

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ**

Інститут (факультет) Автоматизації і комп'ютерних систем  
Кафедра Інформаційних систем

«До захисту в ЕК»  
Директор інституту(декан факультету)  
Форсюк А.В.  
(підпис) (прізвище та ініціали)

« \_\_\_ » \_\_\_\_\_ 2021 р.

«До захисту допущено»  
Завідувач кафедри  
Чумаченко С.М.  
(підпис) (прізвище та ініціали)

« \_\_\_ » \_\_\_\_\_ 2021 р.

**КВАЛІФІКАЦІЙНА РОБОТА**  
**НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ БАКАЛАВРА**

зі спеціальності 122 «Комп'ютерні науки»  
(код та назва спеціальності)

освітньо-професійної програми «Комп'ютерні науки та інформаційні технології»

на тему: “Розроблення інформаційної системи підтримки ремонтного супроводу ТОВ «Рибний Комбінат Херсон»”

Виконав: здобувач 4 курсу, групи 3

Ставаш Анна Євгенівна  
(прізвище, ім'я, по батькові повністю) (підпис)

Керівник Ліманська Наталія Володимирівна  
(прізвище, ім'я та по батькові повністю) (підпис)

Консультанти \_\_\_\_\_  
(прізвище та ініціали) (підпис)

\_\_\_\_\_ (підпис)

\_\_\_\_\_ (підпис)

Рецензент Смітюх Ярослав Володимирович  
(прізвище та ініціали) (підпис)

Засвідчую, що в цій кваліфікаційній роботі немає запозичень із праць інших авторів без відповідних посилань.

Здобувач \_\_\_\_\_  
(підпис)

Київ - 2021р.

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ

Інститут (факультет) Автоматизація і комп'ютерних систем

Кафедра Інформаційних систем

Освітній ступінь Бакалавр

Спеціальність 122 «Комп'ютерні науки»

(код і назва)

Освітньо-професійна програма «Комп'ютерні науки»

(назва)

**ЗАТВЕРДЖУЮ**

Завідувач кафедри

Інформаційних систем

С.М.Чумаченко

“\_\_\_” \_\_\_\_\_ 20\_\_ року

**ЗАВДАННЯ**

**НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА**

Ставаш Анни Євгеніївни

(прізвище, ім'я, по батькові)

1. Тема роботи «Розроблення інформаційної системи підтримки ремонтного супроводу ТОВ «Рибний Комбінат Херсон»»

керівник роботи Ліманська Наталія Володимирівна

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “29” 04 2021 року №248-кс

2. Строк подання здобувачем роботи 31.05.2021 року

3. Вихідні дані до роботи Дані про роботу відділу, посадові інструкції, нормативно-правові акти, приклад заявок які поступають до відділу.

4.

Зміст пояснювальної записки (перелік питань, які потрібно розробити) системний аналіз діяльності відділу, створення функціональної моделі роботи відділу, постановка задачі, розробка системи для відділу, інструкція користувача, заходи з охорони праці, висновки .

5. Перелік графічного матеріалу

функціональна модель діяльності, логічна та фізична моделі бази даних, схема бази даних в середовищі SQL Server, інтерфейс користувача .

## 6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв
Вступ	Ліманська Н.В.		
1	Ліманська Н.В.		
2	Ліманська Н.В.		
3	Ліманська Н.В.		

7. Дата видачі завдання \_\_\_\_\_ 25 березня 2021 року \_\_\_\_\_

## КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів виконання кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Написання вступу		виконано
2	Аналіз діяльності відділу		виконано
3	Виконання функціональної моделі		виконано
4	Вибір програмного забезпечення для розробки системи		виконано
5	Розробка технічного завдання на проектування		виконано
6	Розробка інтерфейсу користувача		виконано
7	Написання інструкції користувача		виконано
8	Розробка заходів з охорони праці		виконано
9	Оформлення пояснювальної записки		виконано
10	Оформлення презентації		виконано

**Здобувач**

\_\_\_\_\_ (підпис)

**Керівник роботи**

\_\_\_\_\_ (підпис)

**Ставаш А.Є.**

\_\_\_\_\_ (прізвище та ініціали)

**Ліманська Н.В.**

\_\_\_\_\_ (прізвище та ініціали)

## АНОТАЦІЯ

Дипломна робота на тему « Розроблення інформаційної системи підтримки ремонтного заводу ТОВ «Рибний Комбінат Херсон»»

Складається з 94 сторінок, 6 таблиць, 34 малюнків, 4 додатки та 15 літературних джерел.

У даній дипломній роботі показані результати дослідження механізму роботи ремонтного відділу заводу, представлено пояснення необхідності розробки інформаційної системи. Для реалізації функцій, описаних у функціональній моделі, здійснено вибір програмного забезпечення, розроблена інструкція користувача. Описані заходи охорони праці. Проведений розрахунок економічної ефективності після впровадження нової системи на підприємстві.

Головною задачею системи є збільшення продуктивності, скорочення часу на обробку заявок та розподіл роботи між бригадами. Метою даної роботи є створення системи, за допомогою якої головний механік зможе отримувати всю необхідну інформацію та передавати її підрозділами.

Функціональна модель роботи відділу збуту розроблена за допомогою CASE-засобу AllFusionProcessModeler. Логічна а та фізична моделі даних розроблені за допомогою CASE-засобу, проектування та кодогенерації — AllFusionERwinDataModeler r7.3. Інтерфейс програми розроблено за допомогою CMS.

Ключові слова: ВІДДІЛ РЕМОНТНОГО СУПРОВОДУ ТОВ «РИБНИЙ КОМБІНАТ ХЕРСОН», ІНФОРМАЦІЙНА СИСТЕМА, ЛОГІЧНА МОДЕЛЬ, ФІЗИЧНА МОДЕЛЬ, БАЗА ДАНИХ, СУБД, SQL SERVER.

## **SUMMARY**

Thesis on the topic "Creation of an information system to support the repair plant of LLC" Fish Factory Kherson ""

It consists of 94 pages, 6 tables, 34 figures, 4 appendices and 15 references.

