.

Слід усунути всі прогалини і переклади рядків між тегами слайдера, так як при використанні `display: inline-block` вони дадуть нам зайві зазори між кнопками.

Додавання стилів для `div`, в яких буде перебувати вміст кадру:

```
.sliderA> div {
    position: absolute;
    top: 0px;
    left: 0px;
    right: 0px;
    bottom: 0px;
    z-index: 0;
}
```

Всі кадри будуть займати одне і те ж положення `position: absolute` дозволяє вилучити блоки з потоку і розмістити з довільним чином. `top`, `left`, `right`, `bottom` задають координати кадру таким чином, що кадр займає весь контейнер слайдера цілком. `z-index` відправляє кадри під кнопки перемикання кадрів.

Найголовніша частина слайдера - показ та приховування кадрів в залежності від натиснутої кнопки. Традиційний підхід приховування елементів сторінки з використанням властивості `display: none` не зовсім підходить, оскільки не дозволяє реалізувати плавне перемикання кадрів, але і CSS не дозволяє задати анімацію значення `display` за допомогою правил `transition`.

Для плавного приховування і появи кадрів були застосовані три властивості: `opacity`, `visibility` і `transition`.

Властивість `opacity` дозволяє у вигляді десяткового дробу задати рівень прозорості елемента сторінки, від 0 (повністю прозорий) до 1 (повністю непрозорий). Наприклад, `opacity: 0.5` - прозорість 50%.

Але навіть повністю прозорий елемент, хоч і невидимий, продовжує реагувати на натискання мишею. Якщо в слайдері знаходяться посилання, натискання на посилання може привести до спрацьовування зовсім не того посилання, яку видно на екрані.

Щоб повністю відключити елемент і зробити його прозорим для клацань мишею, було застосовано властивість: `visibility: hidden`. Однак `visibility` не дозволяє поставити часткову прозорість. Елемент або видно, або ні.

Тому за допомогою `transition` були анімовані обидві властивості, щоб досягти потрібного ефекту.

Суть методу в наступному: коли користувач перемикає кадр, новий кадр з'являється з `visibility: visible`, але з `opacity: 0`.

Потім відбувається анімація значення `opacity` у обох кадрів. У старому кадрі вона плавно опускається до нуля, а у новому - плавно піднімається до одиниці.

Після завершення анімації, `visibility` старого кадру встановлюється в значення `hidden`, щоб він не заважав клацати мишею по новому кадру.

```
.sliderA> div {
  visibility: hidden;
  opacity: 0;
  transition: opacity 0.8s ease-out 0.1s, visibility 0.1s ease-out
0.9s;
}
.sliderA> input: checked + label + div {
  visibility: visible;
  opacity: 1;
  transition: opacity 0.8s ease-out 0.1s, visibility 0.1s ease-out
0s;
}
```

Створено правило для тега `<p>` всередині кадру, щоб зробити підписи до картинок:

```
.sliderA> div> p {
  position: absolute;
  top: 0px;
  left: 0px;
  right: 0px;
  text-align: center;
  color: #fff;
  text-shadow: 1px 1px 2px # 000;
}
```

У слайдері можна розміщувати як звичайні фотографії, так і будь-який текст, посилання на інші сторінки сайту. У шаблон слайдера додаються по ще одній мітці для кожного кадру:

```

<div class = 'sliderA'>
  <input type = "radio" name = "slider2" id = "slider2_1" checked =
"checked">
  <label for = "slider2_1"></ label>
<div></ div>
  <label for = "slider2_2"></ label>
  <input type = "radio" name = "slider2" id = "slider2_2">
  <label for = "slider2_2"></ label>
<div></ div>
  <label for = "slider2_3"></ label>
  <input type = "radio" name = "slider2" id = "slider2_3">
  <label for = "slider2_3"></ label>
<div></ div>
  <label for = "slider2_4"></ label>
  <input type = "radio" name = "slider2" id = "slider2_4">
  <label for = "slider2_4"></ label>
<div></ div>
  <label for = "slider2_1"></ label>
</div>

```

Мітка йде після першого кадру, активує другий кадр. Мітка, розташована після другого кадру, активує третій кадр; після третього - четвертий. Остання мітка активує перший кадр.

Мітки розташовані так, щоб вони перекривали весь слайдер і лежали вище Рисунок , але нижче смуги кнопок. Мітки прозорі, тому крізь них видно вміст кадру. При показі кадру номер N разом з ним відобразиться позначка для переходу на кадр N + 1. (Для останнього кадру - мітка переходу на перший кадр.)

```

.sliderA> input + label + div + label {
    display: none;
}
.sliderA> input:checked + label + div + label {
    display: block;
    position: absolute;
    top: 0px;
    left: 0px;
    right: 0px;
    bottom: 0px;
    z-index: 50;
}

```

Таким чином, при натисканні на зображення спрацьовує активація наступного зображення.

5 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

5.1 Загальні питання з охорони праці

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно–побутові умови повинні відповідати вимогам нормативних актів про охорону праці. В [23] визначається, що охорона праці – це система правових, соціально–економічних, організаційно–технічних, санітарно–гігієнічних і лікувально–профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

Завданням даної дипломної роботи було розробити та дослідити програмні засоби для забезпечення безпеки даних в комп'ютерних системах, і як результат було створено три програми, що забезпечують криптографічний захист інформації. За цим в подальшому розроблятиметься реальна система, яка значно полегшить процес забезпечення захисту інформації.

При роботі з обчислювальною технікою змінюються фізичні і хімічні фактори навколишнього середовища: виникає статична електрика, електромагнітне випромінювання, змінюється температура і вологість, рівень вміст кисню і озону в повітрі. Повітря забруднюється шкідливими хімічними речовинами антропогенного походження за рахунок деструкції полімерних матеріалів, які використовуються для обробки приміщень та обладнання. Неправильна організація робочого місця сприяє загальному і локальній напрузі м'язів шиї, тулуба, верхніх кінцівок, викривлення хребта і розвитку остеохондрозу. На всіх підприємствах, в установах, організаціях повинні створюватися безпечні і нешкідливі умови праці. Забезпечення цих умов покладається на власника або уповноважений ним орган (далі роботодавець). Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно–побутові умови повинні відповідати вимогам нормативних актів про охорону праці. Роботодавець повинен впроваджувати сучасні засоби техніки безпеки, які запобігають виробничому травматизмові, і забезпечувати санітарно–гігієнічні умови, що запобігають виникненню професійних захворювань працівників. Він не має права вимагати від працівника виконання роботи, поєднаної з явною небезпекою для життя, а також в умовах, що не відповідають законодавству про охорону праці. Працівник має

право відмовитися від дорученої роботи, якщо створилася виробнича ситуація, небезпечна для його життя чи здоров'я або людей, які його оточують, і навколишнього середовища.

5.1.1 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 законодавство про охорону праці складається та прийнятих відповідно до них нормативно-правових актів, норм міжнародного договору (ратифіковані Конвенції і Рекомендації МОТ, директиви Європейської Ради).

