

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Тернопільський національний економічний університет
Факультет комп'ютерних інформаційних технологій
Кафедра комп'ютерних наук

ТЕПЛЮК Михайло Сергійович

Веб-орієнтована програмна система державного підприємства "Клеванське лісове господарство"/
Web-based software system of state enterprise
"Klevanske forestry"

напрямок підготовки: 6.050103 - Програмна інженерія
фахове спрямування - Програмне забезпечення систем

Бакалаврська дипломна робота

Виконав студент групи ПЗС-42
М. С. Теплюк

Науковий керівник:
викладач КРЕПИЧ С.Я.

Бакалаврську дипломну роботу
допущено до захисту:

"__" _____ 20__ р.

Завідувач кафедри
_____ **А. В. Пукас**

РЕЗЮМЕ

Дипломна робота містить 89 сторінки, 15 таблиць, 33 рисунків, список використаних джерел із 20 найменувань та 3 додатка.

Метою дипломної роботи є дослідження потенціалу та певних виробничих характеристик державного підприємства «Клеванське лісове господарство», ринку збуту та розробка, на основі отриманих даних, програмної системи реалізації продукції підприємства.

Об'єктом дослідження процеси є роботи державного підприємства «Клеванське лісове господарство».

Предметом дослідження є застосування сучасних технологій для розробки веб-орієнтованої програмної системи державного підприємства «Клеванське лісове господарство».

Методи розробки базуються на мові програмування PHP , бази даних MySQL.

Одержані результати полягають в розробці веб-орієнтованої програмної системи «Клеванське лісове господарство».

Ключові слова: замовлення, товар, продукція, база даних, діаграма класів, реляційна модель бази даних, тестування.

SUMMARY

Thesis contains 89 pages, 15 tables, 33 figures, list of sources with 20 titles and 3 application.

The aim of the thesis is the study of certain production capacity and characteristics of the state enterprise "Klevanske forestry", the market and development on the basis of the data software system product sales company.

The object of research is the process of registration and conduct of the case of patients in the dental office.

The subject of research is the use of modern technology for developing web-based software system state enterprise "Klevanske forestry."

The methods of development are basing on based on in PHP, database MySQL.

The results are in development of a web-based software system "Klevanske forestry."

Keywords: order, product, products, database, diagram classes relational model database testing.

ЗМІСТ

ВСТУП	9
РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	11
1.1 Коротка характеристика об'єкту управління	11
1.2 Опис предметної області	13
1.3 Огляд та аналіз існуючих аналогів.....	17
1.4 Специфікація вимог до програмного продукту	25
РОЗДІЛ 2 ПРОЕКТУВАННЯ WEB-СИСТЕМИ.....	36
2.1 Розробка програмної архітектури системи	36
2.2 Проектування структури бази даних	39
РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ	47
3.1 Програмна реалізація проекту	47
3.2 Програмна реалізація бази даних	52
РОЗДІЛ 4 ТЕСТУВАННЯ ТА ДОСЛІДНА ЕКСПЛУАТАЦІЯ.....	57
4.1 Тестування	57
4.2 Розгортання програмного продукту	58
4.3 Інструкція користувача	60
ВИСНОВКИ	64
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	65
ДОДАТОК А ФУНКЦІОНАЛЬНЕ ТЕСТУВАННЯ.....	66
ДОДАТОК Б КОД UNIT-ТЕСТІВ	68
ДОДАТОК В ВИХІДНИЙ КОД ПРОГРАМНОГО ПРОДУКТУ	70

ВСТУП

На державному підприємстві «Клеванське лісове господарство» основними напрямками розвитку лісового господарства визначено зокрема підвищення продуктивності, поліпшення якісного складу лісових насаджень, розширення застосування методів раціонального використання лісових ресурсів. Від того, як організована система реалізації продукції, залежить безперервність виробничого процесу. Від того, як продається продукція, залежить обсяг виробництва і попит на ринку. Несвоєчасна реалізація негативно впливає на роботу споживачів продукції і, як результат, погіршення партнерських стосунків. Тому актуальним для підприємства є розробка програмного продукту, який надасть змогу створити зв'язок між виробником і споживачами, що, в свою чергу, призведе до збільшення обсягів виробництва та потенціалу підприємства.

Метою роботи є дослідження потенціалу та певних виробничих характеристик державного підприємства «Клеванське лісове господарство», ринку збуту, розробка на основі отриманих даних, програмної системи реалізації продукції підприємства.

Для дослідження мети необхідно:

1. Проаналізувати роботу державного підприємства «Клеванське лісове господарство», а також, схожі аналоги. Виділити переваги та недоліки програмного продукту
2. Розробити вимоги до системи на основі проведеного аналізу.
3. Спроекувати базу даних підприємства.
4. Розробити програмну систему.

Розробку веб-орієнтованої програмної системи планую розділити на декілька етапів:

1. Розробка інтерфейсу головної сторінки та меню сайту
2. Розробка модуля функцій користувача (авторизація, реєстрація та перегляд інформації на сайті).

3. Розробка модуля функцій адміністратора (додавання товару, видалення товару, оновлення товару, формування звітності).
4. Розробка модуля функцій авторизованного користувача (замовлення товару).

Об'єктом дослідження є процеси роботи державного підприємства «Клеванське лісове господарство» .

Предметом дослідження є застосування сучасних технологій для розробки веб-орієнтованої програмної системи державного підприємства «Клеванське лісове господарство».

Під час створення системи використовуватимуть наступні технології:

1. PHP – гіпертекстовий препроцесор.
2. MySQL – реляційна система управління базами даних

Практична цінність розробки веб-орієнтованої програмної системи державного підприємства «Клеванське лісове господарство» полягає у покращенні діяльності підприємства шляхом залучення нових споживачів продукції та розширення ринку збуту продукції.

РОЗДІЛ I

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА СПЕЦИФІКАЦІЇ ВИМОГ

1.1 Коротка характеристика об'єкту управління

Вирішення кожної проблеми завжди полягає в усвідомленні та визначенні труднощів або наявних можливостей. Виявлення таких ознак допомагає визначити проблему загалом. Щоб з'ясувати причини виникнення проблеми потрібно зібрати та здійснити аналіз необхідної внутрішньої та зовнішньої інформації. Рівень розвитку підприємств визначається в першу чергу трьома, однаково важливими показниками їхньої діяльності [1]:

- продуктивністю праці;
- конкурентоспроможністю продукції;
- якістю росту і розвитку, визначеними ступенем інноваційної технічної бази.

Збір вхідного та вихідного матеріалу проводилось на основі аналізу предметної області, використання даних певної ланки діяльності підприємства та вивченням необхідної інформації про продукцію ДП «Клеванське лісове господарство». При визначенні форми і змісту вихідної інформації потрібно забезпечити більш коректне та швидке сприйняття як графічної, так і цифрової інформації. Спосіб представлення інформації істотно впливає на доречність її інтерпретації. Невдало подана на екрані дисплея або роздрукована на принтері інформація може призвести до помилкових висновків про можливий отриманий результат. Аналіз організації документообігу передбачає раціональний рух документів, включаючи переміщення в апараті управління документів, їх отримання, оформлення, відправку, розгляд та операції з ними. Структурне вивчення виробництва в апаратах управління різних рівнів в значній частині було присвячено проблемам організації руху документів.

Організація підприємства представлена на рисунку 1.1.

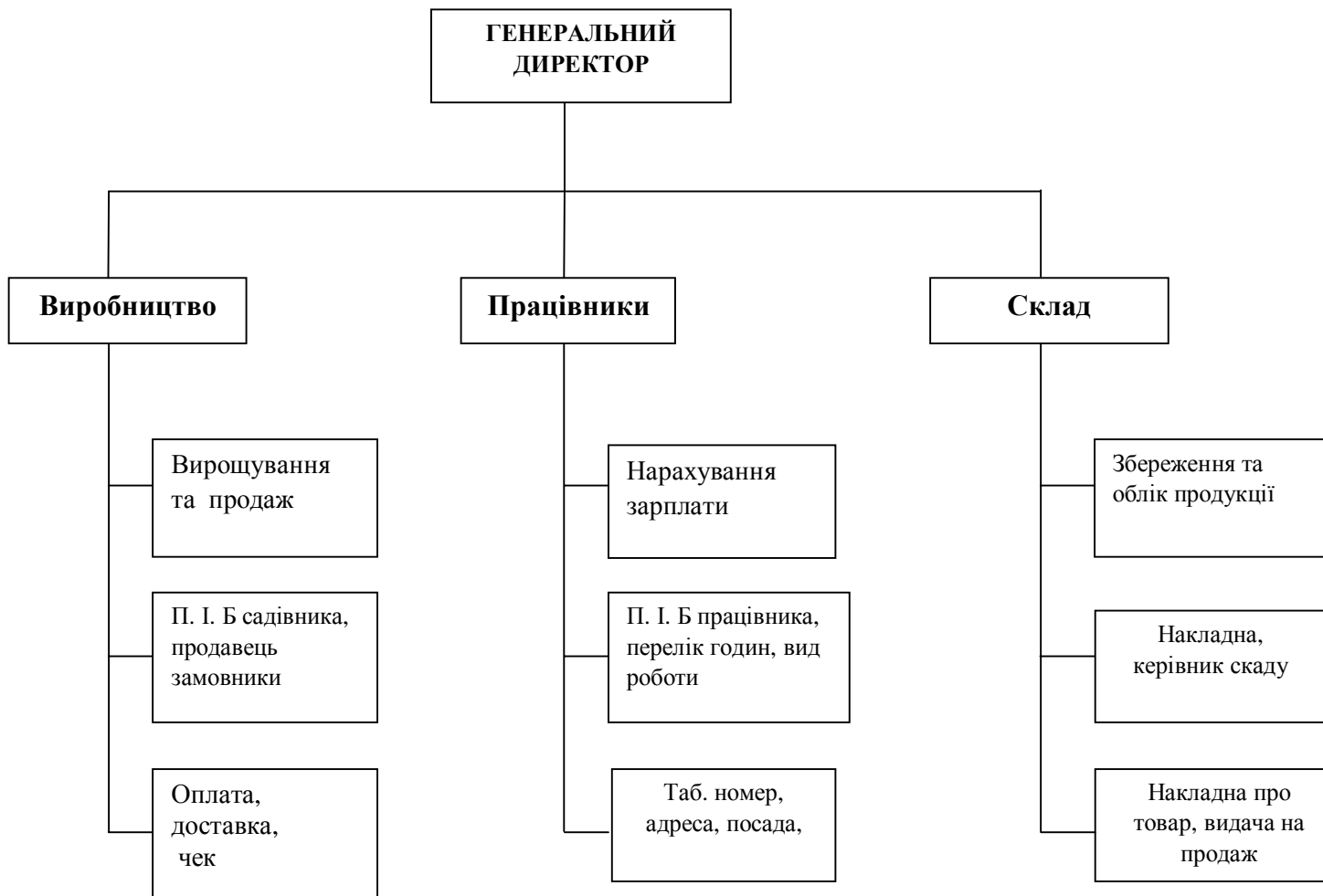


Рисунок 1.1 – Організація підприємства

Документообіг на складському відділі налічує журнал, в якому записується назва товару, кількість, прізвище отримувача, ціна. У відділі заготівлі товару формується документ, в якому вказується кількість і що буде направлено на склад підприємства. У відділі вирощування деревини відбувається аналіз необхідності продукції і, на основі цього, відбувається насадження нових посадок. Звітною документацією підрозділу є накладна про товар, де вказано його вид, ціну та кількість. Певні документи завжди зберігаються у захищеній базі даних підприємства [2]. Відбувається моніторинг документування та збереження архівних копій на певних засобах збереження інформації. На основі цих вихідних даних формується статистика виробленої продукції на підприємстві, в результаті чого стають зрозумілі

досягнення підприємства та в якому напрямку слід працювати, визначення альтернатив. На цьому етапі бажано виявити всі можливі дії, які б усунули причини і проблеми, а отже, дати змогу підприємству досягти своїх цілей.

1.2 Опис предметної області

Основними напрямками розвитку ДП «Клеванське лісове господарство» визначено

- підвищення продуктивності,
- поліпшення якісного складу лісових насаджень,
- розширення застосування методів раціонального використання лісових ресурсів.

Подальший розвиток галузей лісової промисловості потребує повного радикального реформування, що включає в себе заміну фізично та морально застарілого обладнання, розробку ефективних раціональних технологій [3].

На рисунку 1.2. представлено головну діаграму процесів системи.

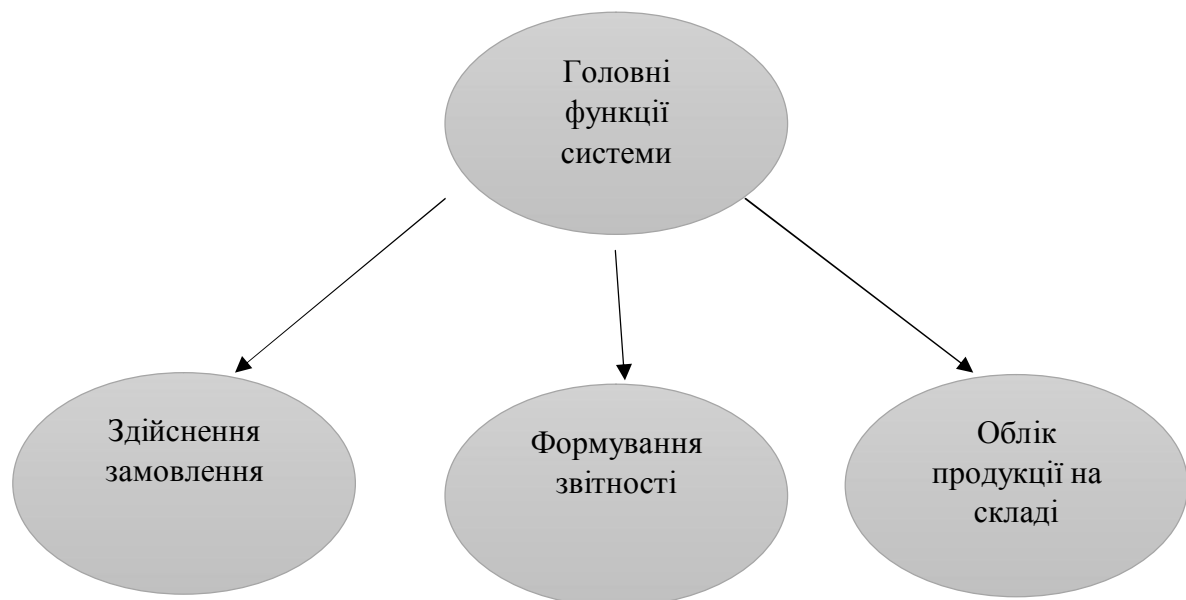


Рисунок 1.2 – Головні функції системи

Процес здійснення замовлення є найголовнішим процесом функціонування підприємства і, відповідно, основною функцією розроблюваного веб-сайту.

На рисунку 1.2 зображено процес реалізації здійснення замовлення.

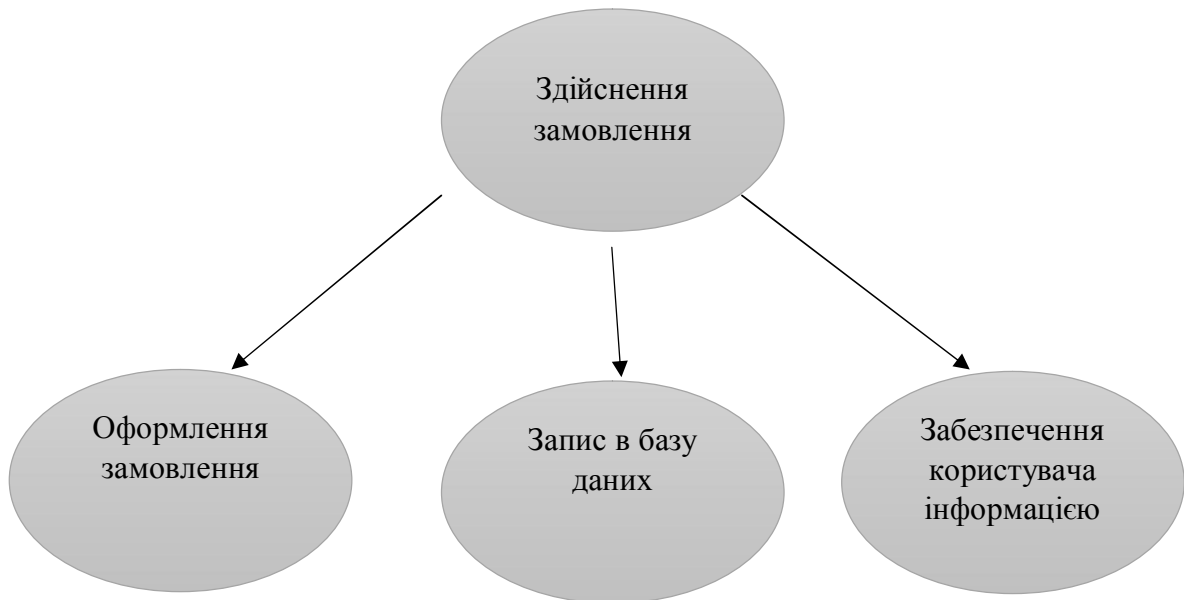


Рисунок 1.2 – Процес реалізації здійснення замовлення

Щоб здійснити замовлення у системі потрібно виконати наступне:

- проаналізувати кількість матеріалу на складі;
- проаналізувати оплати клієнтів за реалізовану продукцію;
- зазвітувати дані по продукції;
- оформити замовлення товару;
- підготувати дані для виписки чеку.

Характеристику здійснення замовлення наведено в таблиці 1.1.