This thesis shows the results of the study of the mechanism of the repair department of the plant, provides an explanation of the need to develop an information system. To implement the functions described in the functional model, the software is selected, the user manual is developed. Measures of labor protection are described. Calculation of economic efficiency after introduction of new system at the enterprise is carried out.

The main task of the system is to increase productivity, reduce the time for processing applications and the distribution of work between teams. The purpose of this work is to create a system through which the chief mechanic will be able to obtain all necessary information and transmit it to departments.

The functional model of the sales department was developed using the CASE-tool AllFusionProcessModeler. Logical and physical data models are developed using CASE-tool, design and code generation — AllFusionERwinDataModeler r7.3.

**Keywords:** REPAIR MAINTENANCE DEPARTMENT OF KHERSON FISH PLANT LLC, INFORMATION SYSTEM, LOGICAL MODEL, PHYSICAL MODEL, DATABASE, SQLU SUDIO.

## Зміст

ВСТУП.....	8
РОЗДІЛ 1.СИСТЕМНИЙ АНАЛІЗ ОБ'ЄКТУ.....	9
1.1.Характеристика об'єкту автоматизації .....	9
1.2. Опис відділу механіки. ....	10
1.3. Взаємодія механічного відділу з іншими відділами заводу.....	12
1.4.Проблеми та задачі, які були виявлені.....	13
1.3.Функціональне програмування .....	13
1.3.1.Загальний опис програми IDEF0 .....	13
1.3.2.Розробка функціональної моделі для ТОВ «Рибний Комбінат Херсон» .....	14
1.4.Аналіз існуючих аналогів розробки .....	16
1.4.1. 1С Підприємство. ....	16
1.4.2.Система «ДЖИТ».....	16
1.4.3. Технологія AJAX.....	17
1.4.4. Мова HTML.....	18
1.4.5.Мова JavaScript.....	19
1.4.6. Результат порівняння аналогів систем. ....	20
1.5. Обґрунтування доцільності проектування й розроблення інформаційної системи для заводу.....	20
1.6. Концептуальна модель майбутньої системи.....	21
1.7. Постановка задачі.....	21
1.7.1. Призначення та мета створення сайту .....	21
1.7.3. Функції які повинна виконувати система .....	22
РОЗДІЛ 2. ТЕХНІЧНЕ ЗАВДАННЯ НА ПРОЕКТУВАННЯ .....	23
2.1. Загальні положення .....	23
2.2 Призначення і цілі створення .....	23
2.3. Характеристика об'єкта автоматизації .....	24
2.4. Вимоги до системи.....	24
2.5. Склад та зміст по створенню системи.....	32

2.6. Порядок контролю і приймання системи. ....	32
РОЗДІЛ 3. ОПИС КОМПЛЕКСУ ЗАДАЧ АВТОМАТИЗАЦІЇ .....	34
3.1. Використані програмне за забезпечення та технології .....	34
3.2. Мова PHP та CMS система .....	34
3.3. Інтерфейс користувача .....	40
3.4. Проектування БД .....	54
3.4. Техніко-економічне обґрунтування .....	55
3.4.1. Визначення розміру оплати праці .....	55
3.4.2. Витрати, пов'язані з розробкою програми на ПК .....	57
РОЗДІЛ 4 . ОХОРОНА ПРАЦІ .....	61
4.1. ЗАГАЛЬНА ІНФОРМАЦІЯ ПО ОХОРОНУ ПРАЦІ .....	61
4.2. Охорона праці користувачів ПК .....	61
4.3. Заходи техніки безпеки на заводі .....	64
4.4. Пожежна безпека відділу .....	65
ВИСНОВОК .....	67
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	68
ДОДАТОК А .....	69
ДОДАТОК Б .....	74
ДОДАТОК В .....	76
ДОДАТОК Г .....	79

## ВСТУП

Інформаційна система, по-перше забезпечує обмін інформацією між підрозділами та робітниками заводу або підприємства, по-друге зберігає та видає по необхідності інформацію користувачу. Вона має в собі сукупність методів які дозволяють зберігати, передавати та обробляти інформацію .

Потрібність даної роботи є в тому що, велика кількість інформації та документів які мають бути систематизовані та подані у відповідному вигляді користувачеві. Надходження заявок на ремонт , технічний стан обладнання, замовлення нових інструментів або обладнання ,розставлення пріоритетності заявок та багато іншої потрібної інформації, використання якої вимагає багато часу та зусиль працівників підприємства.

Об'єктом дослідження моєї кваліфікаційної роботи є ТОВ «Рибний Комбінат Херсон»

Предмет дослідження — організації роботи відділу механіка, а саме ремонтного супроводу на заводі.

# РОЗДІЛ 1.СИСТЕМНИЙ АНАЛІЗ ОБ'ЄКТУ. АВТОМАТИЗАЦІЇ ТА ПОСТАНОВКА ЗАДАЧІ НА ПРОЕКТУВАННЯ

## 1.1. Характеристика об'єкту автоматизації

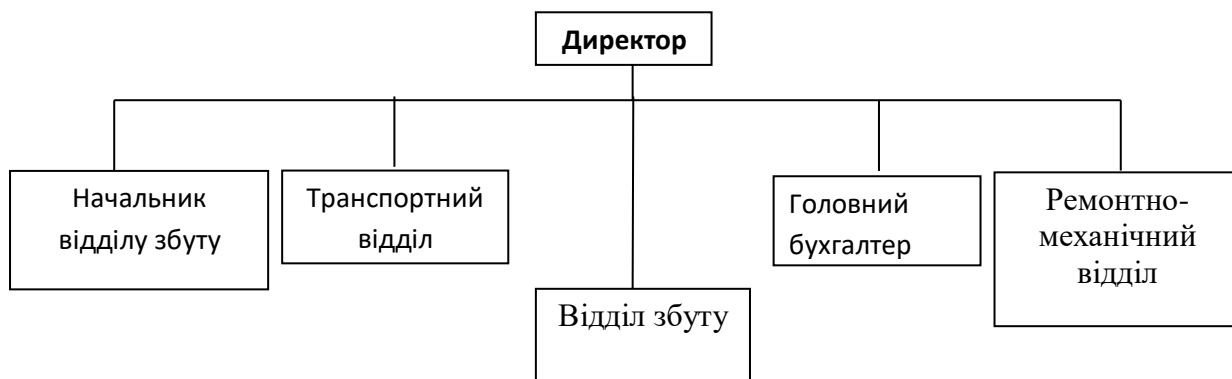
ТОВ «Рибний Комбінат Херсон» - підприємство, яке займається торгівельною діяльністю рибою та іншими морепродуктами.