На законодавчому рівні визначено такі пріоритетні напрямки з безпеки праці:

- кожен працівник несе безпосередню відповідальність за порушення зазначених Законом, нормами і правилами вимог;
- напрямки реалізації конституційного права громадян на їх життя і здоров'я в процесі трудової діяльності:
- пріоритет життя і здоров'я працівників по відношенню до результатів виробничої діяльності підприємства;
- повна відповідальність роботодавця за створення належних – безпечних і здорових умов праці;
- соціальний захист працівників, повне відшкодування збитків особам, які потерпіли від нещасних випадків на виробництві та професійних захворювань;
- комплексне розв'язання завдань охорони праці;
- підвищення рівня промислової безпеки шляхом забезпечення суцільного технічного контролю за станом виробництв, технологій та продукції, а також сприяння підприємствам у створенні безпечних та нешкідливих умов праці;
- соціальний захист працівників, повне відшкодування збитків особам, які потерпіли від нещасних випадків на виробництві та професійних захворювань;
- використання економічних методів управління охороною праці, участь держави у фінансуванні заходів щодо охорони праці;
- використання світового досвіду організації роботи щодо поліпшення умов і підвищення безпеки праці на основі міжнародної співпраці.

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.

Обов'язки працівників щодо додержання вимог нормативно-правових актів з охорони праці (ст. 14), відповідальність робітників всіх категорій за порушення вимог щодо охорони праці (ст. 44) та структура організації/виробництва системи управління охорони праці визначені безпосередньо «Інструкцією на робоче місце № 1», та іншими затвердженими власними нормативними актами з питань охорони праці (правилами, нормами, регламентами, положеннями, стандартами, інструкціями та іншими документами, обов'язковими до виконання), тобто тих, що діють на підприємстві/організації.

Наявні трудові відносини між працівниками і роботодавцями в Україні за темою роботи регулюються, відповідно до якого права працюючої людини на охорону праці охороняються всебічно та норми охорони праці неухильно інтегровані до правил внутрішнього розпорядку організації/підприємства.

5.1.2 Організаційно-технічні заходи з безпеки праці

В організації/підприємстві проводиться навчання і перевірка знань з питань охорони праці відповідно до вимог , затвердженого наказом Держнаглядохоронпраці України від 26.01.2005 N 15, зареєстрованого в Міністерстві юстиції України 15.02.2005 за N 231/10511.

Також впроваджені організаційні заходи з пожежної безпеки – навчання і перевірку знань відповідно до вимог, затвердженого наказом Міністерства України з питань надзвичайних ситуацій та у справах захисту населення від наслідків Чорнобильської катастрофи від 29.09.2003 N 368, зареєстрованого в Міністерстві юстиції України 11.12.2003 за N 1148/8469.

Обов'язковими вимогами враховане наступне:

– 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.

– на підприємстві/організації, де експлуатуються ПК з ВДТ і ПП, розробляється інструкція з охорони праці відповідно до [29], затвердженого наказом

Держнаглядохоронпраці від 29.01.98 N 9, зареєстрованого в Міністерстві юстиції України 07.04.98 за N 226/2666.

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

–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 об'єкта господарювання з усіма працівниками, яких приймають на роботу незалежно від їхньої освіти та стажу роботи за програмою, в якій подають загальні питання охорони праці із врахуванням її особливостей на об'єкті господарювання;

2. Первинного, який проводять керівники структурних підрозділів на місці праці з кожним працівником до початку їхньої роботи на цьому робочому місці.

Пройдення працівником цих інструктажів з питань охорони праці підтверджується записами у відповідних журналах обліку інструктажів і скріплюється підписами осіб, які проводили інструктажі та осіб, які отримали інструктажі.

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)

– обов'язкові організаційні заходи перед початком, під час і після завершення роботи повинні включати перевірку (візуально) наявності і справності електрообладнання та його заземлення, а під час виконання роботи вимогу «не залишати без нагляду обладнання, яке працює». Після закінчення роботи – вимагається прибирання робочого місця, відключення всіх електроприладів від електромережі.

Не допускається:

– виконувати обслуговування, ремонт та налагодження ПК з ВДТ і ПП безпосередньо на робочому місці оператора;

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

– відключати захисні пристрої, самочинно проводити зміни у конструкції та складі ПК з ВДТ і ПП або їх технічне налагодження;

– працювати з ВДТ, у яких під час роботи з'являються нехарактерні сигнали, нестабільне Рисунок на екрані тощо;

–працювати з матричним принтером за відсутності вібраційного килимка та зі знятою (піднятою) верхньою кришкою.

5.2 Аналіз стану умов праці

5.2.1 Вимоги до приміщень

Робота над створенням такої системи проходитиме в приміщенні відповідної установи (компанії, підприємстві тощо). Для даної роботи достатньо однієї людини, для якої надано робоче місце зі стаціонарним комп'ютером. ГПКетричні розміри приміщення зазначені в таблиці 5.1.

Таблиця 5.1 – Розміри приміщення

Найменування	Значення
Довжина, м	5
Ширина, м	5
Висота, м	3
Площа, м ²	25
Об'єм, м ³	75

Згідно з [30] 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 рівного 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.

5.2.2 Вимоги до організації місця праці

При порівнянні відповідності характеристик робочого місця нормативним основні вимоги до організації робочого місця [31] і відповідними фактичними значеннями для робочого місця, констатуємо повну відповідність в таблиці 5.2.

Таблиця 5.2 – 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 кривини спинки в горизонтальній площині, мм	400	400
Відстань від очей до екрану дисплея, мм	800	700 ÷ 800

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 78 м³, площу — 18 м². У цьому кабінеті обладнано три місця праці, з яких два укомплектовані ПК.

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 на робочому місці 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.

У кабінеті є електрична мережа з напругою 220 В, яка створює небезпеку ураження електричним струмом. ПК та периферійні пристрої можуть бути джерелами електромагнітних випромінювань, аерозолів та шкідливих речовин (часток тонеру, оксидів нітрогену та озону).

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

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

5.2.3 Навантаження та напруженість процесу праці

Як приклад наведено опис процесу праці оформлення роботи під час виконання магістерської роботи за фізичним навантаженням робота відноситься до категорії легкі роботи (Ia), її виконують сидячи з періодичним ходінням. Щодо характеру організації роботи, то 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, що призводить до навантаження на окремі системи організму. Такі перекося у напруженні різних систем організму, що трапляються під час роботи з ПК, зокрема, значна напруженість зорового аналізатора і довготривале малорухоме положення перед екраном, не тільки не зменшують загального напруження, а навпаки, призводять до його посилення і появи стресових реакцій.

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

Тобто наявне психофізіологічні небезпечні та шкідливі фактори:

а) фізичного перевантаження:

- статичного;
- динамічного;
- б) нервово–психічного перевантаження:
 - розумового перенапруження;
 - монотонності праці;
 - перенапруження аналізаторів;
 - емоційних перевантажень.

Роботу за дипломним проектом визнано, таку, що займає 50% часу робочого дня та за восьмигодинної робочої зміни рекомендовано встановити додаткові регламентовані перерви:

- для розробників програм тривалістю 15 хв. через кожну годину роботи.

5.3 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.

5.3.1 Аналіз небезпечних та шкідливих факторів при роботі на ПК

Роботу, пов'язану з персональним комп'ютером (далі – ПК) з відео дисплейними терміналами (далі – ВДТ), у тому числі на тих, які мають робочі місця, обладнані ПК з ВДТ і периферійними пристроями (далі – ПП), виконують із забезпеченням виконання [32], які встановлюють вимоги безпеки до обладнання робочих місць, до роботи із застосуванням ПК 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.