Таблиця 1.1

Характеристика бізнес-процесу здійснення замовлення

Назва характеристики	Значення характеристики
Ім'я бізнес-процесу	Здійснення замовлення
Основні учасники	Користувач

Вхідна подія	Запит на здійснення замовлення
--------------	--------------------------------

Продовження таблиці 1.1

Вхідні документи	Дані про товар
Вихідна подія	Повідомлення про успішне здійснення замовлення
Вихідні документи	Підтвердження замовлення
Клієнт бізнес-процесу	Формування звітності, Облік продукції

Не менш важливим завданням підприємства є формування звітності системи. На рисунку 1.3 зображено діаграму формування звітності системи

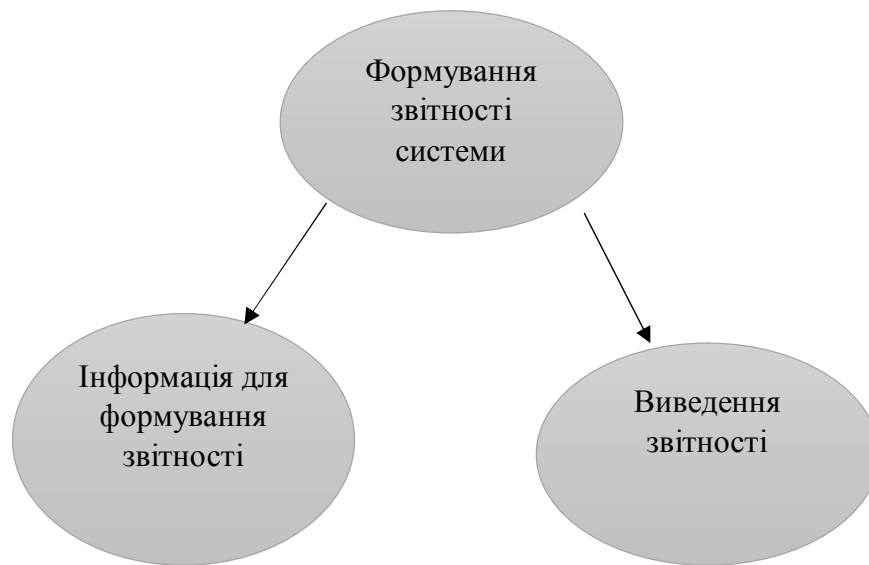


Рисунок 1.3 – Формування звітності системи

Для формування звітності системи на підприємстві обирається інформація для звітності та виведення.

Характеристика оформлення звітності наведено в таблиці 1.2.

Таблиця 1.2

Характеристика бізнес-процесу формування звітності

Назва характеристики	Значення характеристики
----------------------	-------------------------

Ім'я бізнес-процесу	Формування звітності системи
---------------------	------------------------------

Продовження таблиці 1.2

Основні учасники	Адміністратор
Вхідна подія	Запит на оформлення замовлення
Вхідні документи	Дані про товар
Вихідна подія	Повідомлення про успішне виведення звіту
Вихідні документи	Оформлений звіт
Клієнт бізнес-процесу	Формування звітності

Також функцією системи є облік продукції, що зберігається на складі. На рисунку 1.4 зображено діаграму обліку товару на складі

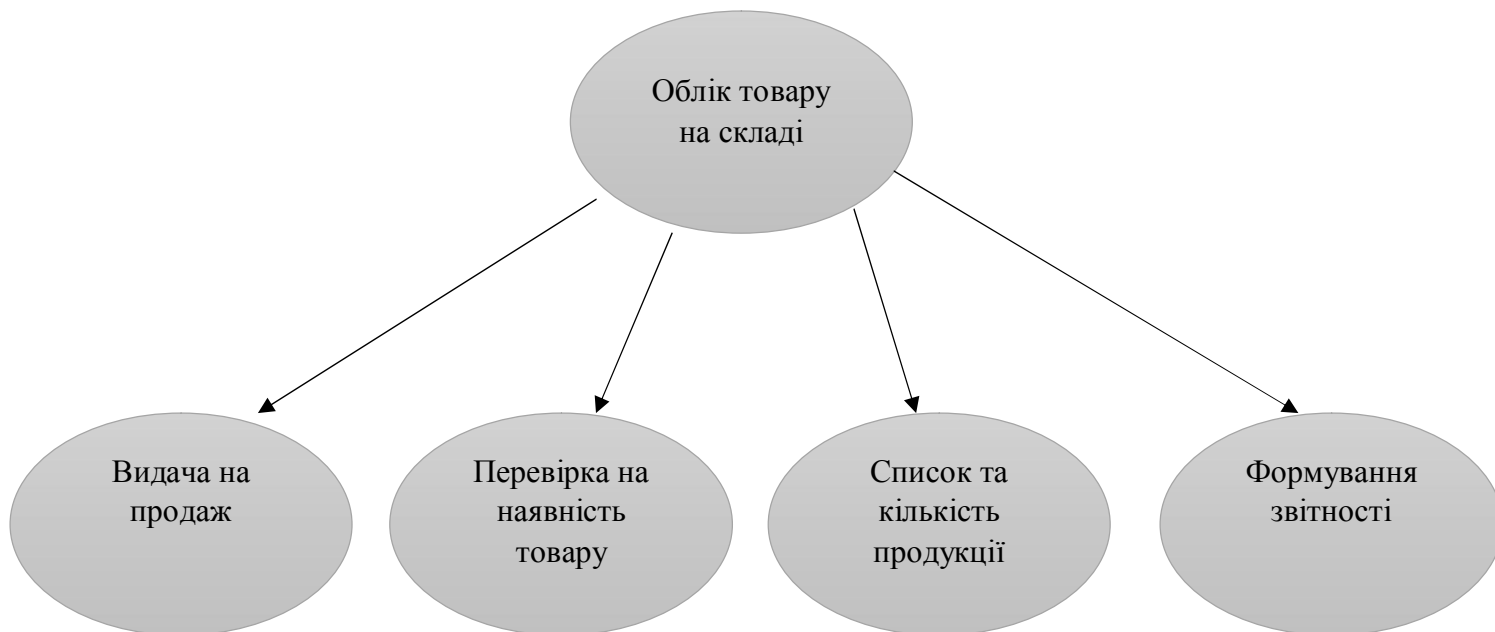


Рисунок 1.4 – Облік товару на складі

Складський підрозділ займається зберіганням продукції, а також організацією її транспортування, обліком отриманої на склад продукції та забезпеченням просування товару на продаж, організація конкурентноспроможності, реклама, замовлення продукції [4].

Облік товару на складі поділяється на такі частини:

- Видача на продаж;
- перевірка наявності товару ;
- список та кількість продукції;
- формування звітності.

Назва характеристики	Облік товару на складі
Ім'я бізнес-процесу	Здійснення обліку товару
Основні учасники	Адміністратор
Вхідна подія	Запит на видалення, додавання, наявність товару
Вхідні документи	Дані про товар
Вихідна подія	Повідомлення про успішний запит
Вихідні документи	Підтвердження запиту
Клієнт бізнес-процесу	Керування адміністратором

1.3. Огляд і аналіз існуючих аналогів

Результативність роботи будь-якого підприємства в наш час багато в чому залежить від вибору програмного забезпечення. З розвитком новітніх технологій на підприємствах виникла конкурентність не лише у якості виготовлення товарів, але, як не дивно, в розробці привабливого програмного забезпечення. Зручне програмне забезпечення надає можливість привабити більшу клієнтську аудиторію і, як результат, збільшити продуктивність підприємства[5].

Перед початком розробки нового програмного продукту доцільним і необхідним є огляд вже існуючих аналогів [6]. Порівняльний аналіз проведемо на прикладі веб-сайтів Хмельницького, Чернівецького і Івано-Франківського обласних управлінь лісового та мисливського господарства.

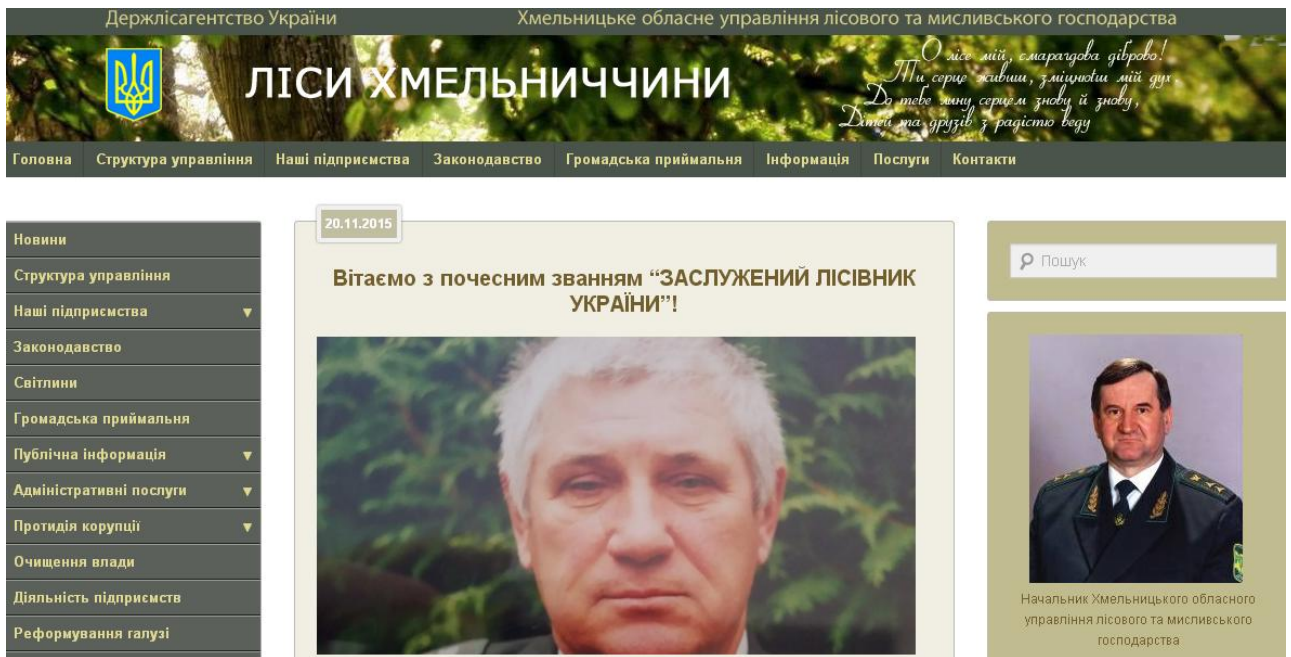


Рисунок 1.5. – Веб-сайт «Ліси Хмельниччини»

На рисунку 1.5 представлена головна сторінка веб-сайту Хмельницького обласного управління лісового та мисливського господарства. Інтерфейс сайту приємний у виваженних зелено-сірих тонах, що якнайкраще підкреслює тематику сайту. Меню продубльовано, що є недоречним та надлишковим, адже ці частини не унікальні.

На рисунку 1.6 зображено веб-сторінку «Структура управління», на якій текстом описується організація підприємства, керівники, відділи тощо, а також передбачена можливість залишити коментар. Інформація на веб-сторінці надана вичерпно, однак невдало підібраний колір тексту.

Держлісагентство України Хмельницьке обласне управління лісового та мисливського господарства

ЛІСИ ХМЕЛЬНИЧЧИНИ

*О лісе мій, смарагдова діброво!
Ти серце живиши, зміцнюєш мій дух
До тебе йду серцем зовбу й зовбу,
Дітей на друзів з радістю вору*

Головна Структура управління Наші підприємства Законодавство Громадська приймальня Інформація Послуги Контакти

Новини

Структура управління

Наші підприємства

Законодавство

Світлина

Громадська приймальня

Публічна інформація

Адміністративні послуги

Протидія корупції

Очищення влади

Діяльність підприємств

Реформування галузі

Контакти

9.09.2014


Структура управління

Обласне управління лісового та мисливського господарства підпорядковується Державному агентству лісових ресурсів України та є його територіальним органом. Основним завданням Управління є участь у сприянні формування та реалізації державної політики в сфері лісових відносин та ведення лісового і мисливського господарства та поповнення на території області.

У своїй діяльності управління керується Конституцією України, Лісовим Кодексом України, законами України, постановами Верховної Ради України, актами Президента України і Кабінету Міністрів України, наказами Міністерства охорони навколишнього природного середовища, Положенням про державну лісову охорону, нормативно-правовими актами Держлісагентства, а також Положенням про його територіальний орган.

Діяльність обласного управління лісового та мисливського господарства та його підвідомчих підприємств спрямовується на збереження лісів, нарощування їх ресурсного потенціалу, посилення ролі державної лісової охорони, недопущення розвитку процесів споживацького ставлення до лісів. Лісові підприємства і їх структурні підрозділи, як правило, розташовані у селах та селищах і є основними роботодавцями у місцях, де людям важко знайти роботу.

Пошук



Начальник Хмельницького обласного управління лісового та мисливського господарства
Лісовий

Рисунок 1.6. — Веб-сторінка «Структура підприємства»

На рисунку 1.7 зображена веб-сторінка «Громадської приймальні» на якій представлений графік особистого прийому громадян керівництвом підприємства. Як ми бачимо користувачу, який потребуватиме вирішення певних питань, доведеться декілька разів приїздити на підприємство, адже кожний з керівників приймає лише в окремі дні, а можливості онлайн запису на прийом не передбачено.

Головна Структура управління Наші підприємства Законодавство Громадська приймальня Інформація Послуги Контакти

Новини

Структура управління

Наші підприємства

Законодавство

Світлина

Громадська приймальня

Публічна інформація

Адміністративні послуги

Протидія корупції

Очищення влади

Діяльність підприємств

Реформування галузі

Контакти

ТЕЛЕФОН ГОРЯЧОЇ ЛІНІЇ (0382) 78 81 55

10.09.2014


Громадська приймальня

ГРАФІК особистого прийому громадян керівництвом Хмельницького обласного управління лісового та мисливського господарства

НАЧАЛЬНИК УПРАВЛІННЯ	ЛІСОВИЙ ВІТАЛІЙ МИТРОФАНОВИЧ	понеділок з 14.00 до 17.00 год.
ПЕРШИЙ ЗАСТУПНИК НАЧАЛЬНИКА УПРАВЛІННЯ	ХОПТИНЕЦЬ ВОЛОДИМИР МИХАЙЛОВИЧ	вівторок з 10.00 до 17.00 год.
ЗАСТУПНИК НАЧАЛЬНИКА	КОЛЕСНИК ВОЛОДИМИР ГРИГОРОВИЧ	середа з 10.00 до 17.00 год.
ЗАСТУПНИК НАЧАЛЬНИКА	СЕМЕНЕЦЬ ОЛЕКСАНДР ПЕТРОВИЧ	четвер з 10.00 до 17.00 год.
ЗАВДУВАЧ СЕКТОРА КАДРІВ ТА ЮРИДИЧНОГО ЗАБЕЗПЕЧЕННЯ	ВТЕРКОВСЬКИЙ ЯРОСЛАВ ЮРІЙОВИЧ	п'ятниця з 10.00 до 16.00 год.

Громадська приймальня | Додати коментар

Пошук



Начальник Хмельницького обласного управління лісового та мисливського господарства
Лісовий Віталій Митрофанович
Шановні відвідувачі!

Рисунок — 1.7. Веб-сторінка «Громадська приймальня»

На головній сторінці сайту Чернівецького обласного управління лісового та мисливського господарства одразу пропонується переглянути Положення про управління, що не надто цікаво для користувача і, тим більше, для потенційного клієнта. Варто було б все ж таки розмістити цікаві новини та пропозиції. Інтерфейс веб-сайту темний та непривітний.

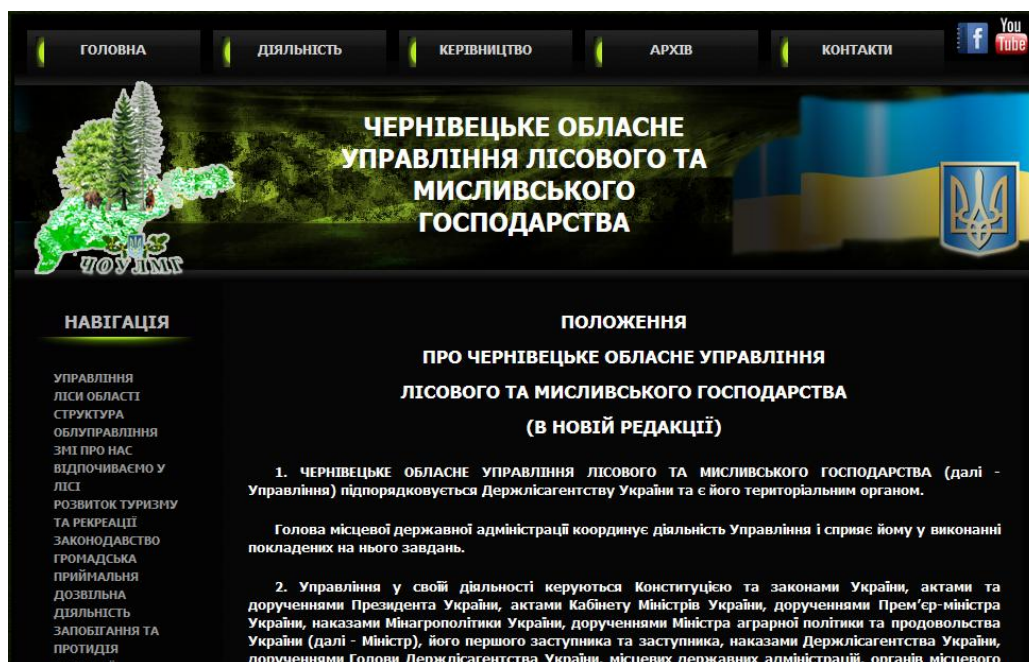


Рисунок 1.8. – Головна сторінка веб-сайту «Чернівецьке обласне управління лісового та мисливського господарства»

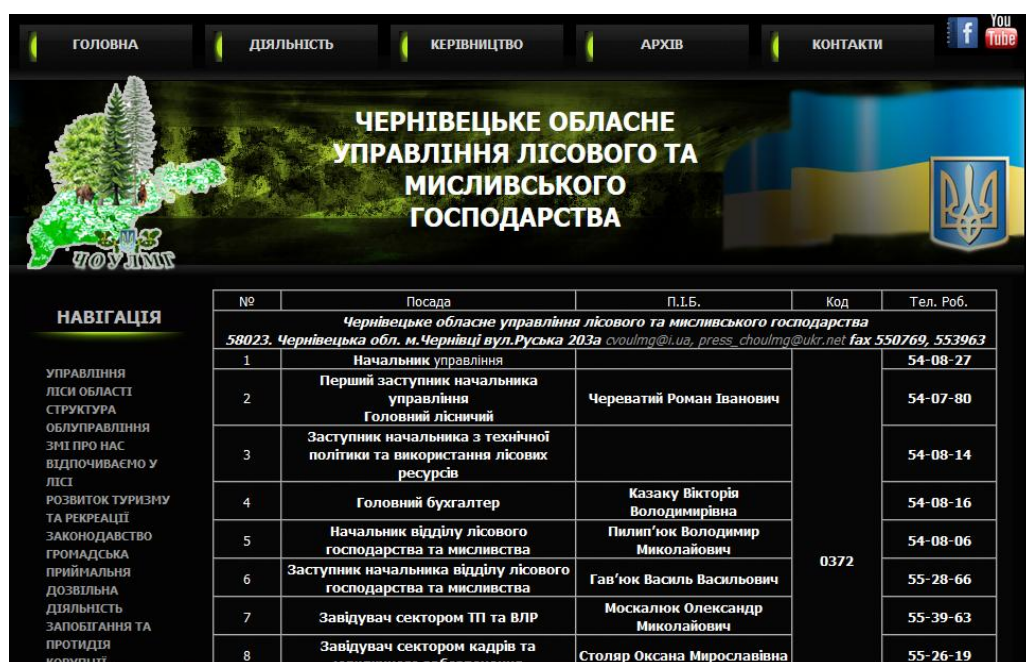


Рисунок 1.9. — Контакти працівників лісового господарства

На рисунку 1.9 представлена сторінка веб-сайту з контактами працівників управління. Контактна інформація подана у вигляді таблиці з зазначенням посади, П.І.П, кодом міста та робочим телефоном працівника. Інформацією поданою у такому вигляді дуже зручно користуватись і це є позитивним моментом сайту.