Предметом діяльності ТОВ «Рибний Комбінат Херсон»:

- виготовлення, продаж та перевезення власної продукції ;
- виробництво, збереження, переробка і реалізація харчової продукції;
- торгівельна діяльність рибою;
- торгівельно-закупівельна та торгово-посередницька діяльність;
- створення оптово-роздрібних підприємств торгівлі
- доставка замовлень власної продукції;

Основними видами продукції є: риба, ракоподіні, молюски . Загальний асортимент продукції становить близько 50 найменувань. Вся продукція виробляється у відповідності до державних стандартів, проходить перевірку за різними фізико-хімічними , мікробіологічними показниками та знаходиться під ветеринарним контролем.

Лінійна (елементарна) структура управління складається із взаємопідпорядкованих органів у вигляді ієрархічної драбини. Кожен учасник підпорядковується тільки одному керівнику і всі зв'язки з вищими рівнями управління відбуваються через нього.



**Рис. 1.** Організаційна структура ТОВ «Рибний Комбінат Херсон»

На рибзаводі значну увагу приділяють переоснащенню і реконструкції виробничих цехів для збільшення кількості продукції та розширення заводу. Це прерогатива спеціалістів та голови підприємства.

На сьогодні, важливим завданням у діяльності рибзаводу є оснащення виробництва і заміна застарілого обладнання, а також розроблення та впровадження нових видів асортименту продукції.

## **1.2. Опис відділу механіки.**

В даній дипломній роботі розглядається саме ремонтно-механічний відділ.

Механічний відділ ТОВ «Рибний Комбінат Херсон», є важливою частиною роботи всього заводу. Від коректної роботи відділу залежить швидкість та якість роботи заводу. Відділ приймає заявки з усього заводу, оброблює та виконує їх по пріоритетності.

### Основні функції структурного підрозділу:

- контроль за роботою та налагодженням технологічного обладнання
- проведення всіх видів ремонту технологічного устаткування
- монтаж та демонтаж обладнання та комплектуючих.
- напрямок виробництва у бік зменшення простоїв обладнання
- зниження вартості ремонтів

- контроль наявності запасу необхідних запасних частин
- пошук і підбір запасних частин
- забезпечення правильного та економного витрачання грошових коштів і матеріалів , призначених на ремонт обладнання.
- розробка підрозділам норм витрати на основні і допоміжні матеріали і комплектуючі вироби для ремонту і технологічного обслуговування устаткування.
- контроль за правильністю використання і завантаження ремонтних цехів і служб цехових механізмів .
- контроль за правильною організацією «мастильного господарства» , застосуванням масел для змащення механізмів та обладнання .
- облік наявності та руху устаткування , що значиться на балансі основної діяльності .
- забезпечення впровадження досягнень нової техніки і передових методів роботи в галузі ремонту і експлуатації технологічного та підйомно -транспортного устаткування.
- подання висновків щодо раціоналізаторських пропозицій та винаходів і забезпечення впровадження прийнятих пропозицій .

Працівники відділу: головний-механік, токар, слюсар-ремонтник, електрозварювальник, слюсар, фрезерувальник.

Права керівника відділу:

- Давати вказівки з експлуатації та ремонту устаткування.
- Приймати рішення про внесення змін у технологію обслуговування устаткування .
- Вимагати від керівників виробничих і технічних підрозділів:
  - Виконання запропонованих норм експлуатації та обслуговування обладнання;
  - Своєчасного надання відомостей про порушення технології обслуговування обладнання;

- Негайного повідомлення про поломки устаткування ;
- Здійснювати примусовий ремонт ( зупиняти роботу обладнання) у разі порушення правил експлуатації обладнання;
- Забороняти роботу на несправному обладнанні.
- Зупиняти роботу обладнання в разі загрози аварії або нещасного випадку.
- Доручати окремим структурним підрозділам підприємства проведення робіт з технічного обслуговування обладнання .
- Брати участь у розробці технічних умов , інструкцій.
- подавати керівництву підприємства: пропозиції про заохочення працівників, що відрізнялися і про притягнення до дисциплінарної відповідальності працівників , які не дотримуються трудової і виробничу дисципліну.

### **1.3. Взаємодія механічного відділу з іншими відділами заводу**

#### 1.3 відділом кадрів:

Отримує: інформацію про обмеженість робочих місць; періодична звітність про кількість робочих місць;

Надає: заявки на оформлення нових співробітників; дані співробітників;

#### 2.3 бухгалтерією:

Отримує: інформацію про наявність основних та обігових коштів у зіставленні з нормативом; повідомлення за нарахування заробітної плати ; підтвердження про оплату нового обладнання;

Надає: акти на списання обладнання з балансу; акти на передачу або продаж обладнання іншим організаціям; звіти про закупівлю нового обладнання; звіти про списання нового обладнання;

### 3.3 відділом збуту

Отримує: інформацію про проблеми з обладнанням; заявки на ремонт;

Надає: підтримку роботи всього обладнання.

#### **1.4.Проблеми та задачі, які були виявлені**

Оскільки рівень автоматизації роботи відділу збуту знаходиться на низькому рівні після проведення аналізу наявної бази можна зробити висновок, що підрозділ потребує розробки інформаційної системи для підтримки діяльності та підвищення ефективності виконання поставлених завдань.

Мета роботи – моделювання інформаційної системи, яка відображатиме роботу відділу механіки рибзаводу. Ціль – автоматизувати роботу відділу шляхом створення програмного продукту для полегшення роботи персоналу.

Система повинна зберігати дані в зручному для користувача вигляді, мати зручний інтерфейс, повинна бути включена можливість додавання даних, розставлення пріоритетності робіт .В якості СУБД використовується SQL Server.