Робочі місця мають відповідати вимогам [31, 32].

Це передбачає, що визначена виробнича діяльність пов'язана з наявністю певної кількості небезпечних та/або шкідливих виробничих факторів. Тому у першій частині цього підрозділу за результатами аналізу повинні бути визначені такі фактори.

Робота ПК та периферійних пристроїв супроводжує виділення багатьох хімічних речовин, зокрема озону, оксидів нітрогену та аерозолів (високодисперсних частинок тонера). Для прикладу, за умов роботи з ПК виникають наступні небезпечні та шкідливі

чинники: несприятливі мікрокліматичні умови, освітлення, електромагнітні випромінювання, забруднення повітря шкідливими речовинами (джерелом яких може бути принтер, сканер та ін.), шум, вібрація, електричний струм, електростатичне поле, напруженість трудового процесу та інше.

Аналіз небезпечних та шкідливих виробничих факторів виконується у табличній формі (табл. 5.3).

Таблиця 5.3 – Аналіз небезпечних і шкідливих виробничих факторів

Небезпечні і шкідливі виробничі фактори	Джерела факторів (види робіт)	Кількісна оцінка	Нормативні документи
<i>фізичні</i>			
– підвищена температура поверхонь обладнання	Експлуатація ПК	2	ДСН 3.3.6.042–99
– підвищений рівень шуму на робочому місці	Система охолодження ПК	2	ДСН 3.3.6.037–99
– підвищений рівень вібрації	Система охолодження ПК, привід	2	ДСН 3.3.6.039–99 ДСТУ ГОСТ 12.1.012–90
– недостатність природного світла	Порушення умов праці (вимог до приміщень)	2	ДБН В.2.5–28:2015
– недостатнє освітлення робочої зони	Порушення гігієнічних параметрів виробничого середовища	3	ДБН В.2.5–28:2015
– підвищена яскравість світла	Порушення умов праці (організації місця праці–налагодження моніторів)	1	ДСанПіН 3.3.2.007–98
<i>психофізіологічні:</i>			
– нервово–психічна перевантаження (розумове, перенапруження аналізаторів–зорових)	– пошук інформації для постановки теми; – пошук та аналіз аналогів і літератури; – пошук наявних технологій, моделювання та аналіз алгоритмів; – виконання роботи за темою диплома, тестування; – оформлення роботи	4	НПАОП 0.00–1.28–10 ДСанПіН 3.3.2.007–98
– фізичні (статичне – сидіння)	порушення умов праці (організації місця праці–сидіння користувача,) та організації робочого часу – безпервна робота)	2	НПАОП 0.00–1.28–10 ДСанПіН 3.3.2.007–98

5.3.2 Пожежна безпека

Небезпека розвитку пожежі на обчислювальному центрі обумовлюється застосуванням розгалужених систем електроживлення ПК, 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. Надійна робота окремих елементів і мікросхем в цілому забезпечується тільки в певних інтервалах температури, вологості і при заданих електричних параметрах. 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 деталях виділяється тепло, що в умовах їх високої щільності може привести до перегріву, і може служити причиною запалювання ізоляційних матеріалів. Слабкий опір ізоляційних матеріалів дії температури може викликати порушення ізоляції і привести до короткого замикання між струмоведучими частинами обладнання (шини, електроди). Також ймовірна небезпека внаслідок перевантаження напруги, розрядки зарядів статичної електрики, пошкодження обладнання та електропроводки. Електростатичний розряд виникає під час тертя двох ізольованих матеріалів.

Пожежна безпека при застосуванні ПК забезпечується:

- системою запобігання пожежі;
- системою протипожежного захисту;
- організаційно–технічними заходами.

Згідно [33] таке приміщення, площею 25 м², відноситься до категорії "В" (пожежонебезпечної) та для протипожежного захисту в ньому проектом передбачено устаткування автоматичною пожежною сигналізацією із застосуванням датчиків–сповіщувачів РІД–1 (сповіщувач димовий ізоляційний) в кількості 1 шт., і застосуванням первинних засобів пожежогасіння. Відповідно до норм первинних засобів пожежогасінні пропонується використовувати:

- ручний вуглекислий вогнегасник ОУ–5 в кількості 1 шт. або хімічний пінний ОХП–10 – 1 шт;
- ковсть 1 1 м², кошму 2Ч1,5 м² або азбестове полотно 2Ч2 м² в кількості 1 шт.

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.

Горючими матеріалами в приміщенні, де розташовані ПК, є:

- поліамід – матеріал корпусу мікросхем, горюча речовина, температура самозаймання 420 °С,
- полівінілхлорид – ізоляційний матеріал, горюча речовина, температура запалювання 335 °С, температура самозаймання 530 °С,
- склотекстоліт ДЦ – матеріал друкарських плат, важкогорючий матеріал, показник горючості 1.74, не схильний до температурного самозаймання,
- пластикат кабельний №.489 – матеріал ізоляції кабелів, горючий матеріал, показник горючості більше 2.1,
- деревина – будівельний і обробний матеріал, з якого виготовлені меблі, горючий матеріал, показник горючості більше 2.1, температура запалювання 255 °С, температура самозаймання 399 °С.

Для відводу теплоти від ПК діє система кондиціонування. Тому кисень, як окиснювач процесів горіння, є в будь-якій точці приміщень ВЦ.

Простори усередині приміщень в межах, яких можуть утворюватися або знаходитися пожежонебезпечні речовини і матеріали відповідно [33] відносяться до пожежонебезпечної зони класу П–Па. Це обумовлено тим, що в приміщенні знаходяться тверді горючі та важкозаймисті речовини та матеріали. Приміщенню, у якому розташоване робоче місце, присвоюється II ступень вогнестійкості.

Потенційними джерелами запалювання можуть бути:

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

Причинами можливого загоряння і пожежі можуть бути:

- несправність електроустановки;
- конструктивні недоліки устаткування;
- коротке замикання в електричних мережах;

–запалювання горючих матеріалів, що знаходяться в безпосередній близькості від електроустановки.

Продуктами згорання, що виділяються на пожежі, є: окис вуглецю; сірчистий газ; окис азоту; синильна кислота; акромін; фосген; хлор і ін. При горінні пластмас, окрім звичних продуктів згорання, виділяються різні продукти термічного розкладання: хлорангідридні кислоти, формальдегіди, хлористий водень, фосген, синильна кислота, аміак, фенол, ацетон, стирол [34].

Для захисту персоналу від дії небезпечних і шкідливих чинників пожежі проектом передбачається застосування промислового протигаза, 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).

Можливе також відповідне застосування фільтрувальної коробки з маркуванням «СО» із фіолетовим забарвленням на фільтрі означає, що він захищає від Чадного газу. Або фільтру для протигазу з літерним маркуванням «SX» із фіолетовим забарвленням захистить від спец речовин таких як (зарин, зоман та фосген).