На рисунку 1.10 представлена веб-сторінка «Діяльність», на якій розміщена інформація про загальну характеристику лісового фонду Чернівецької області. Інформація надана по пунктах «Заповідний фонд», «Лісовідновлення», «Охорона і захист лісів» тощо. Інформації по кожному пункту наведено в достатній мірі, однак читати її важко через дрібний шрифт, темний фон і відсутність фото та відеоматеріалів.

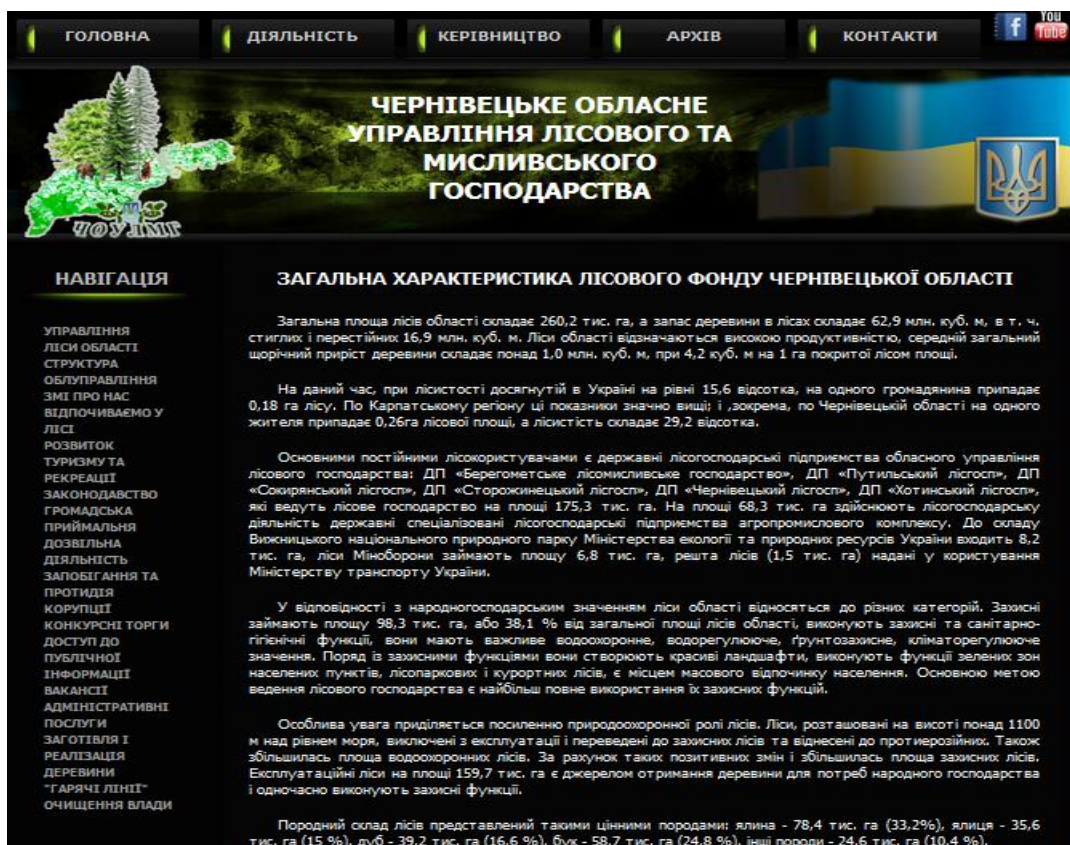


Рисунок 1.10. — Веб-сторінка «Діяльність»

На головній сторінці веб-сайту Івано-Франківського обласного управління лісового та мисливського господарства (див. рис. 1.11.) надається інформація про діяльність підприємства загалом, телефон гарячої лінії для запитань та є можливість замовлення посадкового матеріалу. Інтерфейс

користувача цікавий, адже спочатку інформація представляється на блакитному тлі та при наведені на неї – тло стає брудно зеленим. Такий задум не зручний при користуванні.

Івано-Франківське обласне управління лісового та мисливського господарства
| Держлісагентство | Керівництво | Організаційна структура | Контактні дані | Реформування галузі

ГОЛОВНА ▾ | НОВИНИ | ЗВІТНІСТЬ ▾ | ДОСТУП ДО ПУБЛІЧНОЇ ІНФОРМАЦІЇ | ОЧИЩЕННЯ ВЛАДИ

Івано-Франківське обласне управління лісового та мисливського господарства

СХЕМА територіального розміщення держлісгоспів Івано-Франківської області

Львівська область | Тернопільська область

Івано-Франківська область

Тлумачить: середня лісистість області становить 42%, тоді як по Україні цей показник становить 15,9%. Загальна площа земель лісового фонду постійного користування області складає 626,0 тис.га, що становить 6% від загальної площі лісів держави, ліси розташовані нерівномірно і в основному знаходяться в гірській частині, при цьому лісистість коливається від 5,5% (Снятинський район) до 66,2% (Долинський район).

Основними лісокористувачами області є:

- 17 підприємств Івано-Франківського обласного управління лісового та мисливського господарства, за ними закріплено 75% земель - 468,3 тис.га лісового фонду області.
- 25% (152,9 тис.га) лісового фонду надано в користування іншим користувачам, а саме:
 - спеціалізованим підприємствам ОП "Івано-Франківськблагроліс" – 74,0 тис.га
 - Карпатський природний національний парк - 38,6 тис.га

Рисунок 1.11. — Веб-сайт «Івано-Франківське обласне управління лісового та мисливського господарства»

На рисунку 1.12 представлена сторінка звітності підприємства, яка відсортована по роках і до кожного року організоване посилання на певну форму звітності, що є цікавим для користувача. Однак і тут виникає недолік, тобто звітності лише по балансах, а про реалізацію, вирубку, посадку лісових насаджень немає нічого.

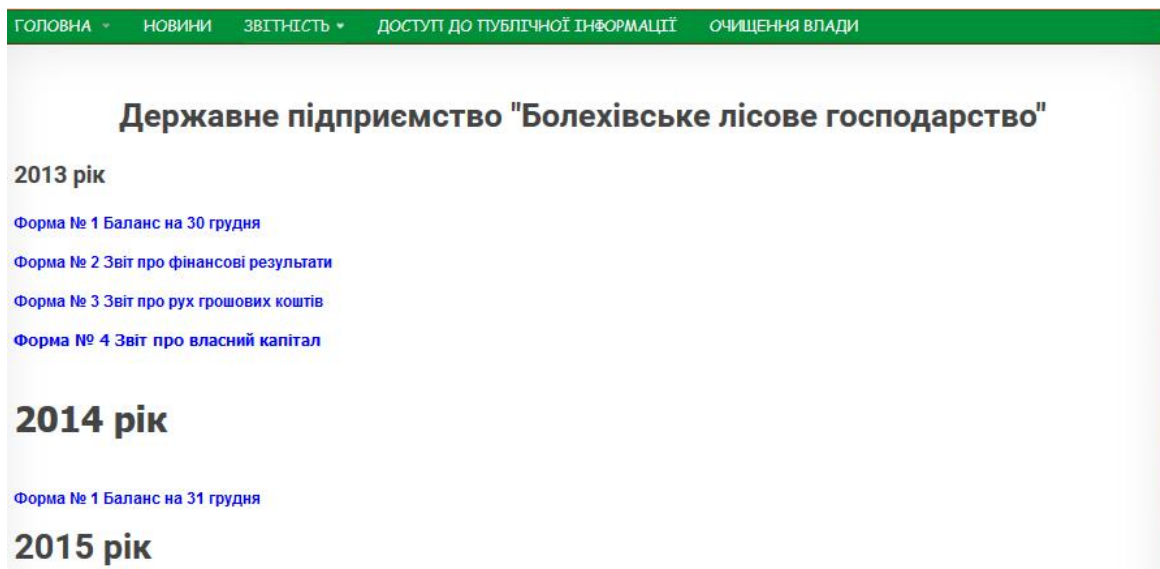




Рисунок 1.12. — Сторінка звітності підприємства

На рисунку 1.13 представлена сторінка з контактними даними працівників підприємства та відділень. Інформація надана в алфавітному порядку відділень, дрібним шрифтом та з великим відступом, що є незручно та нечитабельною.



Івано-Франківське обласне управління
лісового та мисливського господарства

| Держлісагентство | Керівництво | Організаційна структура | Контактні дані | Реформування галузі



ГОЛОВНА * НОВИНИ ЗВІТНІСТЬ * ДОСТУП ДО ПУБЛІЧНОЇ ІНФОРМАЦІЇ ОЧИЩЕННЯ ВЛАДИ

ДП «Болехівське лісове господарство»	Директор:
77200, м.Болехів, вул. Коновальця, 101	Яремко
Тел.: 3-43-73 (03437)	Михайло Михайлович
e-mail: broshniv@fforestry.gov.ua	
ДП «Брошнівське лісове господарство»	Директор:
77611, Рожнятівський р-н, смт. Брошнів, вул. Січових Стрільців, 54	Коляджин
Тел.: 4-69-11 (03474)	Ігор Федорович
e-mail: broshniv@fforestry.gov.ua	

Рисунок 1.13. — Сторінка контактних даних

Схожість та не оригінальність цих трьох веб-сайтів певний плюс, тому що користувачеві не потрібно до кожного сайту знаходити інший підхід роботи. З описаного вище можна зробити висновок, що за основу інтерфейсу можна взяти інтерфейс Хмельницького управління лісовим та мисливським господарством, а оформлення контактних даних зручно представлена на сайті Чернівецького управління.

В таблиці 1.2 зведені результати проведеного порівняльного аналізу.

Порівняльний аналіз розглянутих аналогів програмних продуктів

Фірма-розробник	Чернівецьке обласне управління лісового та мисливського господарства	ПрАТ <u>Мікросистема</u>	Івано-Франківське обласне управління лісового та мисливського господарства
Назва програмного продукту	Сайт «Чернівецьке обласне управління лісового та мисливського господарства»	Сайт «Ліси Хмельниччини»	Сайт «Івано-Франківське обласне управління лісового та мисливського господарства»
Функціональність	Достатня	Достатня	Низька
Інтерфейс користувача	Надто темний	Приємний, зручний	Простий
Допомога користувачу	Присутня	Присутня	Присутня

Розроблюючи програмний продукт та, проаналізувавши всі плюси і мінуси вище розглянутих веб-сайтів, були сформульовані основні вимоги до проєктованого веб-сайту:

- дружній користувацький інтерфейс, витриманий в однаковому стилі на всіх сторінках сайту;
- стисле представлення основної контактної інформації, виключення надлишкових нюансів;
- можливість зв'язку з керівництвом не лише по телефону, але й запису на прийом по конкретних питаннях;
- надання графічної та відеоінформації про планові заходи тощо.

1.4 Специфікація вимог до системи

Специфікація вимог для програмної системи - це повний опис поведінки системи що розробляється. Вона включає множину прецедентів які описують всі взаємодії, які користувачі мають з програмним забезпеченням [7]. Функціональні вимоги до системи також зручно описати діаграмою варіантів використання. Нефункціональні вимоги є вимогами, які накладають обмеження на проект чи реалізацію (такі як вимоги інженерії продуктивності, стандарти якості, чи обмеження проектування).

Таблицею 1.3 представлений глосарій розроблюваного програмного продукту

Таблиця 1.3

Глосарій основних термінів

Термін	Опис терміну
1. Основні поняття та категорії предметної області та проекту	
Веб – система	Реалізований продукт для підприємства та користувачів
Авторизація	Надання особі певних прав та визначення способу використання веб-системи
Замовлення	Процес замовлення товару та перевірки на складі
Продукція	Назва, опис та кількість продукції, що є на складі
Персонал	П.І.Б певних осіб, які працюють на підприємстві
Новини	Відображення головних подій на підприємстві, а також заходи, що були організовані підприємством
2. Користувачі системи	
Користувач	Людина, яка використовує веб-застосунок
Гість	Не авторизований користувач
Авторизований	Особа, яка пройшла авторизацію системи

Адміністратор	Користувач системи, котрий має особливі права доступу до системи та керує нею
3. Вхідні та вихідні документи	
Журнал	Документ у якому записується назва товару, кількість, прізвище отримувача, ціна.
Направлення	Формується документ у якому вказується в якій кількості і що буде направлено на склад підприємства
користувач	

Розкадровка варіантів використання

Щоб чітко визначити дії системи та роль користувачів у ній слід використати діаграму варіантів використання [8]. Діаграма варіантів використання проілюстрована на рисунку 1.14

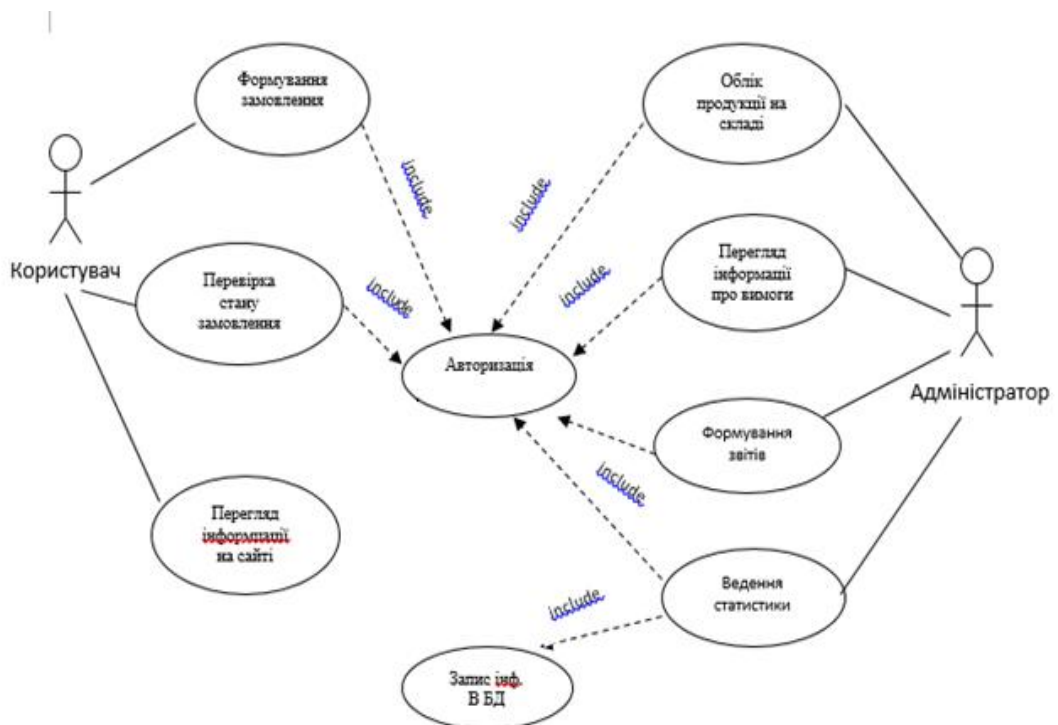


Рисунок 1.14. — Діаграма варіантів використання
Прототип функції для авторизованого користувача

Функція використовується тільки для авторизованого користувача, що дозволяє увійти в систему. Необхідно ввести логін користувача та пароль, що були вказані у реєстрації. Авторизація зображена на рисунку — 1.15

Рисунок1.15 — Авторизація користувача

Варіант використання «Авторизація» подано у таблиці 1.4.

Таблиця 1.4

Авторизація

Контекст використання	Авторизація
Дійові особи	Користувач, адміністратор.
Передумова	Потрібна зареєстрованість користувача в системі
Тригер	Кнопка «Увійти»
Сценарій	Заповнити відповідні дані та натиснути кнопку перевірки та входу
Пост-умова	Користувач авторизований

Прототип функції «Реєстрація» зображено на рисунку 1.16. Ця функція дозволяє новим користувачам зареєструватися в системі. Для цього необхідно ввести свій email , ім'я, прізвище, логін та пароль.

Рисунок 1.16 — Прототип функції «Реєстрація»

Варіант використання «Реєстрація» подано у таблиці 1.5

Реєстрація

Контекст використання	Реєстрація
Дійові особи	Не зареєстрований користувач
Передумова	Відкрита сторінка реєстрації
Тригер	Кнопка «Зареєструватись»
Сценарій	Заповнити поля коректними даними та натиснути кнопку «ОК»
Пост-умова	Користувач зареєстрований

Для виконання замовлення користувачу потрібно натиснути кнопку «Замовлення» та обрати певний товар, його кількість, вказати адресу доставки та номер телефону.