#### **1.3.Функціональне програмування**

##### **1.3.1.Загальний опис програми IDEF0**

Функціональна модель бізнес-процесів розроблюється за допомогою стандарту IDEF0, тому що ця методологія дає змогу описати башато взаємозалежних процесів та функцій. Програма дає можливість досліджувати всі функції організації, та можна не пов'язувати їх з об'єктами, які реалізують.

У стандарті IDEF0 за допомогою входу показують об'єкти – інформаційні та матеріальні потоки, які перетворюються в бізнес-процесі.

За допомогою управління показуються об'єкти – матеріальні та інформаційні потоки, які перетворюються на процесі, та потрібні для його виконання.

Використовуючи механізми програми можливо відобразити інструменти та ресурси, за допомогою яких бізнес-процес реалізується.

### **1.3.2.Розробка функціональної моделі для ТОВ «Рибний Комбінат Херсон»**

Функціональна модель в стандарті IDEF0 для заводу була створена таким програмним засобом як, AllFusion Process Modeler. Моя функціональна модель повністю описує процеси роботи у відділі механіки на заводі.

Модель складається з 6 рівнів, при цьому перший має назву A0, далі формуються набір взаємопов'язаних діаграм ,які відображають всі процеси роботи відділу механіки.(Додаток А).

#### Вхідні дані:

- Заявки;
- Дані про стан техніки;
- Нове обладнання.

#### Вихідні дані:

- Звітність;
- Списане обладнання.

#### Керування:

- Працівники;
- ПО;
- Інструменти.

#### Управління:

- Інструкція;
- Нормативно-правові акти;

Основні процеси організації роботи механічного відділу(Додаток А; рис.2):

- ✓ Технічна підтримка;
- ✓ Отримання та обробка отриманих заявок;
- ✓ Оцінка роботи;
- ✓ Розподіл сили та пріоритетності;
- ✓ Виконання заявок;
- ✓ Інвентаризація;
- ✓ Формування звітів.

Далі декомпозується кожний процес(Додаток А;рис.3;4;5):

#### Оцінка роботи:

- Перевірка можливості виконання заявок;
- Формування заявки на оцінку обладнання;
- Передача можливих до виконання заявок.

#### Розподіл сили та пріоритетність:

- Оцінка фронту роботи;
- Розподіл пріоритетності;
- Розподіл по бригадам.

#### Формування звітів:

- Збір та перевірка звітів;
- Формування звітності.

## **1.4. Аналіз існуючих аналогів розробки**

### **1.4.1. 1С Підприємство.**

1С:Підприємство- дуже відоме програмне забезпечення. На даний момент широко використовуване в середньому та малому бізнесі для ведення різних обліків на підприємстві. Вважаю головним плюсом цієї програми те, що є воно бюджетне та багато-функціональне.

Має в собі можливості бухгалтерського та податкового обліку, які мають норми тієї країни в якій вона використовується.

### **1.4.2. Система «ДЖИТ»**

Це нова форма організації «just in time», котра дає поняття виробництву «виробництво точно в строк». Її фундаментальний сенс: нуль запасів, нуль відмов, нуль дефектів. Детальніше ДЖИТ перед ставлять собою технологію, яка має на увазі зниження запасу матеріалів завдяки подачі деталей на кожному ділянці виробництва в той момент, коли вони там потрібні. Ще ця технологія називається «точно вчасно». Тут немає сильних особливостей, якщо говорить просто, то це боротьба за ліквідацію складів комплектуючих виробів та ідеально поставлене постачання з боку постачальників. Складування про запас - це викинуті гроші, час і нераціональне використання площі.

Однак перехід на ДЖИТ - завдання непросте. Ця система кидає виклик традиційній організації виробництва, надаючи особливо сильний вплив на чотири його області:

- управління матеріально технічним постачанням;
- структуру виробничого центру;
- відносини «постачальник споживач»; відносини «управління - безпосереднє виробництво».

В кінцевому підсумку система ДЖИТ спрямована на інтеграцію та автоматизацію кожної стадії виробництва, починаючи з проектування і аж до гарантійного обслуговування споживача.

### **Мінуси ДЖИТ:**

- Глобальність;
- Характерні риси цієї тенденції
- Проектування з урахуванням можливостей виробництва, автоматизоване виробництво й контроль якості з використанням ЕОМ.

Фактично фахівці з ДЖИТ виступають навіть проти його впровадження до тих пір, поки перераховані вище вимоги не будуть повністю виконані;

При впровадженні системи ДЖИТ у виробництв і виникали труднощі зі залученням постачальників, діяльність котрих не відповідає поставленим вимогам

### **1.4.3. Технологія AJAX**

AJAX - це аббревіатура, яка означає Asynchronous Javascript and XML. Насправді, AJAX не є новою технологією, так як і Javascript, і XML існують вже досить тривалий час, а AJAX - це синтез позначених технологій. AJAX найчастіше асоціюється з терміном Web 2.0 і підноситься як новітнє Web-додаток.

При використанні AJAX немає необхідності оновлювати кожен раз всю сторінку, так як оновлюється тільки її конкретна частина. Це набагато зручніше, так як не доводиться довго чекати, і економічніше, так як не всі мають безлімітним інтернетом. Правда в цьому випадку, розробнику необхідно стежити, щоб користувач був в курсі того, що відбувається на сторінці. Це можна реалізувати з використанням індикаторів завантаження, текстових повідомлень про те, що йде обмін даними з сервером. Необхідно також розуміти, що не всі

браузери підтримують AJAX (старі версії браузерів і текстові браузері). Плюс Javascript може бути відключений користувачем. Тому, не слід зловживати використанням технології і вдаватися до альтернативних методів подання інформації на Web-сайті.

Узагальнимо гідності AJAX:

- Можливість створення зручного Web-інтерфейсу
- Активна взаємодія з користувачем
- Часткова перезавантаження сторінки, замість повної
- Зручність використання

AJAX использует два метода работы с веб-страницей: изменение Web-страницы не перезагружая её, и динамическое обращение к серверу. Друге може реалізовуватися декількома способами, зокрема, XMLHttpRequest.

#### **1.4.4. Мова HTML**

HTML- це мова розмітки гіпертексту (від англ. HyperText Markup Language).

Дана мова застосовується для створення веб-сторінок. Він інтерпретується (обробляється) браузером і відображається у вигляді документа в зручній для людини формі.