5.3.3 Електробезпека

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. Нульовий захисний провідник використовується для заземлення (занулення) електроприймачів. Штепсельні з'єднання та електророзетки крім контактів фазового та нульового робочого провідників мають спеціальні контакти для підключення нульового захисного провідника. Електромережа штепсельних розеток для живлення персональних ПК, укладено по підлозі поруч зі стінами відповідно до затвердженого плану розміщення обладнання та технічних характеристик обладнання. Металеві труби та гнучкі металеві рукави заземлені. Захисне заземлення включає в себе заземлюючих пристроїв і провідник, який з'єднує заземлюючий пристрій з обладнанням, яке заземлюється – заземлюючий провідник.

5.4 Гігієнічні вимоги до параметрів виробничого середовища

5.4.1 Мікроклімат

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 Ia. Отже оптимальні значення для температури, відносної вологості й рухливості повітря для зазначеного робочого місця відповідають [35] і наведені в таблиці 5.4:

Таблиця 5.4 – 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

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 у відповідності [35]. Рівні позитивних і негативних іонів у повітрі мають відповідати [35]. Для забезпечення оптимальних параметрів мікроклімату в приміщенні проводяться перерви в роботі користувача, з метою його провітрювання. Існують спеціальні системи кондиціонування, які забезпечують підтримання в приміщенні балансу оптимальних параметрів мікроклімату. Контроль параметрів мікроклімату в холодний і теплий період року здійснюється не менше 3–х разів на зміну (на початку, середині, в кінці).

5.4.2 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. При поганому освітленні людина швидко втомлюється, працює менш продуктивно, виникає потенційна небезпека помилкових дій і нещасних випадків.

Освітленість приміщення має велике значення при роботі на ППК. Вона багато в чому визначається колірною і мережевий обстановкою. Для зменшеного поглинання світла стеля і стіни вище панелей (1,5–1,7м.). Якщо вони не облицьовані звукопоглинальним матеріалом, фарбуються білою водоемульсійною фарбою (коефіцієнт відбиття повинен бути не менше 0,7). Для забарвлення стіни панелей рекомендується віддавати перевагу світлим фарбам.

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

Робота на ППК може здійснюватися за таких видах освітлення:

–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;

–суміщене освітлення (природне + штучне) тільки при одному і трьох рядном розташуванні робочих місць, коли екран і поверхню робочого столу знаходяться перпендикулярно світла несучій стіні. При цьому штучне освітлення буде виконане стельовими або підвісними люмінесцентними світильниками, рівномірно розміщеними по стелі рядами паралельно світловим прорізам так, щоб екран відео монітора знаходився в зоні захисного кута світильника, і його проекції не доводилися на екран. Працюючі на ППК не повинні бачити відображення світильників на екрані. Застосовувати місцеве освітлення при роботі на ППК не рекомендується.

Природне освітлення, коли робочі місця з ППК розташовуються в один ряд по довжині приміщення на відстані 0,8 – 1,0 м від стіни з віконними прорізами, і екрани знаходяться перпендикулярно цієї стіни. Основний потік природного світла при цій повинен бути зліва. Не допускається спрямування основного світлового потоку природного світла праворуч, ззаду і спереду працює на ППК. Оптимальна відстань очей до екрана відео монітора повинна становити 60–70 см, допустиме не менше 50 см. Розглядати інформацію ближче 50 см не рекомендується.

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), спектральний складом випромінюваного світла, близький до сонячного. При експлуатації ПК виконується зорова робота IV в розряді точності (середня точність). При цьому нормована освітленість на робочому місці (Ен) рівна 200 лк. Джерелом природного освітлення є сонячне світло.

У приміщенні, де розташовані ПК передбачається природне бічне освітлення, рівень якого відповідає [36]. Джерелом природного освітлення є сонячне світло. Регулярно повинен проводитися контроль освітленості, який підтверджує, що рівень освітленості задовольняє [36] і для даного приміщення в світлий час доби достатньо природного освітлення.

Розрахунок освітлення.

Для будівель виробництв світловий коефіцієнт приймається в межах 1/6 – 1/10:

$$\sqrt{a^2 + b^2} \cdot S_b = (1/8 \div 1/10) \cdot S_n \quad (4.1)$$

де S_b – площа віконних прорізів, м²;

S_n – площа підлоги, м².

$$S_n = a \cdot b = 5 \cdot 5 = 25 \text{ м}^2$$

$$S_{\text{вік}} = 1/8 \cdot 25 = 3,125 \text{ м}^2$$

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 типа 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 здійснюється по формулі:

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

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

S – освітлювана площа, м²; $S = 25$ м²;

Z – поправочний коефіцієнт світильника (для стандартних світильників $Z = 1.1$ – 1.3) 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.

Підставивши числові значення у формулу (4.2), отримуємо:

$$n = \frac{300 \cdot 25 \cdot 1,1 \cdot 1,5}{5400 \cdot 0,575 \cdot 2} \approx 2,64$$

Приймаємо освітлювальну установку, яка складається з 3-х світильників, які складаються з двох люмінесцентних ламп загальною потужністю 160 Вт, напругою – 220 В.

Потужність електроосвітлювальної установки з урахуванням місцевого освітлення визначається за формулою:

$$N = \frac{n \cdot W + (0,1 \div 0,2) \cdot n \cdot W}{1000}, \text{ кВт} \quad (4.3)$$

де n – розрахункова кількість ламп для освітлення даного приміщення;

W – потужність однієї лампи, Вт;

(0,1ч0,2) – додаткова потужність для ламп місцевого освітлення, Вт.

$$N = \frac{3 \cdot 160 + 0,2 \cdot 3 \cdot 160}{1000} = 0,576 \text{ кВт}$$

5.4.3 Шум та вібрація, електромагнітне випромінювання

Рівень шуму, що супроводжує роботу користувачів персональних комп'ютерів, а також зовнішніми чинниками, коливається у межах 50–65 дБА [33]. 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, а в приміщеннях, де виконують роботу, що потребує зосередженості, — до 55 дБА. У залах опрацювання інформації та комп'ютерного набору рівні шуму не повинні перевищувати 65 дБА.

Шум часто є причиною зниження рівня працездатності, підвищення рівня загальної та професійної захворюваності, частоти виробничих травм. Шум є загальнобіологічним подразником, який негативно впливає на всі органи і системи організму. У разі тривалого систематичного впливу шуму може виникнути патологія з переважним ураженням слуху, центральної нервової і серцево–судинної систем.

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. Додатковим звукопоглинанням в КВТ можуть бути фіранки, підвішені в складку на відстані 15–20 см. Від огорожі, виконані з щільної, важкої тканини. У приміщенні з ПК коректований рівень звукової потужності не перевищує 45 дБА. Оскільки рівень шуму не перевищує гранично допустимих величин, які встановлені санітарними нормами, заходи для зниження шуму не проводяться.

Віброізоляція можливо здійснювати за допомогою спеціальної прокладки під системний блок, який послаблює передачу вібрацій робочого столу. Вібрація на робочому місці в приміщенні, що розглядається, відповідає нормам [33]. Допустимий рівень вібрацій на робочому місці: – для 1 ступеня шкідливості до 3 дБ; – для 2–3 – 1–6 дБ; – для 3 – більше 6 дБ.

Для захисту від електромагнітного випромінювання передбачаються наступні заходи:

- застосування нових плазмових моніторів,
- віддалення робочого місця не менше, ніж на 0,4 – 0,5 м, оскільки напруженість електричного поля зменшується при віддаленні від джерела поля,

- встановлення раціональних режимів роботи персоналу (обмеження часу перебування),
- раціональне розміщення в робочому приміщенні устаткування, що випромінює електромагнітну енергію.