Рисунок 1.17 — Функція замовлення товару

Варіант використання «Замовлення товару» подано у таблиці 1.6.

Замовлення товару

Контекст	Формування замовлення
Дійові особи	Авторизований користувач
Передумова	Відкрита сторінка замовлення товару

Тригер	Кнопка «Замовлення»
Сценарій	Вибрати та ввести дані у всі поля та натиснути кнопку «ОК»
Пост-умова	Обробка замовлення

Користувач має можливість переглянути інформацію про певний товар: процес виробництва, персонал, кількість на складі і т. д..

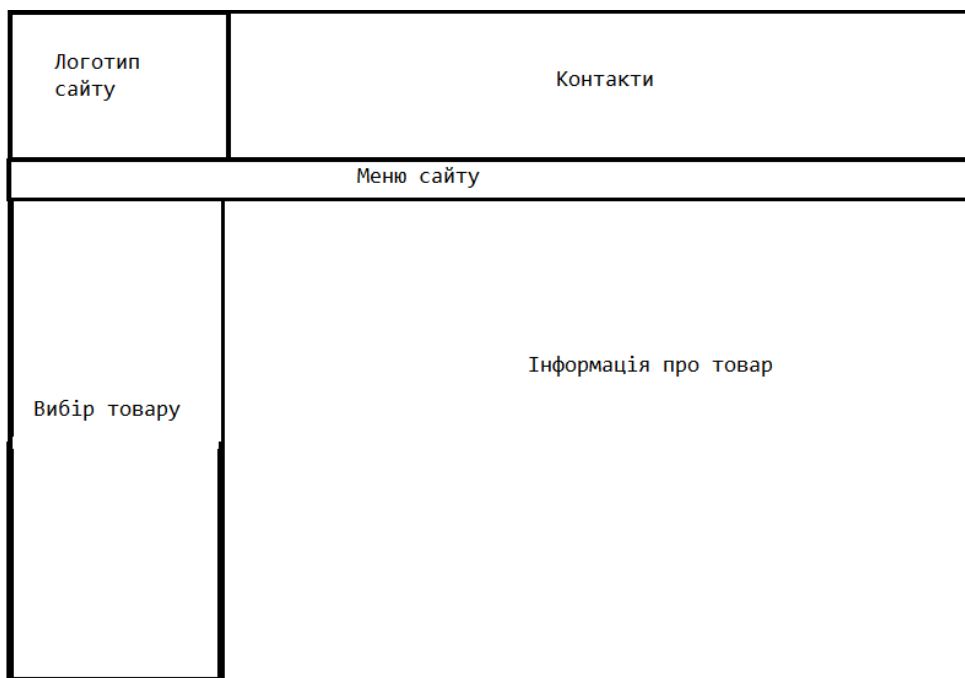


Рисунок 1.18 — Прототип функції «Інформація про товар»

Варіант використання «Інформація про товар» подано у таблиці 1.7. Де зображено функції, що дають змогу користувачу (обрати певний товар, вибрати та натиснути назву товару, переглянути інформацію)

Таблиця 1.7.

Інформація про товар

Контекст використання	Інформація про товар
Дійові особи	Користувач

Передумова	Обрати певний товар
Тригер	Вибрати та натиснути на назву товару
Сценарій	Прегляд інформації
Пост-умова	Отримано інформацію

Користувач системи може перейти на головну сторінку на якій зображено основну інформацію про підприємство. Прототип функції «Перегляд головної сторінки» зображено на рисунку 1.19

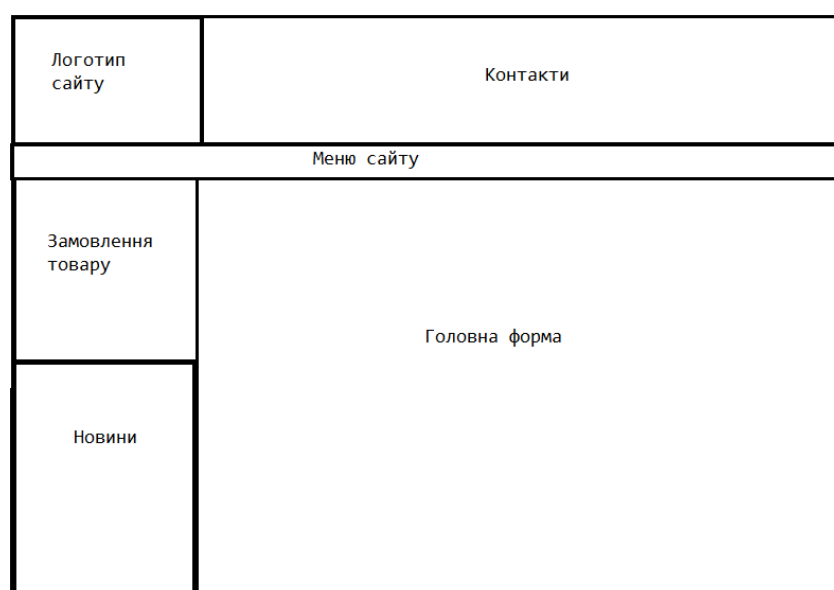


Рисунок 1.19 — Прототип функції «Перегляд головної сторінки»

Варіант використання «Перегляд головної сторінки» подано у таблиці 1.8.

Таблиця 1.8.

Перегляд головної сторінки

Контекст використання	Перегляд інформації на сайті
Дійові особи	Користувач
Передумова	Завантаження адреси сайту
Тригер	Перехід до головної форми
Сценарій	Відкрити головну форму
Пост-умова	Перегляд головної форми

Прототипи функцій та варіанти використання для адміністратора:

Адміністратор системи може додавати новий товар на замовлення та оновлювати дані про нього (кількість, ціна, можливість доставки).

Прототип функції «Додавання товару» складається з п'яти елементів, розташованих вертикально в межах одного контейнера:

- Назва товару
- Кількість
- Ціна товару
- Ціна доставки
- ДОДАТИ

Рисунок 1.20 — Прототип функції «Додавання товару»

Варіант використання «Додавання товару» подано у таблиці 1.9.

Таблиця 1.9.

Додавання товару

Контекст використання	Додавання товару
Дійові особи	Адміністратор
Передумова	Перехід на сторінку додавання товару
Тригер	Кнопка «Додавання товару»
Сценарій	Ввести назву товару, зазначити кількість, ціна за товар, ціна за доставку, кнопка «Додати»
Пост-умова	Товар доданий

Щоб здійснити оновлення товару необхідно обрати його з існуючого списку та внести нову інформацію про товар.

Прототип функції «Оновлення товару» складається з п'яти елементів, розташованих вертикально в межах одного контейнера:

- Назва товару ↓
- Кількість ↓
- Ціна товару ↓
- Ціна доставки ↓
- Оновити

Рисунок 1.21 — Прототип функції «Оновлення товару»

Варіант використання «Оновити товар» подано у таблиці 1.10.

Таблиця 1.10

Оновити товар

Контекст використання	Оновити товар
Дійові особи	Адміністратор
Передумова	Перехід на сторінку оновлення товару
Тригер	Кнопка «Оновити товар»
Сценарій	Вибрати товар зі списку, внести необхідні дані
Пост-умова	Товар оновлений

Для видалення товару необхідно обрати його зі списку. Видалення товару зображено на рисунку 1.22

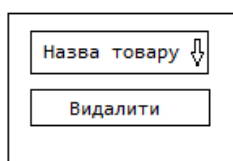


Рисунок 1.22 – Прототип функції «Видалення товару»

Варіант використання «Видалити товар» подано у таблиці 1.11.

Таблиця 1.11

Видалити товар

Контекст використання	Видалити товар
Дійові особи	Адміністратор
Передумова	Перехід на сторінку видалення товару
Тригер	Кнопка «Видалити».
Сценарій	1.Вибрати товар. 2.Підтвердити, натиснувши на кнопку «ОК».
Пост-умова	Товар видалений.

Специфікацію функціональних та не функціональних вимог наведено у таблицях 1.12 та 1.13.

Специфікація функціональних вимог

Ідентифікатор вимоги	Назва вимоги	Атрибути вимог		
		Пріоритет	Складність	Контакт
1	Перегляд головної сторінки	Обов'язкове	Низька	Користувач, Адміністратор
2	Реєстрація	Обов'язкове	Середня	Користувач, Адміністратор
3	Авторизація	Обов'язкове	Низька	Користувач, Адміністратор
4	Замовлення	Обов'язкове	Низька	Користувач, Адміністратор
5	Інформація про товар	Обов'язкове	Середня	Користувач, Адміністратор
6	Додати товар	Обов'язкове	Середня	Адміністратор
7	Оновити товар	Обов'язкове	Середня	Адміністратор
8	Видалити товар	Обов'язкове	Низька	Адміністратор

Таблиця 1.13

Специфікація не функціональних вимог

№	Назва вимоги	Характеристики
1	Застосовність	
1.1	Час навчання звичайних користувачів користувачів	15 хв.
1.2	Час навчання досвідчених користувачів	5 хв.
1.2	Час відгуку	Не більше 30 с.

Продовження таблиці 1.13.

2	Надійність	
2.1	Середній час безвідмовної роботи	24 дні.
	Максимальна норма помилок або дефектів	5
3	Робочі характеристики	
3.1	Швидкодія для транзакції(середнє значення)	2-3 с.
3.2	Місткість	800 користувачів
4	Проектні обмеження	
4.1	Мова програмування	PHP
5	Документація	Інструкція користувачу
6	Інтерфейс користувача	Веб-застосунок

Висновки до першого розділу

Виконавши перший розділ було проаналізовано предметну область та здійснено опис специфікації вимог. Також проведено огляд існуючих аналогів, де було визначено всі характеристики систем та на основі аналізу переваг і недоліків було спроектовано власну веб-систему. Наведено діаграми використання для акторів системи та здійснено їх розкадровку. Визначено всі функціональні та не функціональні вимоги до системи.

РОЗДІЛ 2

ПРОЕКТУВАННЯ

2.1 Розроблення архітектури програмної системи

Для розробки програмної системи було обрано трирівневу архітектуру.

У комп'ютерних технологіях трирівнева архітектура передбачає наявність таких компонентів програми: клієнтську програму, підключений до серверу додаток, який в свою чергу підключений до сервера бази даних [8].

Структурну схему програмної системи зображено на рисунку 2.1.

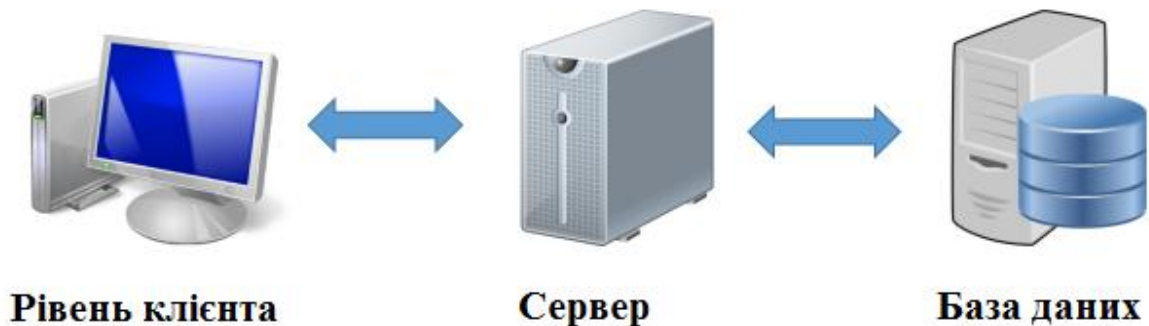


Рисунок 2.1 – Структурна схема програмної системи

Клієнт - це компонент графічного типу, на якому формується перший рівень, по суті, програмний продукт для користувача. На перший рівень може висувається і як правило виноситься: алгоритми шифрування, перевірка введених значень на відповідність і допустимість формату, примітивні операції (угруповання, сортування, підрахунок значень) з даними, інтерфейс користувача. На першому рівні не потрібно мати фактичних зв'язків з базою даних [9].

Сервер розташовується на другому рівні. На другому рівні зосереджена більша частина бізнес-логіки. Поза ним залишаються

фрагменти, що експортуються на термінали (див. вище), а також розміщені в третьому рівні збережені процедури і тригери [10].

Сервер бази даних забезпечує зберігання даних і виноситься на третій рівень. Зазвичай це стандартна реляційна або об'єктно-орієнтована СУБД. Якщо третій рівень являє собою базу даних разом з збереженими тригерами, процедурами і схемою, яка пояснює додаток в термінах реляційної моделі, то другий рівень будується як програмний інтерфейс, що зв'язує клієнтські складові з прикладною логікою бази даних [11].

Функціональну структуру системи можна розділити на такі модулі:

- Модуль обробки інформації адміністратора:

1. Додавання товару;
2. Оновлення товару;
3. Видалення товару;
4. Перевірка коректності замовлень.

- Модуль обробки інформації клієнта:

1. Авторизація;
2. Реєстрація;
3. Замовлення.

Процес функціонування системи можна відобразити побудованими діаграмами станів. Діаграма станів авторизації користувача зображено на рисунку 2.2.



Рисунок 2.2 – Діаграма станів (авторизація)

На рисунку 2.3 зображено діаграму станів стан замовлення, що показує як саме проводиться замовлення товару клієнтом на підприємстві. Клієнт обирає замовлення після цього проводиться виведення інформації про замовлення та виводиться інформація про стан продукту.



Рисунок 2.3 — Діаграма станів (стан замовлення)

На рисунку 2.4 зображено діаграму станів на якій зображено процес формування звіту. Для цього вибирається замовлення, переглядається інформація, оброблюються дані та формується звіт.



Рисунок 2.4 — Діаграма станів (формування звіту)

2.2 Проектування структури бази даних

Щоб спроектувати структуру бази даних необхідно створити DFD-моделі предметної області. Діаграма потоків даних верхнього рівня (див. рисунок 2.5).

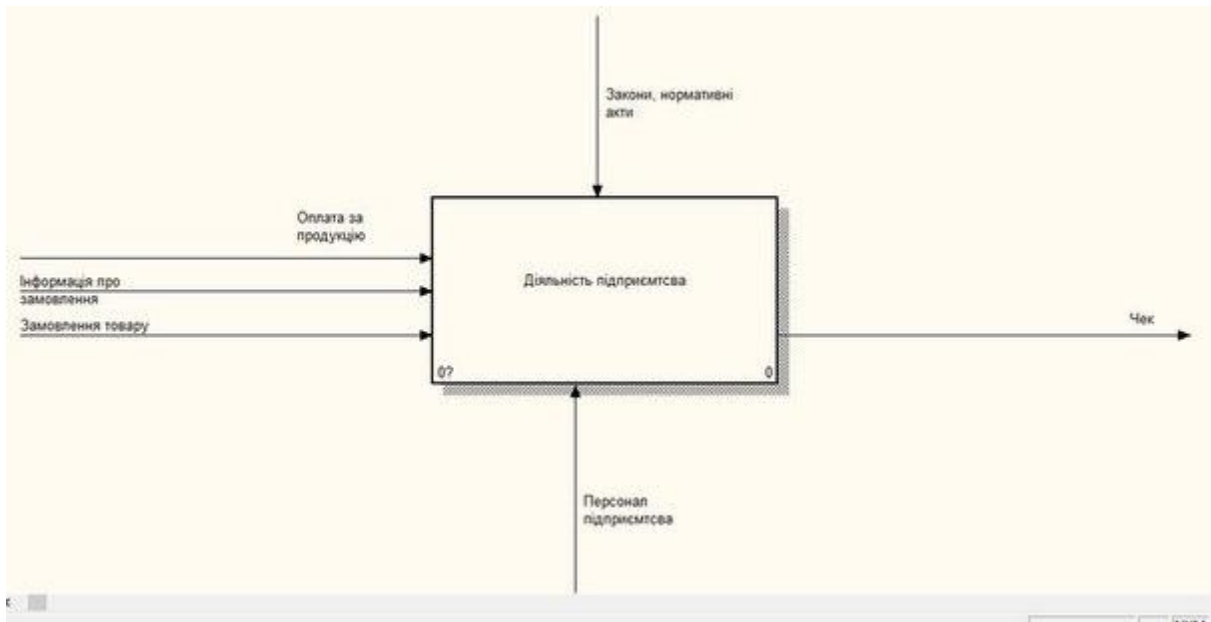


Рисунок 2.5 — IDEFO діаграма верхнього рівня

Діаграма потоків даних верхнього рівня (див. рис. 2.6)

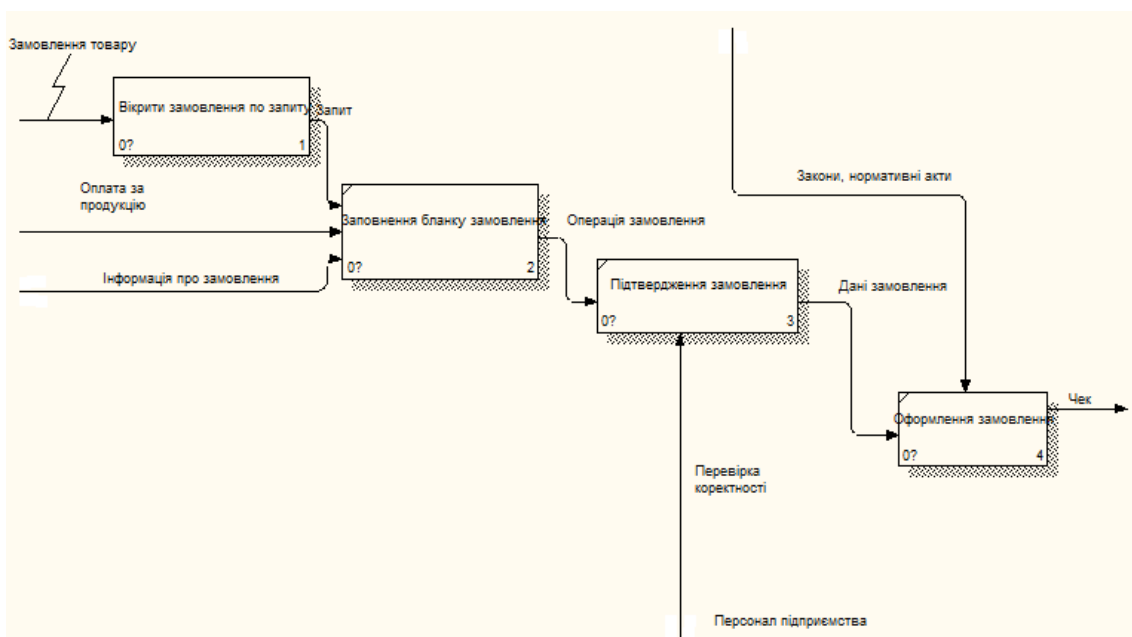


Рисунок 2.6 — Деталізація IDEFO діаграми

На рисунку 2.5 зображено діяльність підприємства, а саме структуру замовлення товару та процес його здійснення. Процес замовлення здійснюється в декілька етапів:

- відкриття замовлення по запиту;
- заповнення бланку замовлення;
- підтвердження замовлення;
- оформлення замовлення.