HTML - це невід'ємна складова і основа практично будь-який веб-сторінки. Мова HTML в першу чергу виступає як засіб логічної розмітки сторінки. Саме HTML дозволяє нам наділяти вміст сторінки певним змістом, а реалізується це за допомогою так званих тегів.

Теги - це спеціальні маркери, які певним чином інтерпретуються браузером. Суть тегів в тому, що вміст сторінки, укладену в різні теги, по-різному обробляється браузером.

Скажімо, ми можемо зробити висновок контент (вміст) сторінки в тег параграфа, і дане вміст буде вважатися браузером параграфом.

Ми можемо зробити висновок вміст в тег списку, і тоді інформація всередині нього буде інтерпретуватися як список. Ми можемо помістити контент

в тег таблиці, і вміст в підсумковому документі буде відображатися у вигляді таблиці і т.д.

Іншими словами, кожен з HTML-тегів, призначених для розмітки текстової інформації, надає цій інформації певний сенс.

Мова HTML має досить тривалу історію розвитку, і за цей час зазнав значних змін. Велика частина змін пов'язана з додаванням в мову нових тегів і «виведенням з обігу» застарілих. Для різних версій HTML передбачено різний оголошення типу документа. Розглянемо кілька прикладів оголошення типу HTML 4.01 (на даний момент його можна вважати вже застарілим).

### **1.4.5. Мова JavaScript**

JavaScript - це мова програмування, який знайшов найбільш широке застосування в браузерях для додання інтерактивності веб-сторінок. Важливо розуміти, що це далеко не єдина область застосування JavaScript, проте в рамках нашої теми найбільш доречно розглядати саме такий варіант використання цієї мови.

Основним завданням JavaScript в розглянутому нами контексті є маніпулювання елементами DOM-моделі web-сторінки.

DOM - це об'єктна модель документа (від англ. Document Object Model). Згідно DOM, документ (наприклад, веб-сторінка) може бути представлений у вигляді дерева об'єктів, що володіють рядом властивостей, які дозволяють виробляти з ним різні маніпуляції:

- отримання вузлів;
- зміна вузлів;
- зміна зв'язків між вузлами;
- видалення вузлів.

Саме ці маніпуляції і дозволяє нам здійснювати над елементами сторінки мову JavaScript. Для додавання JavaScript-коду на сторінку можна використовувати тег script. Його рекомендується поміщати всередині тега head, хоча це і не обов'язково. Контейнером script в одному документі може бути

скільки завгодно. При цьому атрибут «type = 'text / javascript'» вказувати необов'язково, тому що значення javascript є значенням за замовчуванням.

#### **1.4.6. Результат порівняння аналогів систем.**

Вище розглянуті системи являються або повноцінними додатками або WEB-ресурсами, для Кожна з перелічених систем мають як позитивні так і негативні якості. Використовуючи одну з цих систем є можливість задовольнити потреби переважної більшості невеликих підприємств або підприємців.

Вважаю що краще створити інформаційну систему з легким для розуміння інтерфейсом та легше для опанування для людей. Наприклад сайт на якому можливо заходити під своїм ім'ям є досить щоденною процедурою на наш час.

Також деякі з цих програм мають безкоштовні версії, а платні, мають ціну, нижчу за ціну європейських програм подібного рівня, оскільки більшість орієнтована на користувачів з пострадянського простору, де доходи підприємств нижче європейських, а отже і сума, яку підприємство готове віддати за будь-який продукт буде меншою.

### **1.5. Обґрунтування доцільності проектування й розроблення інформаційної системи для заводу**

На ТОВ «Рибний Комбінат Херсон» для відділу механіки взагалі немає, навіть мінімальної інформаційної системи. Всі аналоги інформаційних систем, які були вже перераховані, вони є занадто громіздкі, та мають забагато функцій для одного відділу, які можуть тільки уповільнити та ускладнити користування цими системами.

Вважаю, що система має бути простою для розуміння для швидкого користування, тому враховуючи усе, є актуальним завданням розробити власну інформаційну систему для підтримки ремонтного супроводу ТОВ «Рибний Комбінат Херсон».

## **1.6. Концептуальна модель майбутньої системи**

Інформаційна система створена з метою покращення та прискорення роботи ремонтного відділу на заводі ТОВ «Рибний Комбінат Херсон».

Маючи концептуальну модель «ТО-ВЕ» та додаючи у використання нові інформаційну модель ,можна побачити її вже на моделі «As-IS», яка надає змогу швидше робітникам отримувати завдання, а головному інженеру робити звіти та пріоритетність.

Основні зміна, та час використання ІС можна побачити в додатку Б нарисунках.

Сайт для відділу надасть можливість головному механіку слідкувати за станом заявки, тобто в стані опрацювання, або вже зроблена та чекає на оформлення звіту.

Сайт для робітників бригади дає їм можливість авторизувавшись одразу бачити фронт роботи на конкретну зміну, та бачити кращу послідовність виконання заявок.

Проаналізувавши функціональну модель ТОВ «Рибний Комбінат Херсон», були виявлені такі проблеми як:

- відсутність контролю стану заявки;
- відсутність пріоритетності виконання заявок;
- відсутність додаткових інформації про заявку.

Після впровадження ІС всі проблеми вище вказані мають бути дискваліфіковані.

## **1.7. Постановка задачі**

### **1.7.1. Призначення та мета створення сайту**

Створюваний сайт ,призначений для кращої роботи відділу. Для швидкості обробки заявок, для слідкування за станом обладнання, за станом обробки замовлення та вибору бригад або співробітників на окремі фронти роботи.

Для основних цілей ІС відноситься отримання заявок та передання його до відділу, авторизація робітників, формування звітів, розставлення пріоритетів виконання заявок, а це все значно скорочує час яке раніше використовував головний механік відділу.

При такому масштабуванні, вже обов'язковим буде механізм входу до системи з різними повноваженнями в залежності від того хто входить до системи, для того щоб забезпечити різних працівників, різними повноваженнями в залежності від їх потреб.

### **1.7.2. Вимоги для створеного сайту**

Створювана система повинна забезпечувати підтримку прийняття рішень для головного механіка при пріоритетності заявок ,при формування звітів, та перевірки стану обладнання.