5.4.4 Вентилювання

У приміщенні, де знаходяться ПК, повітрообмін реалізується за допомогою природної організованої вентиляції (вентиляційні шахти). Цей метод має забезпечити приток потрібної кількості свіжого повітря, що визначається [36] (30 м^3 на годину на одного працюючого).

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

5.5 Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій

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

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

- правильне організування місця праці та дотримання оптимальних режимів праці та відпочинку під час роботи з ПК;
- експлуатацію сертифікованого обладнання;
- дотримання заходів електробезпеки;
- забезпечення оптимальних параметрів мікроклімату;
- забезпечення раціонального освітлення місця праці (освітленість робочого місця не перевищувала $2/3$ нормальної освітленості приміщення).

Зниження рівня шуму та вібрації:

- у джерелі виникнення, шляхом застосування раціональних конструкцій, нових матеріалів і технологічних процесів;
- звукоізолювання устаткування за допомогою глушників, резонаторів, кожухів, захисних конструкцій, оздоблення стін, стелі, підлоги тощо;
- використання засобів індивідуального захисту).

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

- 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;
- постійно стежити за справністю ізоляції електромережі та мережевих кабелів, не допускаючи їхньої експлуатації з пошкодженою ізоляцією;
- не тягнути за мережевий кабель, щоб витягти вилку з розетки;
- не закривати меблями, різноманітним інвентарем вимикачі, штепсельні розетки;
- не підключати одночасно декілька потужних електропристроїв до однієї розетки, що може викликати надмірне нагрівання провідників, руйнування їхньої ізоляції, розплавлення і загоряння полімерних матеріалів;
- не залишати включені електроприлади без нагляду;
- 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;
- не ставити на електроприлади матеріали, які можуть під дією теплоти, що виділяється, загорітися (канцелярські товари, сувенірну продукцію тощо).

5.6 Охорона навколишнього природного середовища

5.6.1 Загальні дані з охорони навколишнього природного середовища

Діяльність за темою магістерської роботи в процесі її виконання впливає на навколишнє природне середовище і регламентується нормами діючого законодавства [39–43].

Основним екологічним аспектом в процесі діяльності за даними спеціальностями є процеси впливу на атмосферне повітря та процеси поводження з відходами, які утворюються, збираються, розміщуються, передаються на видалення (знешкодження), утилізацію, тощо в ІТ галузі.

Немає впливу на атмосферне повітря при нормальних умовах праці, бо в приміщенні не використовуються сканери, принтери та інші джерела викиду забруднюючих речовин в повітря робочої зони.

В процесі діяльності користувача виникають процеси поводження з відходами ІТ галузі. Види відходів, утворення, яких можливо:

- відпрацьовані люмінесцентні лампи – І клас небезпеки;

- батарейки та акумулятори (малі) –III клас небезпеки;
- змінні носії інформації – IV клас небезпеки;
- відпрацьований ізолюючий матеріал, дроти та кабелі – IV клас небезпеки;
- макулатура – IV клас небезпеки;
- побутові відходи – IV клас небезпеки.

5.6.2 Вимоги до збору, пакування та розміщення відходів ІТ галузі

Вимоги зберігання виявлених за своєю роботою відходів визначаються відповідно [40].

Відходи в міру їх накопичення збирають у тару, відповідну класу небезпеки, з дотриманням правил безпеки, після чого доставляють до місця тимчасового зберігання відходів відповідно до затвердженої схеми їх розміщення, зазначені для зберігання відходів місця чи об'єкти повинні використовуватися лише для заявлених відходів.

Не допускається зберігання відходів у невстановлених схемою місцях, а також перевищення норм тимчасового зберігання відходів.

Способи тимчасового зберігання відходів визначаються видом, агрегатним станом і класом небезпеки відходів:

–відходи I класу небезпеки зберігаються в герметичній тарі (сталеві бочки, контейнери). У міру наповнення тару з відходами закривають герметично сталевий кришкою;

–відходи II класу небезпеки в залежності від агрегатного стану зберігаються в поліетиленових мішках, бочках, сховищах та інших видах тари, яка запобігає поширенню шкідливих речовин;

–відходи III класу небезпеки зберігаються в тарі, яка забезпечує локалізацію зберігання, дозволяє виконувати вантажно–розвантажувальні і транспортні роботи і виключає поширення в ОС шкідливих речовин;

–відходи IV класу небезпеки можуть зберігатися відкрито на промисловому майданчику у вигляді конусоподібної купи, звідки їх автотранспортом перевантажують у самоскид і доставляють на місце утилізації або захоронення;

–в разі тимчасового зберігання відходів у стаціонарних складах або промислових приміщеннях повинні бути забезпечені санітарно–гігієнічними етичними вимогами до повітря робочої зони згідно [35].

Не допускається змішування відходів різних видів і класів небезпеки з будівельними і побутовими відходами, відходами дерев'яної, металевої, синтетичної тари, відходами текстильних матеріалів (старий спецодяг, ганчірки) та інше.

Проведення заготовки, здачі, переробки та реалізації металобрухту встановлені в [44].

Особливий контроль наділяється збору і зберіганню відпрацьованих ртутьвмісних ламп (енергоощадних) як відходам I класу небезпеки, що збираються і обов'язково передаються на утилізацію підприємствам, що мають ліцензію на поводження з такими небезпечними відходами.

Всі відходи, що утворюються в процесі діяльності/роботи, підлягають обліку.

5.6.3 Визначення впливу та заходів щодо поводження з відходами ІТ галузі

З метою визначення та прогнозування впливу відходів на навколишнє середовище, своєчасного виявлення негативних наслідків, їх запобігання відповідно [45] повинен здійснюватися моніторинг місць утворення, зберігання, і видалення відходів. Відомості про місце утворення та місце розташування відходів зазначаються на «План схемі місці розміщення відходів організації / виробництва» та наводяться у таблиці 5.5, а відомості про склад і властивості відходів, що утворюються, а також ступінь їх небезпечності для навколишнього природного середовища та здоров'я людини у таблиці 5.6.

Таблиця 5.5 Відомості про місце утворення та місце розташування відходів

Код та найменування відходів за ДК –005–96	Технологічний процес або виробництво, де утворюються відходи / клас небезпеки	Місце розташування відходу, тара та її кількість, місткість, розміри у разі наявності майданчиків розташування відходів необхідно зазначити тип покриття та наявність даху)	№ на схемі (додається масштабна схема місць розміщення відходів)
7710.3.1.26 Лампи люмінесцентні, та відходи, які містять ртуть, інші зіпсовані або відпрацьовані (Відпрацьовані ртутьвмісні люмінесцентні лампи)	1	буд.4, в приміщенні кладової S=100м ² , в кількість 20 од.	8401–ТХ

7720.3.1.01 Відходи комунальні (міські) змішані, у т.ч. сміття з урн (Побутові відходи)	4	зовнішній майданчик зберігання побутових відходів біля буд. 4 $S=5\text{м}^2$ $V=2,08\text{м}^3-2\text{од.}$	8401–ТХ
7710.3.1.01 Макулатура паперова та картонна (Макулатура)		буд. 4 4 поверх в кім. 412 $S=5,0\text{ м.}^2$	8401–ТХ
7730.3.1.02 Матеріали пакувальні пластмасові зіпсовані, відпрацьовані чи забруднені (Матеріали пакувальні забруднені)	4	буд. 4 контейнер $V=0,9\text{м}^3$ (3 од.)	8401–ТХ
Змінні носії інформації	4	контейнер $V=0,04\text{м}^3$ (2 од.) буд. 4	8401–ТХ
Батарейки та акумулятори (малі)	3	контейнер $V=0,09\text{м}^3$ (4 од.) буд. 4	8401–ТХ

Таблиця 5.6 – Відомості про склад і властивості відходів, що утворюються, а також ступінь їх небезпечності для навколишнього природного середовища та здоров'я людини.