Наступним кроком є створення діаграми класів UML. Рисунок 2.6

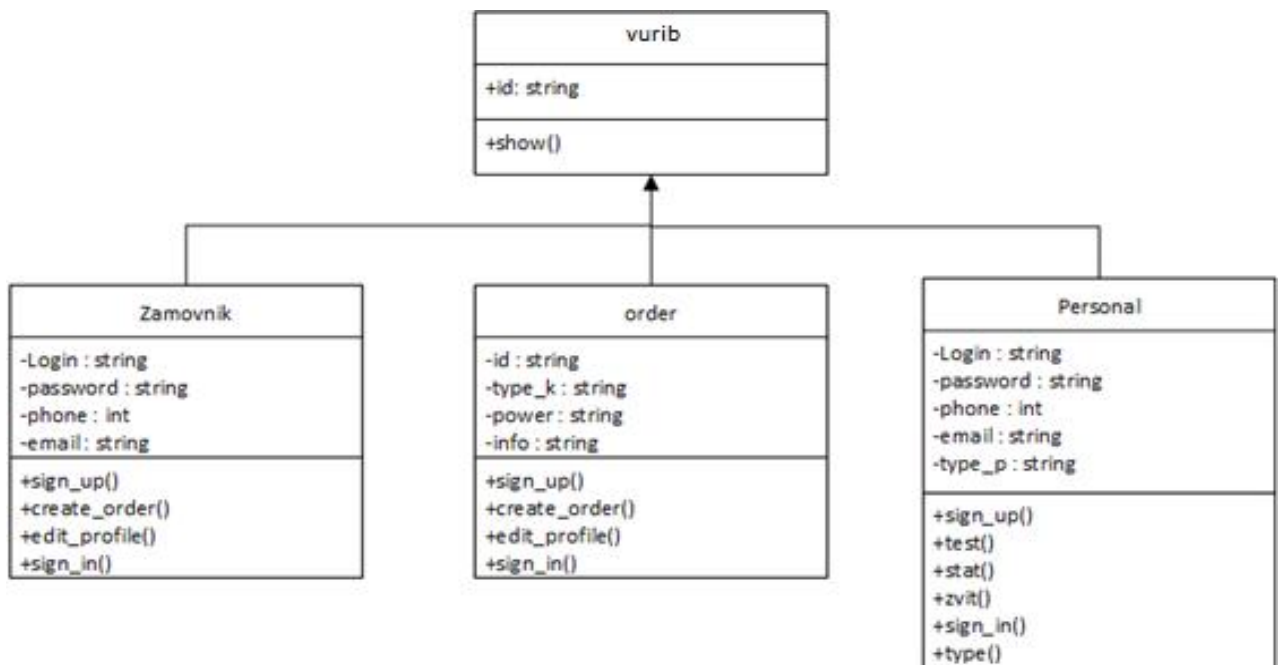


Рисунок 2.6 – Діаграма класів UML

На діаграмах класів показано різноманітні класи, які утворюють систему і їх взаємозв'язки. На цій діаграмі показують класи, інтерфейси, об'єкти й кооперації, а також їхні відносини.

Діаграма кооперативної моделі даних зображує сутності та атрибути певних категорій [12]. На ній чітко визначено основні об'єкти, що будуть представлені у вигляді таблиць у подальшій розробці бази даних. Діаграма кооперативної моделі даних зображена на рисунку 2.7

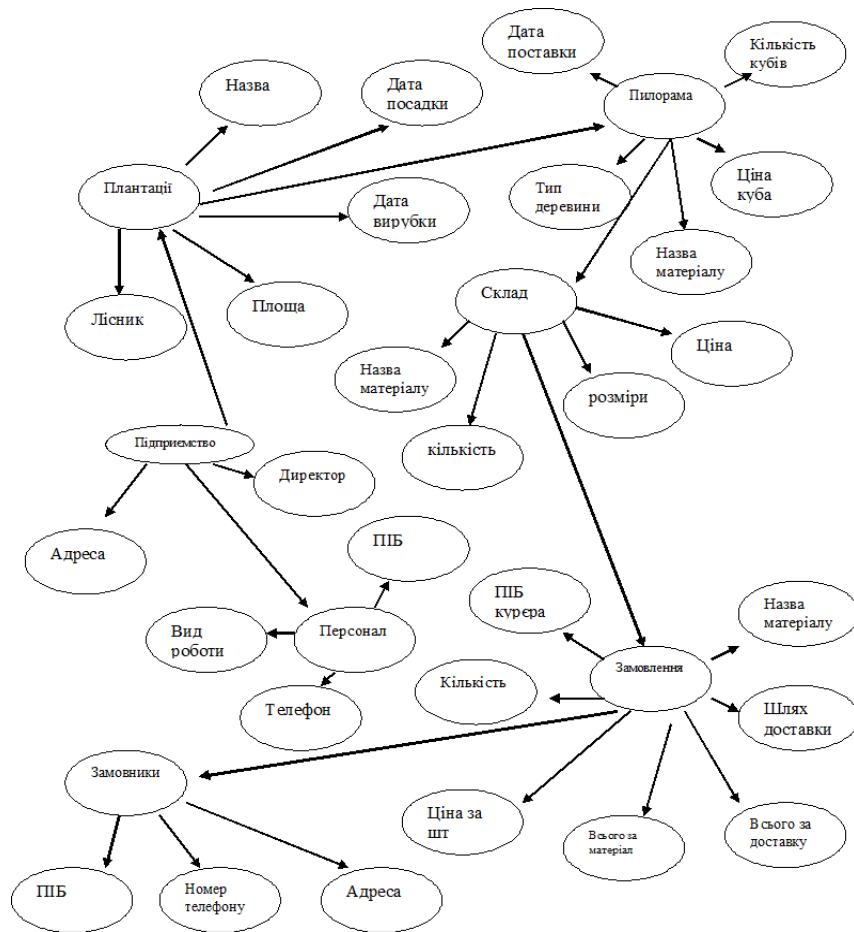


Рисунок 2.7 – Діаграма корпоративної моделі даних.

Було створено таблицю ідентифікаторів, яку подано у таблиці 2.1.

Таблиця 2.1

Таблиця ідентифікаторів

Об'єкт	Властивість	Тип	Розмірність	Ідентифікатор
Підприємство	Директор	Текст	15	pidpriemstvo
	Адреса	Текст	25	Director Adresa
Плантації	Назва	Текст	15	plantatsii
	Дата посадки	Число	10	Nazva_plantasii
	Дата вирубки	Число	10	data_posadku
	Площа плантації	Число	10	data_vurybku
	Лісник	Текст	15	ploshcha_plantasii
	Номер лісника	Число	15	lisnuk_PIB
				nom_lisnuka

Продовження таблиці 2.1

Пилорама	Назва матеріалу Тип деревини Ціна куба Кількість кубів Дата поставки	Текст Текст Число Число Число	20 15 10 10 10	pilorama Nazva_materialu Tip_derevunu sina_kub kilkist_kubiv data_postavku
Склад	Назва матеріалу Кількість Розміри Ціна шт	Текст Число Число Число	20 10 10 10	sklad Nazva_materialu Kilkist Rozmiru sina_sht
Персонал	ПІБ Вид роботи Телефон	Текст Текст Число	30 20 20	personal PIB Vid_roboty Telefon
Замовники	ПІБ Телефон	Текст Число	20 15	Zamovniki PIB Telefon
Кур'єри	ПІБ Телефон Назва авто Номер авто Ціна доставки	Текст Число Текст Текст Число	20 15 20 20 10	kureri PIB Telefon Nazva_avto nomer_avto sina_dost_za_km
Замовлення		Текст Текст Текст Число Число Число Число Число Число	15 15 20 10 10 10 10 10 10	Zamovlennia PIB_zamovnuka PIB_kur Nazva_mateialu Kilkist sina_sht shliah_dostavku_km vsiogo_za_material vsiogo_za_dorogu Vsiogo_do_splatu

Для уникнення непотрібних деталей слід використати концептуальну модель даних, тому що на ній представлені основні атрибути предметної області Проект, який знаходиться в стадії планування не потребує значної деталізації [13]. Потрібно лише знати загальні дані, а іноді тільки типи даних. Маса деталей не є необхідним для цього. Концептуальна модель даних потрібна аби реалізувати цю мету.

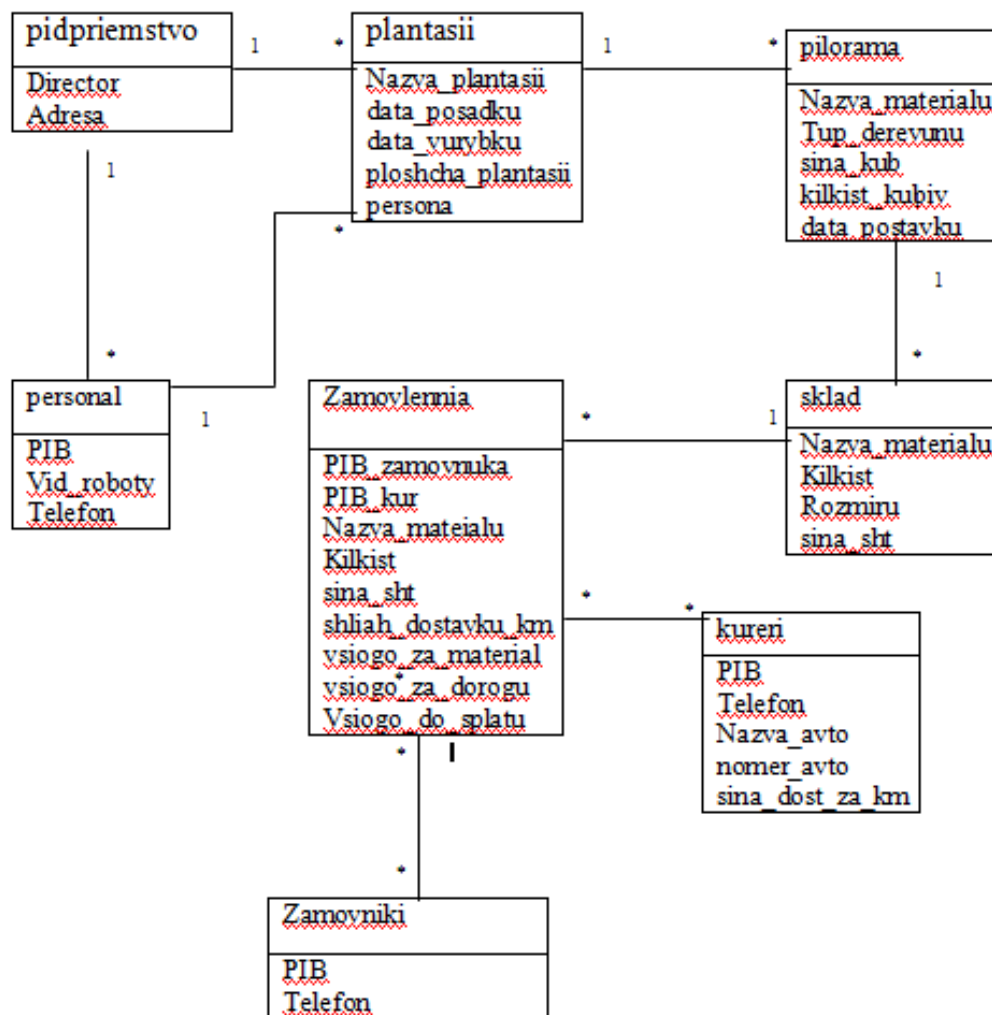


Рисунок 1.8 Концептуальна модель бази даних

Було створено ER-діаграму до бази даних. ER - модель використовується при високорівневої (концептуальному) проектуванні баз даних. З її допомогою можна виділити ключові сутності і позначити зв'язки, які можуть встановлюватися між цими сутностями [13].



Рисунок 2.9 – ER-діаграма

Здійснено планування індексів на якому визначається звернення до записів таблиці під певним набором символів [14]. Це дозволяє прискорити операції пошуку в таблицях. При індексуванні задається: назва відношення, назва індексу та назва атрибуту.

Назва відношення	Назва індекса	Назва атрибта
plantasii	Iplantasii_ Nazva	Nazva
plantasii	Iplantasii_ Nazva_ ploscha	Ploscha
pilorama	Ipilorama_ Nazva_ materialu	Nazva_ materialu
Zamovlennia *	iZamovlennia_ Kilkist	Kilkist
sklad *	Isklad_ Nazva	Nazva
sklad	Isklad_ Nazva_ Kilkist	Nazva, Kilkist
kureri	Ikureri_ PIB	PIB
kureri	Ikureri_ PIB_ Telefon	PIB, Telefon
Zamovniki	iZamovniki_ PIB	PIB
Zamovniki	iZamovniki_ PIB_ Telefon	PIB, Telefon

Таблиця 2.2

Планування індексів

Висновок до другого розділу

Визначено архітектуру системи, спроектовано базу даних, інтерфейс користувача та об'єктну модель програмного засобу. Спроектовано структурну схему програмної системи та здійснений її опис. Також проведено збір та аналіз вхідних даних, побудовано UML-діаграми станів до кожного модуля програмної системи. Визначено основні елементи бази даних та встановлено зв'язки між ними та подано інформаційну модель бази даних.

РОЗДІЛ 3

ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1. Програмна реалізація проекту

Для розробки веб-орієнтованої програмної системи було обрано мову програмування PHP.

PHP – це скриптова мова програмування. PHP може використовуватися на всіх основних операційних системах, включаючи Linux, багато варіантів Unix (в тому числі HP-UX, Solaris і OpenBSD), Microsoft Windows, Mac OS X, RISC OS, і, ймовірно, інші. PHP також сьогодні підтримується для більшості веб-серверів. За допомогою PHP можливо не тільки видавати HTML. Можливості PHP включають в себе формування зображень, PDF-файли і навіть флеш-роликів, які генеруються на льоту. Ви можете також виводити будь-які текстові дані, такі як XHTML і будь-який інший XML-файл. PHP може здійснювати автоматичну генерацію таких файлів і зберігати їх в файлової системі, замість того, щоб друкувати його, формуючи кеш на стороні сервера для динамічного контенту [14].

Основні переваги програмування PHP включають в себе наступне:

Відкрите програмне забезпечення - знаходиться у вільному доступі для використання. Спільнота розробників PHP з відкритим вихідним кодом забезпечує технічну підтримку і постійно вдосконалює оновлення основних PHP функціональних можливостей.

Багатоплатформість - PHP забезпечує високу сумісність з провідними операційними системами і веб-серверів, що дозволяє бути легко використовувати на різних платформах.

Дружній до користувача - розроблений в зручній для користувача формі.

Швидкість - PHP розроблений, щоб добре працювати з мережею, і тому такі речі, як отримати доступ до GET і POST, і працювати з HTML і URL,

вбудовані модулі на мові PHP. Це робить його дуже простим і зрозумілим, щоб зробити веб-сайт.

Простота розробки програмного забезпечення. Існує багато хостинг-компаній, які дають змогу надійно користуватися ресурсами свого серверу, для створення веб-сайту, що розроблений на PHP, так що ви можете зробити сайт дійсно легко.

Захист - PHP пропонує безпеку, а також, що допомагає запобігти зловмисні атаки. Ці рівні безпеки можна налаштувати в певному файлі.

Для швидкого розроблення програмного продукту його було розбито на певні модулі. Модуль є частиною програми, програми складаються з однієї або більше незалежно розроблених модулів, які не поєднуються, поки програма не пов'язана. Один модуль може містити один або кілька підпрограм.

Програмна система складається із восьми модулів, які мають між собою спільні дані та поєднуються зв'язками, що в свою чергу призводить до ефективності виконання операцій.

Перший модуль – «Реєстрація та авторизація користувача» призначений для занесення в базу даних дані про нового користувача та перевірку введених даних на коректність при вході на сайт.

Другий модуль – «Плантації – таблиця» призначений для перегляду, додавання, видалення, редагування даних в таблицю та організований пошук по ній.

Третій модуль – «Пилорама – таблиця» призначений для перегляду, додавання, видалення, редагування даних в таблицю та організований пошук по ній.

Четверний модуль – «Склад – таблиця» призначений для обліку продукції лісового господарства на складі.

П'ятий модуль – «Кур'єри – таблиця» призначений для перегляду інформації про кур'єрів.

Шостий модуль – «Замовники – таблиця» призначений для перегляду інформації про замовників.

Сьомий модуль – «Замовлення – таблиця» призначений для організації здійснення замовлення користувачем та перегляду адміністратором.

Восьмий модуль – «Контакти» - призначений для зворотного зв'язку між користувачем та адміністратором сайту.

У структурному програмуванні застосовується ієрархічний підхід, в якому окремі структурні одиниці перетворюються на функції, що можуть приймати параметри та повертати результат.

Нижче наведено програмний код функцій та описано логіку його розробки.