Головне завдання сайт це легке отримання завдання для співробітників заводу при цьому бути в будь-якому місці на заводі,лише потрібно мати свій логін та пароль.

Має бути зручний та легкий інтерфейс для розуміння людини будь-якого віку. Також система має надавати швидкий доступ для співробітників заводу

Для проектування функціональної моделі організації навчального процесу було обрано програмний засіб CA ERwin Process Modeler (раніше BPwin).

### **1.7.3. Функції які повинна виконувати система**

Сайт, який розробляється для відділу заводу, повинен відображувати, а також реалізовувати наступні функції:

- забезпечити авторизацію співробітника;
- забезпечити відображення заявок ,які тільки відправили до відділу;
- забезпечити відображення статусу заявки ;
- формування звітів виконаних заявок.

## РОЗДІЛ 2. ТЕХНІЧНЕ ЗАВДАННЯ НА ПРОЕКТУВАННЯ

### 2.1. Загальні положення

1. Найменування системи : «Інформаційна система підтримки ремонтного супроводу ТОВ «Рибний Комбінат Херсон»».
2. Результати робіт зі створення системи оформлюються згідно з вимогами ДСТУ на відповідні етапи розробки. Порядок оформлення і передачі результатів у даному випадку визначається змістом і календарним планом виконання розробки.
3. У випадку необхідності на наступних стадіях робіт по створенню системи окремі положення можуть уточнюватися і розвиватися.

### 2.2 Призначення і цілі створення

#### 1. Призначення системи

Система призначена для:

Система призначена для:

- автоматизація всіх процесів, які виникають при отриманні заявок;
- автоматизація процесів при розгляді заявок;
- пришвидчення процесу ремонту обладнання;
- зручне користування веб інтерфейсом;
- надання доступу до системи всім співробітникам заводу;
- аналізу введених даних та генерації на їх основі звітів;

#### 2. Цілі створення системи

Метою даної роботи є:

- Автоматизація всіх процесів, які виникають при ремонті;
- Автоматизація всіх процесів, які виникають при отриманні заявок;
- автоматизація процесів ,які виникають при розгляді заявок;
- пришвидшення процесу ремонту обладнання;
- зручне користування інтерфейсом;

- надання доступу до системи всім співробітникам заводу;
- аналізу введених введених даних та генерації на їх основі звітів.

## **2.3. Характеристика об'єкта автоматизації**

### 2.3.1 Короткі відомості про об'єкт автоматизації.

Об'єктом автоматизації є автоматизація ремонтного супроводу ТОВ «Рибний Комбінат Херсон».

Предметом діяльності ТОВ «Рибний Комбінат Херсон»:

- виготовлення, продаж та перевезення власної продукції ;
- виробництво, збереження, переробка і реалізація харчової продукції.
- торгівельна діяльність рибою;
- торгівельно-закупівельна та торгово-посередницька діяльність;
- створення оптово-роздрібних підприємств торгівлі
- доставка замовлень власної продукції;

Основними видами продукції є: риба, ракоподіні, моллюски . Загальний асортимент продукції становить близько 50 найменувань. Вся продукція виробляється у відповідності до державних стандартів, проходить перевірку за різними фізико-хімічними, мікробіологічними показниками та знаходиться під ветеринарним контролем.

## **2.4. Вимоги до системи**

### 4.1. Вимоги до системи в цілому.

#### 4.1.1 Вимоги до функціонування та структури системи.

4.1.1.1. Система повинна надавати можливість введення даних(тобто інформації про обладнання яке потребує ремонт).

4.1.1.2. Система повинна надавати користувачеві такі функції:

- авторизація користувача;
- отримання інформації про заявок, які були внесені різними співробітниками.

- додавання нових робітників;
- перевірка стану заявки;
- розставлення пріоритетності заявок.

4.1.1.3. Розвиток та покращення системи повинні проводитися шляхом уточнення заміни функцій, модернізація технічних та програмних засобів згідно із мірою розробки та впровадження сучасних поколінь комп'ютерів. Структура системи повинна забезпечити простоту модернізації та розвитку, задля можливості використовуванню їх на інших нових комп'ютерах.

4.1.1.4. Функціонування системи мусить забезпечувати діалогову та мережну обробку даних.

4.1.2. Вимоги до кваліфікації персоналу

4.1.2.1. Персонал який використовує нову систему, має знати та виконувати деякі вимоги:

- варто пройти навчання та отримати навички користування системою;
- дотримуватися певних інструкції при використанні;
- дотримуватися певних умов експлуатації ПК;
- дотримуватися правил користування ПК;
- дотримуватися правил техніки безпеки роботи за ПК;

4.1.2.2. Користувачем системи може виступати як головний механік відділу, співробітник з іншого відділу так і член якоїсь із бригади. Вхід у систему буде здійснюватися через пароль, який відображає рівень користувача: тобто в залежності від прав користувач бачить певну інформацію.

4.1.3. Показники призначення

4.1.3.1. Перелік та допустимі значення показників, при яких зберігається цільове призначення системи, повинні бути повинні бути визначені на стадії техноробочого проектування.

4.1.3.2. Система повинна мати можливість налаштування на параметри об'єкта управління та периферійного обладнання при її модернізації та розвитку, а також зміні процесів та методів організаційного управління .

4.1.4. Вимоги до надійності.

Система є багатофункціональною і призначена для використання протягом всього робочого дня. Створювана ІС повинна виключати можливість отримання прямого доступу будь-якого користувача.

У випадку виникнення непередбачених критичних ситуацій, аварій, відмов технічних засобів (у тому числі і зникнення напруги), збоїв у роботі загально-системного програмного забезпечення, збоїв у роботі, ІС повинна мати можливість відтворення своєї працездатності з резервних копій за короткий проміжок часу та з мінімальними втратами.

4.1.5. Вимоги до безпеки

Для забезпечення безпеки при експлуатації, налагодженні, монтажі, обслуговуванні і ремонті технічних засобів системи потрібно дотримуватись вимог ДСТУ: ДСТУ 2293-99, ДСТУ ISO 6309:2007, ДСТУ 12.0.230:2008, ДСТУ 7237:2011, ДСТУ 7238:2011, ДСТУ 7239:2011; по доступним рівням освітленості, вібраційних і шумових навантажень слід дотримуватися вимог відповідно ДСТУ Б А.3.2-15:2011, ДСТУ EN 14253:2018, ДСТУ 2867-94.