Назва відходів	Клас небезпечності	Хімічний (у долях відсотків складників або інших одиницях виміру) та морфологічний склад	Фізико–хімічні властивості
Відпрацьовані люмінесцентні лампи	I	Ртуть – 0,013 Hg Скло – 98,787(Na , K) ₂ O 2SiO₂ Алюміній – 1,2 Al	Ртуть – $T_{\text{кип.}} = 356,58^{\circ}\text{C}$ $T_{\text{плав.}} = -38,87^{\circ}\text{C}$ Скло – $T_{\text{плав.}} = 800^{\circ}\text{C}$ Алюміній – $T_{\text{кип.}} = 2348^{\circ}\text{C}$ $T_{\text{плав.}} = 660,1^{\circ}\text{C}$
Макулатура	IV	Цинк – 0,000053 – 0,000056 Zn Свинець – 0,000049 – 0,000051 Pb Хром – 0,000051 – 0,000054 Cr Мідь	Уривки та обрізки з паперових мішків Цинк $T_{\text{кип.}} = 913^{\circ}\text{C}$ $T_{\text{плав.}} = 4,19^{\circ}\text{C}$ Свинець $T_{\text{кип.}} = 1751^{\circ}\text{C}$ $T_{\text{плав.}} = 327,3^{\circ}\text{C}$ Хром $T_{\text{кип.}} = 1890^{\circ}\text{C}$

		– 0,000033 – 0,000035 Cu Целюлоза – 97,299814 – 96,999804 $(C_6H_{10}O_5)_n$ Вода – 2,7 – 3,0	$T_{\text{плав.}} = 2480^{\circ}\text{C}$ Мідь $T_{\text{кип.}} = 2580^{\circ}\text{C}$ $T_{\text{плав.}} = 1083^{\circ}\text{C}$ Целюлоза $T_{\text{возг. с обуглив.}} \geq 100^{\circ}\text{C}$
Побутові відходи	IV	Побутові відходи – 100 – 100, в т. ч.: Папір – 30 – 17; $[(C_6H_{10}O_5)_n$ – целюлоза] Поліетилен – 20 – 24; $(-CH_2 - CH_2 -)_n$ Деревина – 5 – 3; $[(C_6H_{10}O_5)$ – целюлоза, лігнін] Матеріали текстильні – 4 – 3; $[(C_6H_{10}O_5)_n$ – целюлоза Мінеральні домішки (пісок, глина) – 4 – 9 Харчові відходи – 37 – 44;	Поліетилен – $T_{\text{размяг.}} \geq 150^{\circ}\text{C}$ Твердий матеріал рослинного походження, не розчиняється у воді. Целюлоза, лігнін $T_{\text{возг. с обуглив.}} \geq 120^{\circ}\text{C}$ Харчові відходи $T_{\text{биоразл.}} \geq 4^{\circ}\text{C}$

Негативний вплив на ОС і людини визначається його хімічним складом.

Ртуть. У природних водах міститься в концентрації 0,00003 ... 0,0028 мг / л. Являючись потужним кумулятивним отрутою, з можливою канцерогенною і мутагенною дією. Процеси самоочищення водою порушують концентрація ртуті понад 0,018 мг / л, порогова концентрація ртуті за впливом на санітарний режим водою – 0,01 мг / л. Наприкінці концентрація понад 0,03 є токсичною практично для всіх видів водних організмів. Надзвичайно токсична при попаданні з питною водою для теплокровних організмів, надходження ртуті з питною водою в кількості 75,0 ... 300,0 мг / доб. є смертельним. Відрізняється високою токсичністю для будь-яких форм життя. При отруєнні парами спостерігається слабкість, головний біль, біль в шлунку, роздратування по-чек, навіть нефрит; катаральні явища. Розвивається тремтіння рук, ніг, всього тіла. Виникає стан підвищеної психічної збудливості/ Пари ртуті проявляють нейротоксичність, особливо страждають вищі відділи нервової системи [46].

Скло. Нетоксичні, безпечно в навколишньому середовищу, не шкідлива в нирках і водоймах. Вдихання скляного пилу (волокон) призводить до силікоз в зв'язку з високим вмістом сполук кремнію. Шкідливої дії не робить, але є небезпека механічних пошкоджень (порізи, травми).

Алюміній. Токсичний для водної біоти, теплокровних тварин і людей, в концентрації > 1 мг / л чинить негативний вплив на зростання с / г культур. У концентрації > 1 мг / л гальмує зростання мікрофлори водойм і стримує процеси самоочищення водойм. Рівень токсичності визначається формою, в якій знаходиться елемент. Впливає на обмін речовин і функції нервової системи . При попаданні на ґрунт, в воду і атмосферними повітря надає негативного впливу на НС і здоров'я людини.

Цинк. Малотоксичний для теплокровних тварин при надходженні з їжею і питної водою—концентрація в питній воді 11,2 ... 26,6 мг / л переноситься без будь-яких ознак інтоксикації. Дуже корисний для флори, будучи одним з найважливіших мікроелементів харчування, однак лише в концентрації до 0,2 мг / л, крім того, елемент силється до кумуляції в грантах. Дуже токсичний для водних організмів, порушуючи процеси самоочищення водойм і стаючи токсичним для іхтіофауни в концентрації 0,15 ... 5,0 мг / л. Мутагенна і онкогенна небезпеку.

Свинець. У природних водах міститься в концентрації 0,001 – 0,023 мг / л. У концентрації 2,0 мг / л надає воді металевий присмак. Можливо має мутагенну і канцерогенну дію, значно збільшує токсичну дію інших металів. В концентрації 1,90 мг / л згубно діє на дафній, концентрація 0,1 мг / л погіршує процеси самоочищення водойм. Свинець токсичний для рослин в концентрації понад 5,0 мг / кг ґрунту.

Помірно токсичний. Викликає хронічне отруєння. Має здатність вражати центральну і периферичну нервову систему, кістковий мозок і кров, судини, синтез білка, генетичний апарат клітини.

Хром. Міститься в природних водах в концентрації 0,001 ... 0,112 мг / л. LK50 Сг (VI) для риби—30,0 ... 50,0 мг / л, LK50 Сг (III) для риби – 117,0 мг / л. Низькі концентрації хрому позитивно впливають на ріст рослин. Володіє канцерогенними властивістю.