Підключення до бази даних:

```
<?php
$db = mysql_connect('localhost', 'hholik', '888000');
mysql_select_db('misha', $db);
mysql_query("set_client = 'utf8'");
mysql_query("set character_set_results = 'utf8'");
mysql_query("set collation_connection = 'utf8_general_ci'");
mysql_query("SET NAMES utf8");
?>
```

Формування таблиці:

```

        <table class = "table_plant">
        <tr>
        <td class = "table_plant_head1">№</td>
        <td class = "table_plant_head2">Назва плантації</td>
        <td class = "table_plant_head3">Дата посадки</td>
        <td class = "table_plant_head4">Дата вирубки</td>
        <td class = "table_plant_head5">Площа</td>
        <td class = "table_plant_head6">ПІБ лісника</td>
        <td class = "table_plant_head7">Телефон</td>
        </tr>
    <?php
        include "db.php";
        $result = mysql_query("SELECT * FROM plantasii");
    if (mysql_num_rows($result) > 0)
    {$row = mysql_fetch_array($result);
        do
        {echo '
            <tr class = "table_plant_head11">
            <td class = "table_plant_head1"><a><p>'. $row[id]. '</p></a></td>
            <td class = "table_plant_head2"><a><p>'. $row[Nazva_plantasii]. '</p></a></td>
            <td class = "table_plant_head3"><a><p>'. $row[data_posadku]. '</p></a></td>
            <td class = "table_plant_head4"><a><p>'. $row[data_vurybku]. '</p></a></td>
            <td class = "table_plant_head5"><a><p>'. $row[ploshcha_plantasii]. '</p></a></td>
            <td class = "table_plant_head6"><a><p>'. $row[lisnuk_PIB]. '</p></a></td>
            <td class = "table_plant_head7"><a><p>'. $row[mob_nom_lisnuka]. '</p></a></td>
            </tr>';}
        while ($row = mysql_fetch_array($result))}
    ?>
</table>

```

Створення форми для додавання інформації:

```

<form action="add_res.php" method="post">
    Назва:</br>
    <input type="text" name="Nazva_plantasii"/></br></br>
    Дата посадки:</br>
    <input type="text" name="data_posadku"/></br></br>
    Дата вирубки:</br>
    <input type="text" name="data_vurybku"/></br></br>
    Площа:</br>
    <input type="text" name="ploshcha_plantasii"/></br></br>
    ПІБ лісника:</br>
    <input type="text" name="lisnuk_PIB"/></br></br>
    Телефон:</br>
    <input type="text" name="mob_nom_lisnuka"/></br></br>
    <input type="submit" name="submit" value="Додати"/>
</form>
<a class="close" title="Закрити" href="#close"></a>
</div>

```

Додавання інформації:

```

<?php
    include "db.php";
    $result = mysql_query("INSERT INTO pulorama (data_postavku,Nazva_plantasii,
    Tup_derevunu,kilkist_kubiv,sina_kub,Nazva_materialu) VALUES ('$_POST[data_postavku]
    ', '$_POST[Nazva_plantasii]', '$_POST[Tup_derevunu]', '$_POST[kilkist_kubiv]',
    '$_POST[sina_kub]', '$_POST[Nazva_materialu]')");
    if($result == true)
    {
        echo "Дані успішно додано.";
    }
    else
    {
        echo mysql_error();
    }
?>
<a href="index.php"> Назад </a>

```

Редагування інформації:

```

<?php
    include "db.php";
    $result = mysql_query("UPDATE INTO pulorama (data_postavku,Nazva_plantasii,
    Tup_derevunu,kilkist_kubiv,sina_kub,Nazva_materialu) VALUES ('$_POST[data_postavku]',
    '$_POST[Nazva_plantasii]', '$_POST[Tup_derevunu]', '$_POST[kilkist_kubiv]', '$_POST[sina_kub]',
    '$_POST[Nazva_materialu]')");
    if($result == true)
    {
        echo "Дані успішно додано.";
    }
    else
    {
        echo mysql_error();
    }
?>
<a href="index.php"> Назад </a>

```

Видалення інформації:

```

<?php
include "db.php";
$result= mysql_query("DELETE FROM plantasii $_GET[type]\
WHERE id= '$_POST[id]'");
    if($result == true)
    {
        echo "Дані успішно видалено.";
    }
    else
    {
        echo mysql_error();
    }
?>
<a href="index.php"> Назад </a>

```

Пошук:


```

<form action="index.php" method="POST">
<p>
Введіть ПІБ клієнта: <input name="surname" type="text">
<input name="send" type="submit" value="Пошук" />
</p>
</form>
<?php
include ("db.php");
if (isset($_POST['send'])){ $send = $_POST['send'];}
if (isset($_POST['surname'])){ $surname = $_POST['surname'];}
if ($send)
$result = mysql_query ("SELECT * FROM klientu WHERE surname = '.$surname.'", $bd);
while ($myrow = mysql_fetch_array ($result))
{
?>
<?php echo $surname; ?>
<?php echo $name; ?>
<?php echo $patronymic; ?>
<?php echo $post; ?>
<? } ?>

```

Вимоги до апаратного забезпечення.

Програма реалізації продукції ДП Клеваньське лісове господарство призначена для експлуатації на звичайних персональних комп'ютерах. Оперативна пам'ять — не менше 1 Гб та простір на жорсткому диску — не менше 1,5 Гб. Розроблений програмний продукт функціонує на комп'ютерах з установленими операційними системами:

- Windows XP;
- Windows 7;
- Windows 8;
- Windows 10.

Встановлений програмний засіб перегляду веб-сторінок, тобто, браузер Opera версії 10 та вище, Mozilla Firefox версії 6.0 та вище, Google Chrome 14.0 та вище.

Сумісність програми має забезпечити легке використання з іншим програмним забезпеченням. Також були використані всі правила та норми при розробці програмного продукту, а саме: природність інтерфейсу, узгодженість інтерфейсу, люб'язність інтерфейсу, принцип здорового

зв'язку, простота інтерфейсу, гнучкість інтерфейсу та естетична привабливість [15].

3.2. Програмна реалізація бази даних

Для легкого і простого маніпулювання величезним обсягом даних програмісти створили так звані системи керування базами даних або скорочено СУБД. СУБД дасть користувачу можливість швидкого пошуку потрібної інформації; взаємозв'язок усіх введених даних між собою; використання даних різним програмним забезпеченням (наприклад, прикладними чи системними) [16]. База даних повинна полегшити роботу усіх працівників, що задіяні в базі даних, зменшити об'єм роботи та час на пошук потрібної інформації про продукцію. Для створення бази даних було обрано систему керування базами даних MySQL, що є реляційною системою управління базами даних, яка є відкритим середовищем для створення бази даних.

Особливості MySQL:

1. Через свою унікальну архітектуру зберігання продуктивність MySQL дуже висока.
2. Підтримує велику кількість вбудованих додатків, що робить MySQL дуже гнучким.
3. Використання тригерів, збережених процедур і уявлень, що дозволяє розробнику, дати більш високу продуктивність.
4. Дозволяє повертати транзакції, здійснювати фіксацію і відновлення після збою.

Додавання нового запису передбачає створення у іншому відношенні його значення, якщо, відповідно, зазначено зв'язок до попереднього відношення [17]. Що в свою чергу приведе до збереження цілісності бази даних та уникнення дублювання даних. Додавання нових плантацій, пилорам та персоналу передбачає додавання нових записів у відповідні відношення. Додавання підприємств не передбачене, а при реєстрації нового опису

замовлення будуть використані вже існуючі коди відповідного замовника та доставки. Пошук даних таблиці буде здійснюватися по відповідному атрибуту, в результаті буде отримано дані по критеріям пошуку. Показати список всіх таблиць в базі даних.

База даних також міститиме інформацію про замовників (код замовлення, ПІБ замовника, номер телефону, адреса). База даних міститиме таблицю склад, що матиме атрибути: назва матеріалу, кількість, розміри, ціна. Таблиця «Кур'єри» з атрибутами: код кур'єра, ПІБ, телефон, автомобіль, номер авто, ціна доставки) Завдання бази даних надати зручний доступ до потрібної інформації користувачу, виконувати друк інформації, виведення її на екран та оформлення звітів по певних таблицях. Створення запитів та форм для ефективного використання бази даних.

Користувач повинен мати можливість знайти потрібну інформацію про процес реалізації продукції на лісовому господарстві, дані про замовлення певного покупця. База даних повинна полегшити роботу усіх працівників, що задіяні в базі даних, зменшить об'єм роботи та час на пошук потрібної інформації про продукцію.

Створено ряд таблиць, які поєднані між собою зв'язками та залежать один від одного:

- Таблиця «Склад»;
- таблиця «Плантації»;
- таблиця «Пилорама»;
- таблиця «Замовники»;
- таблиця «Замовлення»;
- таблиця «Кур'єри»;

Створюючи базу даних головною необхідністю було уникнути надлишковості даних, здійснити нормалізацію БД. Ці дії допоможуть уникнути формування помилок [18]. Також потрібно здійснити

- Правильність введених аби в подальшому реалізувати коректну звітність системи;

- створити зв'язки між таблицями;
- надати певний тип даних кожному запису.

Основним завданням веб-орієнтованої програмної системи «Клеванське лісове господарство» полягає в замовленні певної продукції та виведення звітності. Тому головну увагу при розробці необхідно приділити даним функціям. Нижче наведено головні завдання системи у мові структурованих запитів SQL.

Таблиця «Замовлення»:

```
CREATE TABLE `zamovlennia` (
  `Kod_zamovlennia` INT(11) UNSIGNED NOT NULL AUTO_INCREMENT,
  `PIB_zamovnuka` VARCHAR(255) NULL DEFAULT NULL,
  `PIB_kur` VARCHAR(255) NULL DEFAULT NULL,
  `Nazva_mateialu` VARCHAR(255) NULL DEFAULT NULL,
  `kilkist` VARCHAR(255) NULL DEFAULT NULL,
  `sina_sht` INT(11) NULL DEFAULT NULL,
  `shliah_dostavku_km` INT(11) NULL DEFAULT NULL,
  `vsiogo_za_material` INT(11) NULL DEFAULT NULL,
  `vsiogo_za_dorogu` INT(11) NULL DEFAULT NULL,
  `Vsiogo_do_splatu` INT(11) NULL DEFAULT NULL,
  PRIMARY KEY (`Kod_zamovlennia`)
)
COLLATE='utf8_general_ci'
ENGINE=InnoDB
AUTO_INCREMENT=26;
```

Таблиця «Замовники»:

```
CREATE TABLE `zamovniku` (
  `Kod_zamovnuka` INT(11) UNSIGNED NOT NULL AUTO_INCREMENT,
  `PIB_zamovnuka` TEXT NULL,
  `Nom_telefonu` INT(255) NULL DEFAULT NULL,
  `Adresa` VARCHAR(255) NULL DEFAULT NULL,
  PRIMARY KEY (`Kod_zamovnuka`)
)
COLLATE='utf8_general_ci'
ENGINE=InnoDB
AUTO_INCREMENT=20;
```


Висновки до третього розділу

Розробку веб-орієнтованої програмної системи «Клеванське лісове господарство» було поділено на два етапи:

1. Розробити веб-сайт.
2. Реалізувати базу даних.

При розробці веб-орієнтованої програмної системи було обрано мову PHP, та розділено програмний продукт на вісім модулів. Реалізуючи базу даних було обрано систему управління базами даних MySQL.

РОЗДІЛ 4

ТЕСТУВАННЯ ТА ДОСЛІДНА ЕКСПЛУАТАЦІЯ

4.1 Тестування

Процес виконання програми з метою виявлення помилок в програмному забезпеченні називається тестуванням. Крім цього можна сказати, що програмний продукт відповідає технічних вимогам, працює як і очікувалося [19]. Фактично тестування програмного продукту є методом оцінки функціональних можливостей програмного забезпечення. Існує багато різних типів тестування.

Для розроблювального програмного продукту, тобто, веб-орієнтованої програмної системи «Клеванське лісове господарство» було обрано нижче наведені типи тестування:

1. Функціональне тестування;
2. GUI тестування;
3. Тестування безпеки.

Функціональне тестування

Функціональне тестування виконується для перевірки того, що програмний додаток виконує всі функції належним чином відповідно до проектних специфікацій. Під час тестування функціональності ми перевіряємо функції основного додатку, введення тексту, функції меню, установку і налаштування в мережевих системах [20].

Нижче наведено дії для виконання функціонального тестування:

1. Встановлення та налаштування на персональних комп'ютерах під управлінням операційної системи;
2. Перевірка основних функцій програми (обробка рядків, тексту, даних);
3. Клавіші без будь-якого недоцільного дублювання;
4. Обробка тексту;
5. Перевірка на правильну функціональність після усунення недоліків системи.

Беручи за основне перелік наведених дій при функціональному тестування було розроблено десять тестів. Після чого здійснилася перевірка веб-орієнтованої програмної системи «Клеванське лісове господарство». Отримані результати були позитивними. Програмна системи відповідає усім вимогам що були поставлені при розробці та проектуванні веб-сайту. Текст функціонально тестування наведено в додатку А.

Тестування безпеки

Тестування безпеки це тип тестування програмного забезпечення який призначений для перевірки чи програмний продукт вразливий до атак, визначення того чи програмна система забезпечує захист даних і підтримує функціонульні характеристики, які передбачалися.

Для забезпечення безпеки системи необхідно дотриматися певних заходів:

1. Конфіденційність;
2. Цілісність;
3. Аутентифікація;
4. Доступність;
5. Авторизація;
6. Безвідмовність.

4.2 Розгортання програмного продукту

Для користування веб-орієнтованою програмною системою «Клеванське лісове господарство» необхідно мати програмний засіб для перегляду веб-сайтів, тобто, браузер та підключення до мережі Інтернет. Необхідно ввести в посилання веб-сайту в адресну строку браузера та натиснути клавішу «Enter».

Головна сторінка веб-сайту «Клеванське лісове господарство»

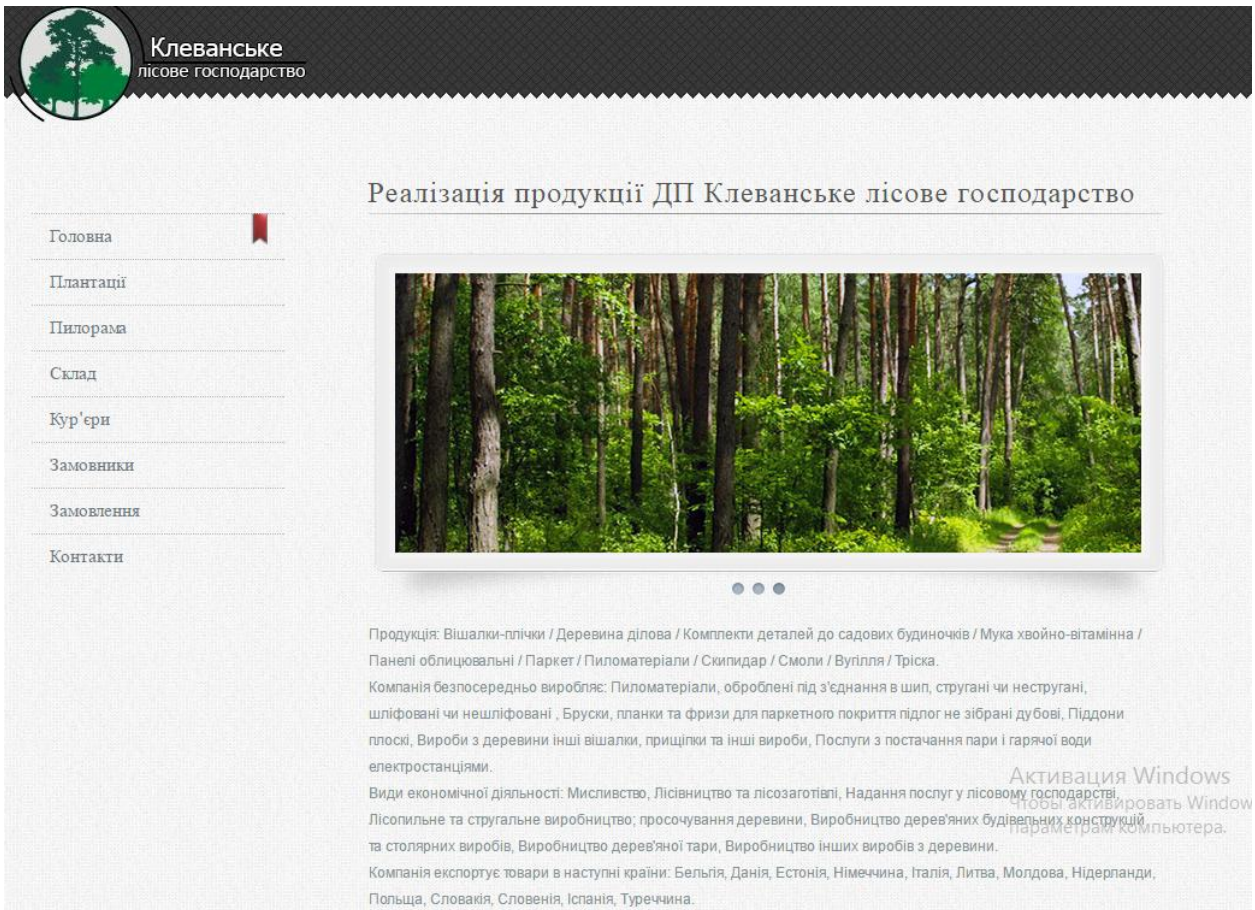


Рисунок 4.1 – Головна сторінка веб- сайту

Організація замовлення здійснюється у вигляді таблиці. Для цього необхідно натиснути вниз таблиці клавішу «Додати» та ввести коректні дані в поля. Процес замовлення представлений на рис. 4.2.