4.1.6. Вимоги з ергономіки та технічної естетики.

Загальні ергономічні і естетичні вимоги до системи повинні відповідати держстандартам ДСТУ 8604:2015, ДСТУ 7298:2013. Освітленість робочого місця повинна відповідати ДСТУ EN 12464-1:2016, ДБН В.2.5-28-2006.

Засоби відображення повинні розміщуватися таким чином, щоб кут спостереження екрану складав не більше, ніж 45 градусів, мінімальна відстань спостереження екрану — 0,3 м, рекомендована — 0,5 м.

При розробленні ПЗ слід створити зручний інтерфейс для запобігання втомлюваності користувача.

4.1.7. Вимоги по експлуатації, технічного обслуговування, ремонту і зберігання компонентів системи.

4.1.7.1. Види обслуговування системи визначаються у відповідності з ДСТУ EN 13306:2019. Загальні вимоги по експлуатації, технічному обслуговуванню і ремонту повинні відповідати ДСТУ 3576-97. 21

4.1.7.2. Кількість, кваліфікація і режим роботи обслуговуючого персоналу повинні відповідати рекомендаціям, зазначеним в інструкціях з експлуатації окремих ТЗ.

4.1.7.3. Склад, розміщення і умови зберігання компонентів технічних засобів системи визначається рекомендаціями, зазначеними в експлуатаційній документації на ці елементи.

4.1.7.4. Регламент обслуговування повинен відповідати їх рівню та умовам роботи, щоб у випадку відмови системи та забезпечити роботу в аварійному стані.

4.1.8. Вимоги до захисту інформації від несанкціонованого доступу.

Передача інформації між компонентами системи повинна виконуватися стандартними протоколами на рівні ПЗ або на рівні платформи.

4.1.9. Вимоги щодо збереження інформації при аваріях.

4.1.9.1. Варто передумовно передбачити автозбереження БД при аварійних ситуаціях.

4.1.9.2. Резервний архів та БД має знаходитись на різних технічних засобах.

4.1.10. Вимоги до захисту від впливу зовнішніх діянь.

4.1.10.1. Електрична складова електромагнітного поля завад в приміщеннях не повинна перевищувати  $0,3 \text{ В/м}^2$  в діапазоні частот від 0.15 до 300 МГц. Для захисту від впливу електромагнітних полів та індустриальних завад слід передбачити екрани та фільтри.

4.1.10.2. Засоби ,які виключають вплив шкідливих факторів на функціонування комплексу технічних засобів, повинні бути запроектовані згідно з ДБН В.2.2-9-2009. Обчислювальні засоби по стійкості до зовнішніх впливів повинні відповідати ДСТУ 2506-94.

#### 4.1.11. Вимоги до патентної чистоти.

При створенні системі для даної сфера подібних досліджень немає.

#### 4.2. Вимоги до функцій.

##### 4.2.1. Перелік функцій із зазначенням інформації наведено в таблиці 2.1.

Таблиця 2.1. Основні функції системи

Функції	Опис функцій
Автоматизація ремонтного супроводу	- автоматизації всіх ремонтних процесів , які виникають при роботі на заводі;  -пришвидшення процесу заміни обладнання;
Забезпечення безпеки	-забезпечення захищеного  -зберігання введених даних;  -надання доступу до системи іншим співробітникам
Перегляд інформації	-перегляд всього обладнання  -перегляд заявок  -перегляд інформації заявок
Права Адміна	- додавання робітників  - оброблення заявок

Спрощення введення даних	Зрозумілий для користувача веб інтерфейс дозволяє без труднощів зрозуміти новому користувачеві що і де вводити.
--------------------------	---

#### 4.3. Вимоги до видів забезпечення.

4.3.1. У вимогах до математичного забезпечення (МЗ) система не вимагає спеціального математичного забезпечення для реалізації покладених на неї функцій. Достатньо можливостей обраної СУБД.

#### 4.3.2. Вимоги до інформаційного забезпечення (ІЗ).

4.3.2.1. Дані ІС повинні знаходитись у структурованому вигляді. Структура даних повинна зберігати інформацію для точної ідентифікації її в ІС. Організація даних повинна забезпечувати гарну швидкість роботи з даною інформацією.

4.3.2.2. Слід передбачити захист даних від руйнування при аваріях і порушеннях у енергоживленні системи-використання резервних копій БД.

#### 4.3.3. Вимоги до лінгвістичного забезпечення (ЛЗ).

4.3.3.1. Для розробки програмних засобів, які реалізують виконання функцій і забезпечують сервіс користувачів повинні використовуватись мови високого рівня.

4.3.3.2. Інтерфейс користувача ІС повинен бути виконаний українською мовою. Допускається при виконанні регламентних процедур використання англійської або російської мови, що повинно бути затверджено у документації.

#### 4.3.4. Вимоги до програмного забезпечення(ПЗ).

4.3.4.1. Загальносистемне ПЗ має давати надійне та якісне виконання всіх можливостей системи. До загальносистемного ПЗ належить:

- операційна система (далі ОС) — Windows .....

- система управління БД (далі СУБД) — MySQL.

4.3.4.2. Загальні вимоги до системного ПЗ можна сформулювати так:

- мінімальні вимоги до ресурсів ТЗ;
- максимально швидкодія;
- повне задоволення потреб функціональних завдань.

Прикладне програмне забезпечення повинно базуватись на використанні вільно розповсюдженого програмного забезпечення, яке не потребує придбання ліцензій, у разі некомерційного використання.

4.3.4.3. Вимоги до ОС:

- мінімальне використання ресурсів комп'ютера для власних потреб співробітників;
- ОС сервера-Windows, ОС клієнта-Windows;
- Максимальна швидкодія при управлінні зовнішніми пристроями.

4.3.4.4. Вимоги до СУБД:

- задоволення всіх потреб співробітників;
- надійність;
- швидкість;
- комфортність;
- мінімізовані вимоги до ТЗ.