Мідь. У природних водах міститься в концентраціях 0,001...0,98 мг/л. У концентрації 0,5 мг/л забарвлює воду, в концентрації $> 1,0$ мг / л—помітно збільшує мутність води. Дуже токсична як для водних організмів, так і для рослин. У концентрації 0,001 мг / л гальмує розвиток синьо зелених водоростей, LK50 практично для всіх видів риби становить 0,18 ... 1,35 мг / л (короп, карась, окунь, щука, сом). Накопичується ґрунтом і рослинами. У концентрації 0,1 ... 0,2 мг / л надає токсичну дію на ріст рослин. Високотоксичний метал. Викликає гостре отруєння, має широкий спектр токсичної дії).

Целюлоза. Нетоксична. Досить легко піддається біодеструкції лігнін – і целюлозоруйнучими бактеріями і деякими класами нищих грибів. У зв'язку з нетоксичністю LD50 для тваринах не установлена. Токсичність визначається за вмістом

важких металів, здатних мігрувати з неї в навколишнє середовище. При попаданні на ґрунт, в воду і атмосферне повітря чинить негативний вплив на ОС і здоров'я людини.

Поліетилен. Нетоксичний для всіх видів флори і фауни в зв'язку з дуже високою біологічною інертністю. Нерозчинний у водних середовищах і не впливає на санітарний режим водойм. Використання його не вимагає запобіжних заходів. Отруєння можливі при виробництві та переробці плівки, в результаті виділення окису вуглецю, альдегідів, органічних кислот [47]

Деревина. Нетоксична. Досить легко піддається біодеструкції лігнін- і целюлозоруйнуючими бактеріями і деякими класами нижчих грибів. У зв'язку з нетоксичністю LD50 для тварин не встановлена. Деревина нетоксична при використанні. Але дія деревного пилу при рубці і переробці деревини викликає захворювання дихальних шляхів і шкіри.

Текстильне волокно. Нетоксична в зв'язку з біогенним походженням, проте для біодеструкції необхідна наявність вологи. Нетоксична при використанні. Токсична дія виникає (як результат механічної дії – наслідок пилу) при виробництві тканин і при переробці вторинних матеріалів; слабкий алерген.

5.7 Висновки до розділу

У четвертому розділі магістерської роботи проведений аналіз умов праці, шкідливих та небезпечних чинників, з якими стикається робітник. Визначено параметри і певні характеристики приміщення для роботи над запропонованим проектом написаному в дипломній 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. Наведена схема, розміри приміщення та визначені 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, визначені основні екологічні аспекти впливу на навколишнє природне середовище та зазначені заходи щодо поводження з ними.

ВИСНОВКИ

Метою магістерської роботи було дослідження існуючих систем, придатних для реалізації інтернет-аукціону, проблем і недоліків у вже використовуваних системах.

1 Актуальність проблеми та недоліків у запропонованих і використовуваних системах для реалізації проекту інтернет-аукціону.

2 На даний момент в безкоштовних рішеннях відсутня оптимальна система управління контентом для високонавантажених проектів.

3. Проведено аналіз серед найпопулярніших систем управління контентом, перелічені їх недоліки та обґрунтована необхідність дослідження і розробки іншої системи управління для поставленого завдання.

4 Досліджено основні проблеми систем CMS і на основі проведеного аналізу були запропоновані шляхи їх вирішення та реалізації.

5 Розроблене програмне забезпечення може бути використане для створення проектів, пов'язаних з комерцією як готова база для торговельного майданчика, що вимагає мінімальних налаштувань і втручання в розроблену систему. Для повноцінної роботи потрібне створення бази необхідних для кінцевого користувача товарів.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. <http://www.php.net/learnphp/phpoo/> - PHP і об'єктно-орієнтоване програмування.
2. <https://www.mysql.com/> - Довідник розробника баз даних MySQL.
3. <http://php.net/> - Довідник розробника на мові PHP.
4. <http://www.yiiframework.com/> - Довідник розробника та вихідний код для Yii2 Framework.
5. Книга "Model View Controller (MVC) is becoming the definitive architecture of websites. Автор Chris Pitt. Рік випуску 2012.
6. <http://learn.javascript.ru/> - Підручник JavaScript.
7. <https://angularjs.org/> - Документація по використанню JavaScript framework AngularJS.
8. <https://getbootstrap.com/> - Документація і вихідний код для CSS framework Bootstrap.

ДОДАТОК А

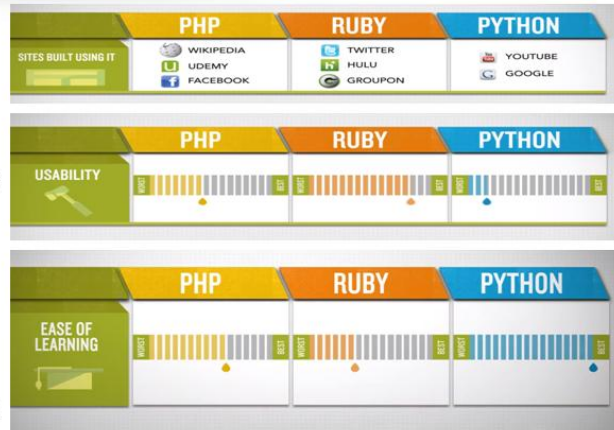
Тема

Дослідження та реалізація комп'ютерної
системи інтернет-аукціону



Порівняння мов

На сьогоднішній день найпопулярнішими мовами web-програмування є PHP, Python, Ruby. Саме ці три скриптових серверних мови програмування найчастіше вибирають для розробки серверних додатків.



fppt.com

Порівняння Framework

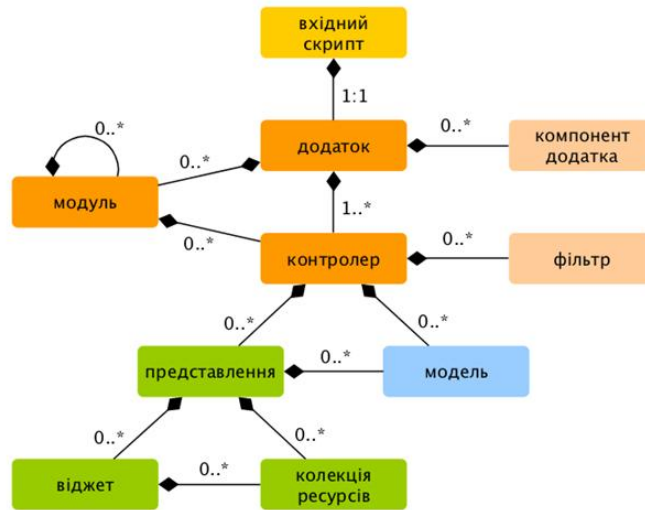
Найчастіше використовуються системи управління контентом (CMS), або фреймворки.