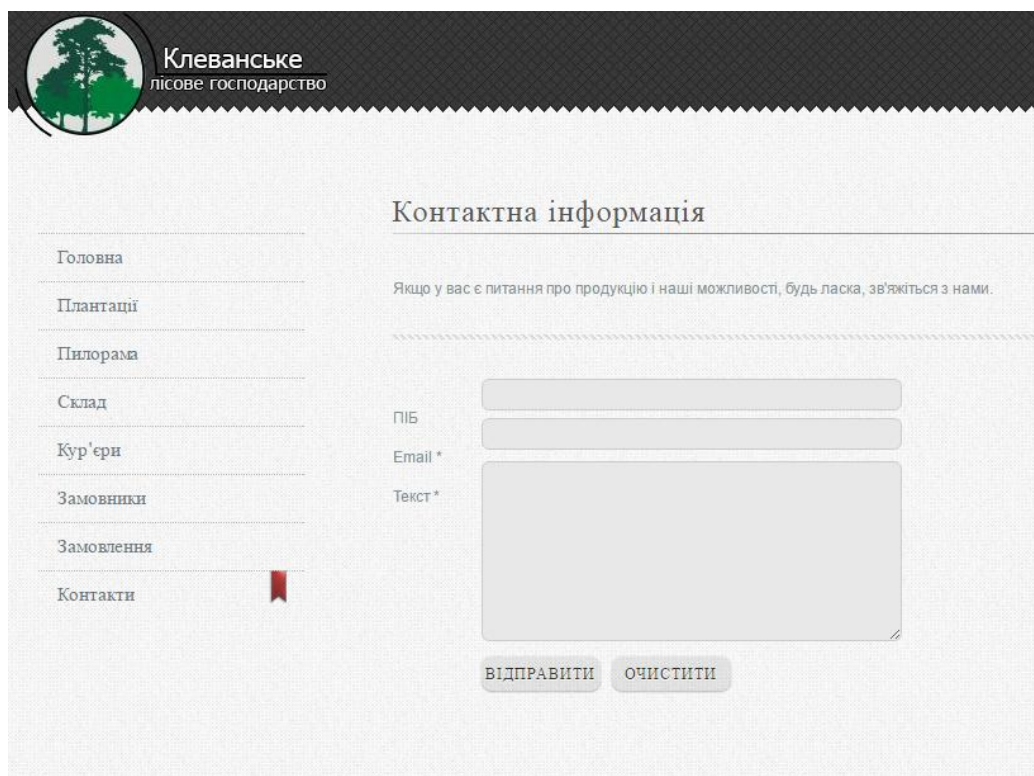
Замовлення

№	ПІБ замовника	ПІБ кур'єра	Матеріал	Кількість	Ціна	Шлях доставки	Сума за матеріал	Сума за дорогою	До оплати
6	Петренко Андрій	Шкляр Артем	Дошка	12	25	32	300	1440	1740
7	Гепешко Дмитро	Шилок Андрій	Дошка	12	30	23	360	1035	1395
8	Шнайдер Іван	Шилок Андрій	Дошка	12	30	231	360	10395	10755
9	Солома Дмито	Шкляр Григорій	Дошка	121	25	231	3025	1155	4180
10	Дацюк Андрій	Шилок Андрій	Дошка	121	25	2311	3025	11555	14580
12	Опейда Сергій	Шилок Андрій	вара	46	2	45	92	135	227
13	Козак Анатолій	Шкляр Григорій	Брус	12	100	23	1200	69	1269
15	Полохович Микола	Шилок Андрій	Дошка	23	30	34	690	102	792
16	Тюняєв Артем	Сазонов Григорій	Брус	12	100	23	1200	1035	2235
17	Мендалюк Віталій	Шилок Андрій	Брус	12	100	23	1200	69	1269
18	Чук Дмитро	Шкляр Григорій	вара	2	2	3	4	9	13
19	Петлюк Ілона	Шкляр Григорій	Брус	50	40	200	2000	1000	3000
20	Найтін Сашко	Шкляр Григорій	вара	4	2	58	8	2610	2618
21	Найтін Сашко	Шилок Андрій	вара	5	2	5	10	225	235
23	Шевченко Степан	Шкляр Григорій	Дошка	2	25	5	50	15	65
24	Степанюк Григорій	Шкляр Григорій	Дошка	01	30	2	30	10	40
25	Момотюк Євгеній	Шкляр Григорій	Дошка	3	25	2	75	10	85

Додати Редагувати Видалити Пошук

Рисунок 4.2 – Таблиця «Замовлення»

В розробленій програмній системі створено і протестовано форму зворотного зв'язку, яка дає змогу користувачеві написати повідомлення до державного підприємства «Клеванське лісове господарство», яке в свою чергу приходить на пошту директора державного підприємства. Зворотній зв'язок показаний на рис 4.3.



The screenshot shows a web interface for 'Клеванське лісове господарство'. On the left is a vertical menu with items: Головна, Плантації, Пилорама, Склад, Кур'єри, Замовники, Замовлення, and **Контакти** (highlighted with a red bookmark icon). The main content area is titled 'Контактна інформація' and contains the text: 'Якщо у вас є питання про продукцію і наші можливості, будь ласка, зв'яжіться з нами.' Below this is a form with three input fields: 'ПІБ', 'Email *', and 'Текст *'. At the bottom of the form are two buttons: 'ВІДПРАВИТИ' and 'ОЧИСТИТИ'.

Рисунок 4.3 – Контактна інформація

4.3. Інструкція користувача

Компоненти ПЗ

Даний програмний продукт було розроблено на мові програмування PHP. Розроблення бази даних проводилось на основі СУБД MySQL. Було використано набір інструментів Open Server. Open Server – це програмне середовище і портативна серверна платформа, створене для веб-розробників. Програмний комплекс має великий набір серверного програмного забезпечення, багатофункціональний продуманий інтерфейс, володіє можливостями з налаштування компонентів та адміністрування.

Встановлення

Для правильного функціонування програмного продукту необхідно встановити інтерпретатор мови програмування PHP, MySQL. Підключення бази даних. Найбільш важливим є підключення до мережі Інтернет.

Налаштування ПЗ.

Додаткові налаштування не потрібні

Базові функції ПЗ.

Були реалізовані наступні функції:

1. Здійснення замовлення;
2. Запис в базу даних;
3. Забзечення користувача інформацією;
4. Формування звітності;
5. Облік продукції на складі;
6. Перевірка наявності товару;
7. Список та кількість продукції;

Висновки до четвертого розділу

Було проведене функціональне тестування, де було визначено як позитивні так і негативні результати виконання тестів. Описано детально розгортання системи, для функціонування розробленої веб-орієнтованої програмної системи та інструкцію користувача для кращої продуктивності системи: компоненти пз, встановлення пз, налаштування пз, базові функції пз.

ВИСНОВКИ

Під час виконання дипломного проекту було проаналізовано предметну область та здійснено опис специфікації вимог. Також проведено огляд існуючих аналогів, де було визначено всі характеристики систем та на основі аналізу переваг і недоліків було спроектовано власну веб-систему. Наведено діаграми використання для акторів системи та здійснено їх розкадровку. Визначено всі функціональні та не функціональні вимоги до системи.

Визначено архітектуру системи, спроектовано базу даних, інтерфейс користувача та об'єктну модель програмного засобу. Спроектовано структурну схему програмної системи та здійснений її опис. Також проведено збір та аналіз вхідних даних, побудовано UML-діаграми станів до кожного модуля програмної системи. Визначено основні елементи бази даних та встановлено зв'язки між ними та подано інформаційну модель бази даних.

При розробці веб-орієнтованої програмної системи було обрано мову PHP, та розділено програмний продукт на вісім модулів. Реалізуючи базу даних було обрано систему управління базами даних MySQL.

Для коректної роботи веб-орієнтованої програмної систему було проведено функціональне тестування, де було визначено як позитивні так і негативні результати виконання тестів. Описано детально розгортання системи, для функціонування розробленої веб-орієнтованої програмної системи та інструкцію користувача для кращої продуктивності системи: компоненти пз, встановлення пз, налаштування пз, базові функції пз

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Синякевич Ігор Макарович «Екологічна і лісова політика»
2. Вакулюк Павло Гаврилович «Лісовідновлення та лісорозведення в Україні: монографія»
3. Голубчак Олексій Іванович «Формування молодих деревостанів Горган: результати дослідження та практичні рекомендації»
4. Пітер Пірс «ОСНОВИ ЕКОНОМІКИ ЛІСОВОГО ГОСПОДАРСТВА»
5. Анфилатов В.С. и др. Системный анализ в управлении. М.: Финансы и статистика, 2002.
6. Барышников А. А., Кузьмин А. М. Формы применения функционально-стоимостного анализа //«Машиностроитель», № 6, 2001.
7. Богданов А. А. Всеобщая организационная наука (тектология). В 3-х томах. М.: 1995. - Т 1-3.
8. Волкова В.Н., Денисов А.А. Основы теории систем и системного анализа. Учебник, издание 2. СПб.: Изд-во СПбГГУ, 1999.
9. Воцинин А.П, Сотиров Г.Р. Оптимизация в условиях неопределенности. МЭИ-СССР, Техника: НРБ (книга+дискета), 1989.
10. Денисов А.А., Колесников Д.Н. Теория больших систем управления. -Л.: Энергоатомидат, 1992.
11. Дивак М.П, Гладій Г.М. Методичні вказівки до самостійного вивчення розділу "Використання методів інтервального аналізу для моделювання економічних систем" курсу "Системний аналіз". Тернопіль, ТІНГ, 1994.
12. Калянов Г.Н. CASE - структурный системный анализ. -М.:Лори 1996. 242.
13. Катренко А.В. Системний аналіз об'єктів та процесів комп'ютеризації. Навчальний посібник. Львів, «Новий світ-2000», 2003. 424 с.
14. Основы системного анализа и проектирования АСУ./ под. ред.А.А.Павлова. -К: Вища школа, 1991.
15. Перегудов Ф.И, Тарасенко Ф.П. Введение в системный анализ. М: Высшая школа, 1992.

16. Саченко А.А., Володарский Е.Т. Методические указания к выполнению самостоятельной работы по дисциплинам: "Системный анализ", "Основы планирования эксперимента". Тернополь, ТИНХ,1990.
17. Системный анализ и структуры управления./ под ред. В.Г.Щорина -М.: Знание, 1975.
18. Герберт Шілдт. РНР для початківців. Пер. з англ. М: ЕКОМ Паблішерз 2007.
19. Герберт Шілдт. Повний довідник по РНР 4-е видання. Пер. з англ. М: видавничий дім «Вільямс» 2010.
20. Максим Кузнецов , Ігор Сімдянов - MySQL 5

Додаток А
Функціональне тестування

№	Кроки	Очікуваний результат	Фактичний результат	Пройшов/не пройшов	Середовище	Попередня умова
1	1. Заповнено форму реєстрації користувача. 2. Натиснуто на «Зареєструватись»	Створення нового користувача	Користувач створений	Пройшов	Windows 8 x64	Відкрита сторінка реєстрації
2	1. Заповнено форму авторизації. 2. Натиснуто на «Увійти»	Авторизація користувача	Користувач авторизований	Пройшов	Windows 8 x64	Користувач зареєстрований
3.	1. Заповнено форму додавання замовлення 2. Натиснуто «Додати»	Додавання замовлення	Замовлення додано	Пройшов	Windows 8 x64	Відкрита сторінка замовлення
4.	1. Заповнено форму додавання замовника 2. Натиснуто «Додати»	Додавання замовника	Замовник доданий	Пройшов	Windows 8 x64	Відкрита сторінка замовники
5.	1. Заповнено форму додавання кур'єра 2. Натиснуто «Додати»	Додавання кур'єра	Кур'єр доданий	Пройшов	Windows 8 x64	Відкрита сторінка кур'єри

6.	1. Заповнено форму додавання товару	Додавання товару	Товар доданий	Пройшов	Windows 8 x64	Відкрита сторінка складу
7.	1. Заповнено форму додавання плантації 2. Натиснуто «Додати»	Додавання плантації	Плантацію додано	Пройшов	Windows 8 x64	Відкрита сторінка плантації
8.	1. Заповнено форму додавання пилорами 2. Натиснуто «Додати»	Додавання пилорами	Пилораму додано	Пройшов	Windows 8 x64	Відкрита сторінка пилорами
9.	1. Відкрита сторінка відправлення замовлення 2. Натиснути «Відправити»	Зворотній зв'язок	Повідомлення відправлено	Пройшов	Windows 8 x64	Відкрита сторінка контакти

ДОДАТОК Б

UNIT-ТЕСТУВАННЯ

Лістинг коду тесту :

```
<body>
<table cellpadding="1" cellspacing="1" border="1">
<thead>
<tr><td rowspan="1" colspan="3">New Test</td></tr>
</thead><tbody>
<tr><td>open</td>
      <td>/ru/
product_id=1461&amp;filter=MANUFFUNAM=%D0%9C%D1%83%D0%B6%D1%81%D0%BA%D0
%BE%D0%B9=12=%D0%9F%D0%9E%D0%9B</td>
      <td></td>
</tr>
<tr><td>clickAndWait</td>
      <td>link=На складе</td>
      <td></td>
</tr>
<tr>
      <td>clickAndWait</td>
      <td>css=img[alt=&quot;21 Le Fou men&quot;]</td>
      <td></td>
</tr>
<tr>
      <td>click</td>
      <td>css=#button-cart &gt; span</td>
      <td></td>
</tr>
<tr>
      <td>clickAndWait</td>
      <td>xpath=(//button[@type='button'])[2]</td>
      <td></td>
</tr>
```

```
<tr>
  <td>clickAndWait</td>
  <td>css=img[alt=&quot;Удалить&quot;]</td>
  <td></td>
</tr>
<tr>
  <td>click</td>
  <td></td>
</tr>
</tbody></table>
</body>
</html>
```

ДОДАТОК В

ВИХІДНИЙ КОД ПРОГРАМНОГО ПРОДУКТУ

```
<title>Реалізація продукції ДП Клеванське лісове господарство</title>

</head>

<body>

<?php
    include "db.php";
    $result = mysql_query("INSERT INTO pulorama
(data_postavku,Nazva_plantasii,Tup_derevunu,kilkist_kubiv,sina_kub,Nazva_materialu) VALUES
('$_POST[data_postavku]', '$_POST[Nazva_plantasii]', '$_POST[Tup_derevunu]',
'$_POST[kilkist_kubiv]', '$_POST[sina_kub]', '$_POST[Nazva_materialu]')");
    if($result == true)
        { echo "Дані успішно додано."; }else
        { echo mysql_error(); }?>
<a href="index.php"> Назад </a>
</body>
</html>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ru" lang="ru">
<head>
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
<title>Реалізація продукції ДП Клеванське лісове господарство</title>
</head>
<body>
<?php
    include "db.php";

    $result = mysql_query("INSERT INTO plantasii
(Nazva_plantasii,data_posadku,data_vurybku,ploshcha_plantasii,lisnuk_PIB,mob_nom_lisnuka)
VALUES ('$_POST[Nazva_plantasii]', '$_POST[data_posadku]', '$_POST[data_vurybku]',
'$_POST[ploshcha_plantasii]', '$_POST[lisnuk_PIB]', '$_POST[mob_nom_lisnuka]')");

    if($result == true)
```

```

        { echo "Дані успішно додано."; } else

        {echo mysql_error(); }?>

<a href="index.php"> Назад </a>

</body>

</html>

<?php

$db = mysql_connect('localhost', 'hholik', '888000');

mysql_select_db('misha', $db);

mysql_query("set_client = 'utf8'");

mysql_query("set character_set_results = 'utf8'");

mysql_query("set collation_connection = 'utf8_general_ci'");

mysql_query("SET NAMES utf8");

?><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ru" lang="ru">

<head>

    <meta http-equiv="content-type" content="text/html; charset=utf-8" />

<title>Реалізація продукції ДП Клеванське лісове господарство</title>

</head>

<body>

<?php

include "db.php";

$result= mysql_query("DELETE FROM plantasii $_GET[type] WHERE id= '$_POST[id]'");

    if($result == true)

        { echo "Дані успішно видалено."; } else

        { echo mysql_error(); }

?><a href="index.php"> Назад </a>

```

```
</body>
```

```
</html>
```

```
<div class="tab">
```

```
<h1>Кур'єри</h1>
```

```
<table class = "table_kureru">
```

```
<tr>
```

```
<td class = "table_kureru_head1">№</td>
```

```
<td class = "table_kureru_head2">ПІБ кур'єра</td>
```

```
<td class = "table_kureru_head3">Телефон</td>
```

```
<td class = "table_kureru_head4">Назва авто</td>
```

```
<td class = "table_kureru_head5">Номер авто</td>
```

```
<td class = "table_kureru_head6">Ціна доставки</td>
```

```
</tr>
```

```
<?php
```

```
include "db.php";
```

```
$result = mysql_query("SELECT * FROM kureru");
```

```
if (mysql_num_rows($result) > 0)
```

```
{ $row = mysql_fetch_array($result);
```

```
do
```

```
{ echo ' <tr class = "table_kureru_head11">
```

```
<td class = "table_kureru_head1"><a><p>'. $row[Kod_kur]. '</p></a></td>
```

```
<td class = "table_kureru_head2"><a><p>'. $row[PIB_kur]. '</p></a></td>
```

```
<td class = "table_kureru_head3"><a><p>'. $row[Mob_nom_kur]. '</p></a></td>
```

```
<td class = "table_kureru_head4"><a><p>'. $row[Nazva_avto]. '</p></a></td>
```

```
<td class = "table_kureru_head5"><a><p>'. $row[nomer_avto]. '</p></a></td>
```

```
<td class =
```

```
"table_kureru_head6"><a><p>'. $row[sina_dost_za_km]. '</p></a></td>
```

```
</tr> }
```

```

        while ($row = mysql_fetch_array($result));
    } ?> </table>

        <!-- Ссылки на вызов модальных окон-->

    <a href="#win11111" class="button13">Додати</a>

    <a href="#win22222" class="button13">Редагувати</a>

    <a href="#win33333" class="button13">Видалити</a>

    <a href="#win44444" class="button13">Пошук</a>

    <!-- Модальное окно №1 -->

    <a href="#x" class="overlay" id="win11111"></a>

    <div class="popup">

        <form action="add_kur.php" method="post">

            ПИБ кур'єра:<br>

            <input type="text" name="PIB_kur"/></br></br>

            Номер телефону:<br>

            <input type="text" name="Mob_nom_kur"/></br></br>

            Назва авто:<br>

            <input type="text" name="Nazva_avto"/></br></br>

            Номер авто:<br>

            <input type="text" name="nomer_avto"/></br></br>

            Ціна доставки:<br>

            <input type="text" name="sina_dost_za_km"/></br></br>

            <input type="submit" name="submit" value="Додати"/>

            </form>

            <a class="close" title="Закрити" href="#close"></a>

        </div>

    <!-- Модальное окно №2 -->

    <a href="#x" class="overlay" id="win22222"></a>