Критерій	Laravel	Symfony 2	Nette	CodeIgniter	Yii 2
Опис	MVC, підтримка модулів, дуже багато методів коробки «ActiveRecord»	MVC, Twig, Doctrine, бандали	Швидкий, вбудовані макроси в шаблонизатор, технологія Context-Aware Escaping	MVC, швидкий	MVC, ActiveRecord, DAO, велика база модулів, кешування даних.
Необхідний рівень знань	PHP, ООП	PHP5, ООП, консоль, ORM	PHP5, ООП	PHP5, ООП	PHP, ООП
Проекти	Всі типи	Великі	Невеликі, середні	Невеликі, середні	Всі типи
PHP 5.4	Підтримує	Піддержує	Немає	Немає	Підтримує
Жорстка структура каталогів	Немає	Так	Так	Так	Немає
Критерій	Laravel	Symfony 2	Nette	CodeIgniter	Yii 2
Офіційна підтримка інтернаціоналізації	Присутня	Присутня	Немає	Присутня	Присутня
Повна підтримка ORM	Присутня	Присутня	Немає	Присутня	Присутня
Документація і приклади	Гарна	Гарна	Нормальна	Гарна	Гарна
Unit-тести для вихідного коду фреймворка	Так	Так	Немає	Немає	Так



fppt.com

Внутрішній устрій Yii



Життєвий цикл запиту

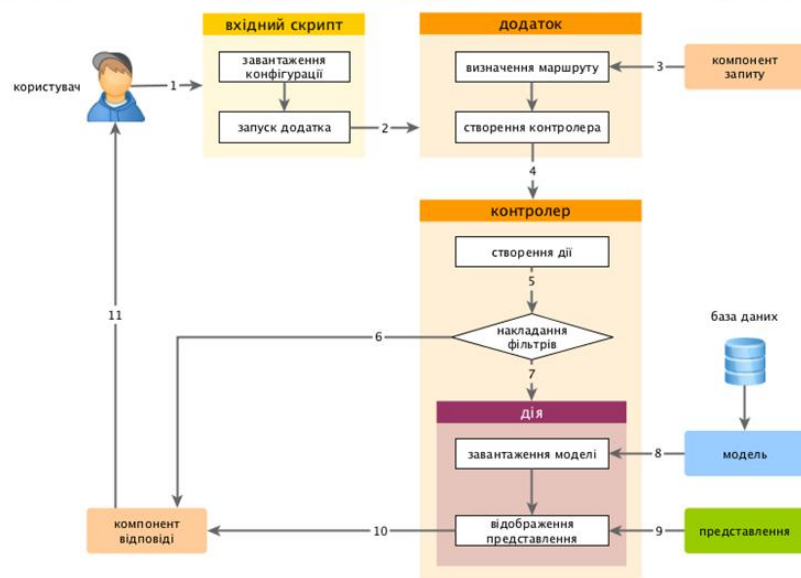
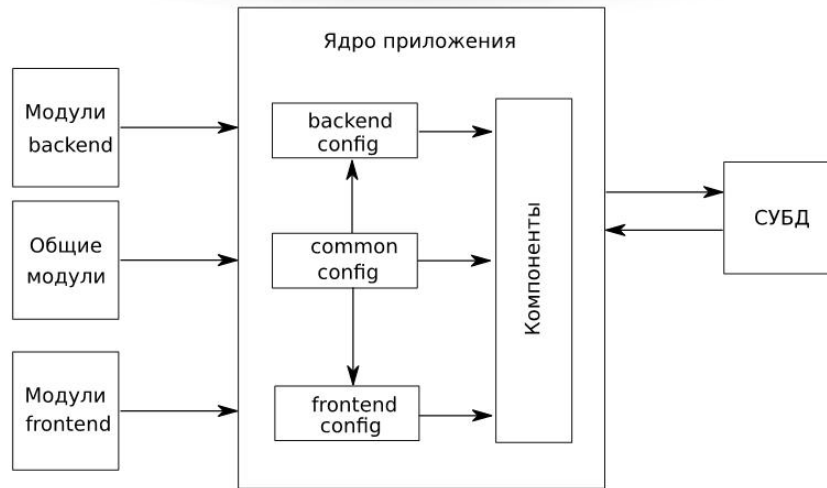
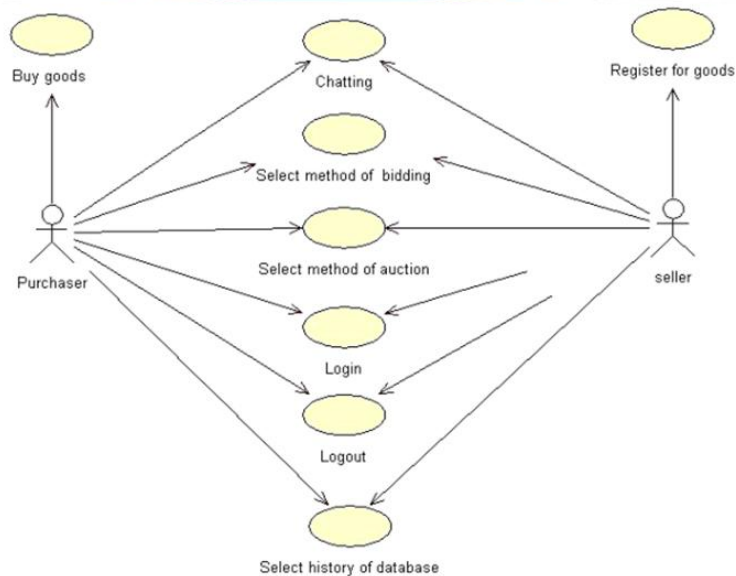


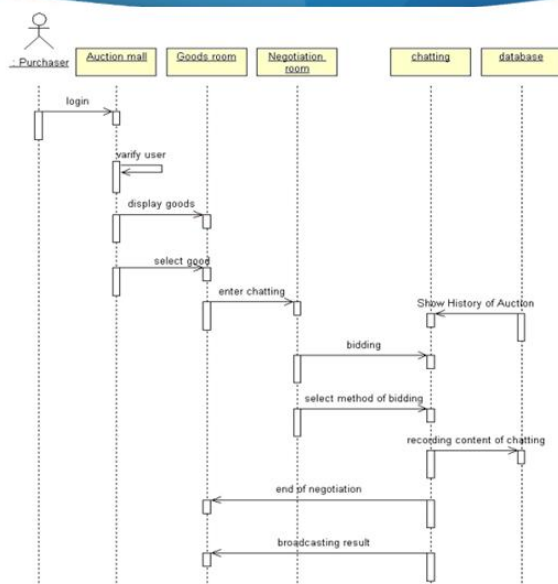
Схема конфігурації програми



Діаграма «Використання справи»

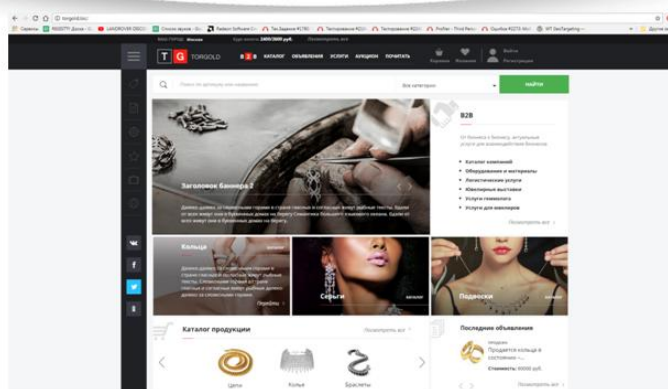


Поведінка покупця

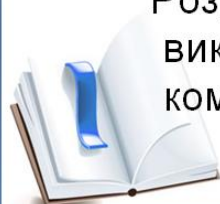


fppt.com

Розроблений проект



Розроблене програмне забезпечення може бути використане для створення проектів, пов'язаних з комерцією



fppt.com