```

```
<div class="popup">
```

Здесь вы можете разместить любое содержание, текст с картинками или видео!

```
<a class="close" title="Закрити" href="#close"></a>
```

```
</div>
```

```
<!-- Модальное окно №3 -->
```

```
<a href="#x" class="overlay" id="win33333"></a>
```

```
<div class="popup">
```

```
<form action="delete_kur.php" method="post">
```

```
  ID</br>
```

```
  <input type="text" name="id" size="10"/></br></br>
```

```
  <input type="submit" name="submit" value="Видалити"/>
```

```
</form>
```

```
<a class="close" title="Закрити" href="#close"></a>
```

```
</div>
```

```
<!-- Модальное окно №4 -->
```

```
<a href="#x" class="overlay" id="win44444"></a>
```

```
<div class="popup">
```

```
<a class="close" title="Закрити" href="#close"></a>
```

```
</div>
```

```
</div>
```

```
<div class="tab">
```

```
  <h1>Плантації</h1>
```

```
    <table class = "table_plant">
```

```
      <tr>
```

```
        <td class = "table_plant_head1">№</td>
```

```
        <td class = "table_plant_head2">Назва плантації</td>
```



```
<td class = "table_plant_head3">Дата посадки</td>
<td class = "table_plant_head4">Дата вирубки</td>
<td class = "table_plant_head5">Площа</td>
<td class = "table_plant_head6">ПІБ лісника</td>
<td class = "table_plant_head7">Телефон</td>
</tr>
```

```
<?php
```

```
include "db.php";
```

```
$result = mysql_query("SELECT * FROM plantasii");
```

```
if (mysql_num_rows($result) > 0)
```

```
{
```

```
    $row = mysql_fetch_array($result);
```

```
do
```

```
{
```

```
    echo '
```

```
<tr class = "table_plant_head11">
```

```
<td class = "table_plant_head1"><a><p>'. $row[id]. '</p></a></td>
```

```
<td class = "table_plant_head2"><a><p>'. $row[Nazva_plantasii]. '</p></a></td>
```

```
<td class = "table_plant_head3"><a><p>'. $row[data_posadku]. '</p></a></td>
```

```
<td class = "table_plant_head4"><a><p>'. $row[data_vurybku]. '</p></a></td>
```

```
<td class =
```

```
"table_plant_head5"><a><p>'. $row[ploshcha_plantasii]. '</p></a></td>
```

```
<td class = "table_plant_head6"><a><p>'. $row[lisnuk_PIB]. '</p></a></td>
```

```
<td class =
```

```
"table_plant_head7"><a><p>'. $row[mob_nom_lisnuka]. '</p></a></td>
```

```
</tr>
```

```

        ;
    }
    while ($row = mysql_fetch_array($result));
}

?>
</table>

<!-- Ссылки на вызов модальных окон-->

<a href="#win1" class="button13">Додати</a>
<a href="#win2" class="button13">Редагувати</a>
<a href="#win3" class="button13">Видалити</a>
<a href="#win4" class="button13">Пошук</a>

<!-- Модальное окно №1 -->

<a href="#x" class="overlay" id="win1"></a>
<div class="popup">
    <form action="add_res.php" method="post">
        Назва:<br>
        <input type="text" name="Nazva_plantasii"/><br><br>
        Дата посадки:<br>
        <input type="text" name="data_posadku"/><br><br>
        Дата вирубки:<br>
        <input type="text" name="data_vurybku"/><br><br>
        Площа:<br>
        <input type="text" name="ploshcha_plantasii"/><br><br>
        ПИБ лісника:<br>
        <input type="text" name="lisnuk_PIB"/><br><br>

```

Телефон:</br>

<input type="text" name="mob_nom_lisnuka"/></br></br>

<input type="submit" name="submit" value="Додати"/>

</form>

</div>

<!-- Модальное окно №2 -->

<div class="popup">

2Здесь вы можете разместить любое содержание, текст с картинками или видео!

</div>

<!-- Модальное окно №3 -->

<div class="popup">

<form action="delete_res.php" method="post">

ID</br>

<input type="text" name="id" size="10"/></br></br>

<input type="submit" name="submit" value="Видалити"/>

</form>

</div>

<!-- Модальное окно №4 -->

<div class="popup">

4Здесь вы можете разместить любое содержание, текст с картинками или видео!


```
</div>
```

```
</div>
```

```
<div class="tab">
```

```
<h1>Пилорама</h1>
```

```
<table class = "table_pulor">
```

```
<tr>
```

```
<td class = "table_pulor_head1">№</td>
```

```
<td class = "table_pulor_head2">Дата поставки</td>
```

```
<td class = "table_pulor_head3">Назва плантації</td>
```

```
<td class = "table_pulor_head4">Тип деревини</td>
```

```
<td class = "table_pulor_head5">Кількість</td>
```

```
<td class = "table_pulor_head6">Ціна</td>
```

```
<td class = "table_pulor_head7">Матеріал</td>
```

```
</tr>
```

```
<?php
```

```
include "db.php";
```

```
$result = mysql_query("SELECT * FROM pulorama");
```

```
if (mysql_num_rows($result) > 0)
```

```
{
```

```
    $row = mysql_fetch_array($result);
```

```
do
```

```
{
```

```
    echo '
```

```
<tr class = "table_pulor_head11">
```

```
<td class = "table_pulor_head1"><a><p>'.$row[Kod_postavku].</p></a></td>
```

```

        <td class = "table_pulor_head2"><a><p>'. $row[data_postavku]. '</p></a></td>
        <td class = "table_pulor_head3"><a><p>'. $row[Nazva_plantasii]. '</p></a></td>
        <td class = "table_pulor_head4"><a><p>'. $row[Tup_derevunu]. '</p></a></td>
        <td class = "table_pulor_head5"><a><p>'. $row[kilkist_kubiv]. '</p></a></td>
        <td class = "table_pulor_head6"><a><p>'. $row[sina_kub]. '</p></a></td>
        <td class =
"table_pulor_head7"><a><p>'. $row[Nazva_materialu]. '</p></a></td>
    </tr>
    ;
}
while ($row = mysql_fetch_array($result));
}

```

```
?>
```

```
</table>
```

```
<!-- Ссылки на вызов модальных окон-->
```

```
<a href="#win11" class="button13">Додати</a>
```

```
<a href="#win22" class="button13">Редагувати</a>
```

```
<a href="#win33" class="button13">Видалити</a>
```

```
<a href="#win44" class="button13">Пошук</a>
```

```
<!-- Модальное окно №1 -->
```

```
<a href="#x" class="overlay" id="win11"></a>
```

```
<div class="popup">
```

```
<form action="add_2.php" method="post">
```

```
Дата поставки:</br>
```

```
<input type="text" name="data_postavku"/></br></br>
```

Назва плантації:</br>

<input type="text" name="Nazva_plantasii"/></br></br>

Тип деревини:</br>

<input type="text" name="Tup_derevunu"/></br></br>

Кількість кубів:</br>

<input type="text" name="kilkist_kubiv"/></br></br>

Ціна за куб:</br>

<input type="text" name="sina_kub"/></br></br>

Назва матеріалу:</br>

<input type="text" name="Nazva_materialu"/></br></br>

<input type="submit" name="submit" value="Додати"/>

</form>

</div>

<!-- Модальное окно №2 -->

<div class="popup">

Здесь вы можете разместить любое содержание, текст с картинками или видео!

</div>

<!-- Модальное окно №3 -->

<div class="popup">

Здесь вы можете разместить любое содержание, текст с картинками или видео!

</div>

<!-- Модальное окно №4 -->

```
<a href="#" class="overlay" id="win44"></a>
```

```
<div class="popup">
```

```
43десь вы можете разместить любое содержание, текст с картинками или видео!
```

```
<a class="close" title="Закрити" href="#close"></a>
```

```
</div>
```

```
</div>
```

```
<div class="tab">
```

```
<h1>Склад</h1>
```

```
<table class = "table_sklad">
```

```
<tr>
```

```
<td class = "table_sklad_head1">№</td>
```

```
<td class = "table_sklad_head2">Назва матеріалу</td>
```

```
<td class = "table_sklad_head3">Кількість</td>
```

```
<td class = "table_sklad_head4">Розміри</td>
```

```
<td class = "table_sklad_head5">Ціна(шт)</td>
```

```
</tr>
```

```
<?php
```

```
include "db.php";
```

```
$result = mysql_query("SELECT * FROM sklad");
```

```
if (mysql_num_rows($result) > 0)
```

```
{
```

```
    $row = mysql_fetch_array($result);
```

```
do
```

```
{
```

```
    echo '
```

```
<tr class = "table_sklad_head11">
    <td class = "table_sklad_head1"><a><p>'. $row[Kod_mat]. '</p></a></td>
    <td class =
"table_sklad_head2"><a><p>'. $row[Nazva_materialu]. '</p></a></td>
    <td class = "table_sklad_head3"><a><p>'. $row[kilkist]. '</p></a></td>
    <td class = "table_sklad_head4"><a><p>'. $row[rozmiru]. '</p></a></td>
    <td class = "table_sklad_head5"><a><p>'. $row[sina_sht]. '</p></a></td>
</tr>
```

```
    ;
}
while ($row = mysql_fetch_array($result));
}
```

```
?>
```

```
</table>
```

```
<!-- Ссылки на вызов модальных окон-->
```

```
<a href="#win111111" class="button13">Додати</a>
```

```
<a href="#win222222" class="button13">Редагувати</a>
```

```
<a href="#win333333" class="button13">Видалити</a>
```

```
<a href="#win444444" class="button13">Пошук</a>
```

```
<!-- Модальное окно №1 -->
```

```
<a href="#x" class="overlay" id="win111111"></a>
```



```
<div class="popup">
  <form action="add_sklad.php" method="post">
    Назва матеріалу:<br>
    <input type="text" name="Nazva_materialu"/></br></br>
    Кількість:<br>
    <input type="text" name="kilkist"/></br></br>
    Розміри:<br>
    <input type="text" name="rozmiru"/></br></br>
    Ціна (шт):<br>
    <input type="text" name="sina_sht"/></br></br>
    <input type="submit" name="submit" value="Додати"/>
  </form>
  <a class="close" title="Закрити" href="#close"></a>
</div>
```

```
<!-- Модальное окно №2 -->
```

```
<a href="#" class="overlay" id="win222222"></a>
```

```
<div class="popup">
```

Здесь вы можете разместить любое содержание, текст с картинками или видео!

```
<a class="close" title="Закрити" href="#close"></a>
```

```
</div>
```

```
<!-- Модальное окно №3 -->
```

```
<a href="#" class="overlay" id="win333333"></a>
```

```
<div class="popup">
```

```
<form action="delete_sklad.php" method="post">
```

```
ID</br>
```

```
<input type="text" name="id" size="10"/></br></br>
```

```
<input type="submit" name="submit" value="Видалити"/>
```

```
</form>

<a class="close" title="Закрити" href="#close"></a>

</div>

<!-- Модальне окно №4 -->

<a href="#" class="overlay" id="win444444"></a>

  <div class="popup">

    43десь вы можете разместить любое содержание, текст с картинками или видео!

    <a class="close" title="Закрити" href="#close"></a>

  </div>

</div>

      <div class="tab">

        <h1>Склад</h1>

        <table class = "table_sklad">

          <tr>

            <td class = "table_sklad_head1">№</td>

            <td class = "table_sklad_head2">Назва матеріалу</td>

            <td class = "table_sklad_head3">Кількість</td>

            <td class = "table_sklad_head4">Розміри</td>

            <td class = "table_sklad_head5">Ціна(шт)</td>

          </tr>

        </table>

        <?php

          include "db.php";

          $result = mysql_query("SELECT * FROM sklad");

          if (mysql_num_rows($result) > 0)

            {

              $row = mysql_fetch_array($result);
```

```

do
{
    echo '

        <tr class = "table_sklad_head11">
        <td class = "table_sklad_head1"><a><p>'. $row[Kod_mat]. '</p></a></td>
        <td class =
"table_sklad_head2"><a><p>'. $row[Nazva_materialu]. '</p></a></td>
        <td class = "table_sklad_head3"><a><p>'. $row[kilkist]. '</p></a></td>
        <td class = "table_sklad_head4"><a><p>'. $row[rozmiru]. '</p></a></td>
        <td class = "table_sklad_head5"><a><p>'. $row[sina_sht]. '</p></a></td>
        </tr>

    ';
}
while ($row = mysql_fetch_array($result));
}

?>

</table>

<!-- Ссылки на вызов модальных окон-->

<a href="#win111111" class="button13">Додати</a>
<a href="#win222222" class="button13">Редагувати</a>
<a href="#win333333" class="button13">Видалити</a>

```

```
<a href="#win444444" class="button13">Пошук</a>
```

```
<!-- Модальное окно №1 -->
```

```
<a href="#x" class="overlay" id="win111111"></a>
```

```
<div class="popup">
```

```
<form action="add_sklad.php" method="post">
```

```
Назва матеріалу:</br>
```

```
<input type="text" name="Nazva_materialu"/></br></br>
```

```
Кількість:</br>
```

```
<input type="text" name="kilkist"/></br></br>
```

```
Розміри:</br>
```

```
<input type="text" name="rozmiru"/></br></br>
```

```
Ціна (шт):</br>
```

```
<input type="text" name="sina_sht"/></br></br>
```

```
<input type="submit" name="submit" value="Додати"/>
```

```
</form>
```

```
<a class="close" title="Закрити" href="#close"></a>
```

```
</div>
```

```
<!-- Модальное окно №2 -->
```

```
<a href="#x" class="overlay" id="win222222"></a>
```

```
<div class="popup">
```

```
2Здесь вы можете разместить любое содержание, текст с картинками или видео!
```

```
<a class="close" title="Закрити" href="#close"></a>
```

```
</div>
```

```
<!-- Модальное окно №3 -->
```

```
<a href="#x" class="overlay" id="win333333"></a>
```

```
<div class="popup">
```

```
<form action="delete_sklad.php" method="post">
```

```
ID</br>

<input type="text" name="id" size="10"/></br></br>

<input type="submit" name="submit" value="Видалити"/>

</form>

<a class="close" title="Закрити" href="#close"></a>

</div>

<!-- Модальное окно №4 -->

<a href="#x" class="overlay" id="win444444"></a>

  <div class="popup">

    <a class="close" title="Закрити" href="#close"></a>

  </div>

</div>

<div class="tab">

  <h1>Замовлення</h1>

  <table class = "table_zamovlennia">

    <tr>

      <td class = "table_zamovlennia_head1">№</td>

      <td class = "table_zamovlennia_head2">ПІБ замовника</td>

      <td class = "table_zamovlennia_head3">ПІБ кур'єра</td>

      <td class = "table_zamovlennia_head4">Матеріал</td>

      <td class = "table_zamovlennia_head5">Кількість</td>

      <td class = "table_zamovlennia_head6">Ціна</td>

      <td class = "table_zamovlennia_head7">Шлях доставки</td>

      <td class = "table_zamovlennia_head8">Сума за матеріал</td>

      <td class = "table_zamovlennia_head9">Сума за дорогу</td>

      <td class = "table_zamovlennia_head10">До оплати</td>

    </tr>
```

```

<?php

        include "db.php";

        $result = mysql_query("SELECT * FROM zamovlennia");

if (mysql_num_rows($result) > 0)

{

    $row = mysql_fetch_array($result);

    do

    {

        echo '

                <tr class = "table_zamovlennia_head11">

                    <td class =

"table_zamovlennia_head1"><a><p>'. $row[Kod_zamovlennia]. '</p></a></td>

                    <td class =

"table_zamovlennia_head2"><a><p>'. $row[PIB_zamovnuka]. '</p></a></td>

                    <td class = "table_zamovlennia_head3"><a><p>'. $row[PIB_kur]. '</p></a></td>

                    <td class =

"table_zamovlennia_head4"><a><p>'. $row[Nazva_mateialu]. '</p></a></td>

                    <td class = "table_zamovlennia_head5"><a><p>'. $row[kilkist]. '</p></a></td>

                    <td class = "table_zamovlennia_head6"><a><p>'. $row[sina_sht]. '</p></a></td>

                    <td class =

"table_zamovlennia_head7"><a><p>'. $row[shliah_dostavku_km]. '</p></a></td>

                    <td class =

"table_zamovlennia_head8"><a><p>'. $row[vsioغو_za_material]. '</p></a></td>

                    <td class =

"table_zamovlennia_head9"><a><p>'. $row[vsioغو_za_dorogu]. '</p></a></td>

                    <td class =

"table_zamovlennia_head10"><a><p>'. $row[Vsioغو_do_splatu]. '</p></a></td>

```

</tr>

;

}

while (\$row = mysql_fetch_array(\$result));

}

?>

</table>

<!-- Ссылки на вызов модальных окон-->

Додати

Редагувати

Видалити

Пошук

<!-- Модальное окно №1 -->

<div class="popup">

<form action="add_res.php" method="post">

111Назва матеріалу:</br>

<input type="text" name="Nazva_plantasii"/></br></br>

Кількість:</br>

<input type="text" name="data_posadku"/></br></br>

Розміри:</br>

<input type="text" name="data_vurybku"/></br></br>

Ціна (шт):</br>

<input type="text" name="ploshcha_plantasii"/></br></br>

ПІБ лісника:</br>

<input type="text" name="lisnuk_PIB"/></br></br>

Телефон:</br>

<input type="text" name="mob_nom_lisnuka"/></br></br>

<input type="submit" name="submit" value="Додати"/>

</form>

</div>

<!-- Модальное окно №2 -->

<div class="popup">

23десь вы можете разместить любое содержание, текст с картинками или видео!

</div>