4.3.4.5. При розробленні спеціального ПЗ варто враховувати:

- ІС має бути в легкому доступі для кожного співробітника;
- ІС повинна функціонувати 24/7 без необхідності зупинки її роботи та перезапуску програми.
- ІС повинна мати можливість розгортання на декількох сервера для забезпечення надлишковості та мати функціонал балансування навантаження при розгортанні на декількох серверах.

4.3.5. Вимоги до технічного забезпечення.

Для функціонування ІС Набувач надає необхідні технічні засоби, які відповідають наступним вимогам(Табл 2)

Найменування	Основні характеристики
Процесор	2 x Intel Xeon E5-2670V3 чи аналогічний
Оперативна пам'ять	32GB DDR4-2400 ECC Registered
Мережеві адаптери	1Gbit Ethernet
Жорсткі диски	Enterprise 7200 RPM 8Tb SATA/SAS з MTBF не менш 2000000 годин
Монітор	15''
Миша та клавіатура	USB

Таблиця 2.2. Вимоги до технічного забезпечення системи

4.3.7. Вимоги до організаційного забезпечення.

4.3.7.1. Організаційне забезпечення системи розробляється відповідності з вимогами стандартів.

4.3.7.2. При впровадженні системи передбачаються збільшення штату. Територіальне розміщення робочих місць, на яких встановлена система може змінюватись, та визначається підприємством.

4.3.7.3. До функціонування системи висуваються вимоги:

- За наказом директора, попередньо погоджено з головним відділу, визначається список співробітників, які мають доступ до системи;
- Контроль та прийняття рішень при екстрених ситуаціях здійснює головний на сміні.

## **2.5. Склад та зміст по створенню системи.**

5.1. Стадії створення системи і терміни виконання робіт наведені в Таблиці 2.3.

<b>№ п/п</b>	<b>Найменування робіт</b>	<b>Строки виконання робіт</b>
1	Передпроектне дослідження об'єкта автоматизації	11.01.2021
2	Технічне завдання	08.05.2021
3	Технічний проект	19.05.2021
4	Оформлення документації	03.06.2021

Таблиця 2.3 Найменування робіт при створенні системи

## **2.6. Порядок контролю і приймання системи.**

6.1. Система вводиться на діючому ДП СТХТ НУХТ. При введенні в дію система повинна пройти початкові випробування згідно с ДСТУ 3974-2000.

6.2. Випробування системи для працездатності та рішення про можливість приймання системи проводиться замовником разом з розробником системи. Замовник затверджує систему, а розробник складає програму випробувань.

6.3. Здача в дослідну експлуатацію здійснюється на основі технічного завдання та також інструкції для користувачів. Після випробувань формується перелік доробок, які потрібно виправити або переробити та бажані строки їх реалізації.

6.4. Введення системи в роботи для відділу здійснюється оформленням актом здачі-прийому.

## **2.7. Вимоги до складу і змісту робіт із підготовки до введення системи в дію.**

Для того, щоб ввести в дію систему, замовник виконує певний ряд дій для підготовки об'єкта:

- проводить укомплектування всіх потрібних технічних засобів;
- організовує навчання співробітників, для коректного використання системи;
- проводить дослідну експлуатацію та вводить систему в роботу.

## **2.8. Вимоги до документації.**

8.1. На систему розробляється комплект документів таких, як: технічне завдання та технічний проект.

8.2. Документація на систему робиться у відповідності з вимогами Державних стандартів серії 19 «Єдина система програмної документації», та серії 24 «Єдина система стандартів автоматизованих систем управління».

## **2.9. Джерела розробки**

9.1. При розробці технічного завдання на систему використано такі документи, як:

- ДСТУ 3008-2015. Документація. Звіти у сфері науки та техніки. Структура та правила оформлювання;
- ДСТУ 3973-2000 Система розроблення та поставлення продукції на виробництво;
- ДСТУ Б В. 2. 3-82.2017 Електробезпека в будівлях та спорудах. Вимоги до захисних кодів від ураження електричним струмом.

## РОЗДІЛ 3. ОПИС КОМПЛЕКСУ ЗАДАЧ АВТОМАТИЗАЦІЇ

### 3.1. Використані програмне за забезпечення та технології

#### 3.1.1. Використані програмні засоби

Для виконання дипломної роботи та розробки системи були використані наступні програмні середовища та засоби:

- **AllFusion Erwin Process Modeler-** CASE засіб для функціонального моделювання різних процесів використовуючи при цьому нотації IDEF0, IDEF3, DFD.
- **ALLFusion Erwin Data Modeler-** CASE засіб котрий надає можливість робити дослідження предметної області з використанням об'єктно-орієнтованого підходу ті проектування баз та сховищ для різних БД.
- **MS SQL SERVER 2008-** СУБД, котра виступає як в ролі сервера для різних баз даних та також має можливість бути встановленою на різних видах комп'ютерів.
- **PHP-** як широко використовувана мова сценаріїв загального призначення з відкритим вихідним кодом.
- **CMS (Content Management System)** - це програмне забезпечення, що дозволяє управляти контентом сайту: створювати і публікувати записи, розміщувати віджети, змінювати дизайн, редагувати різні елементи, додавати функціонал і т. Д. Систему управління контентом сайту також називають двигуном.

### 3.2. Мова PHP та CMS система

**PHP** - це широко використовувана мова сценаріїв загального призначення з відкритим вихідним кодом. Мова PHP спеціально розроблений для написання web-додатків (сценаріїв), що виконуються на Web-сервері.

Перевагою PHP є надання web-розробникам можливості швидкого створення динамічно генеруються web-сторінок.

Головним фактором мови PHP є практичність. PHP має надати програмісту кошти для швидкого і ефективного вирішення поставлених завдань. Практичний характер PHP обумовлений п'ятьма важливими характеристиками:

- традиційністю;
- простотою;
- ефективністю;
- безпекою;
- гнучкістю.

Існує ще одна «характеристика», яка робить PHP особливо привабливим: він поширюється безкоштовно! Причому, з відкритими початковими кодами (Open Source).

### **Традиційність**

Мова PHP здаватиметься знайомим програмістам, що працюють в різних областях. Багато конструкцій мови запозичені з Cі, Perl.

Код PHP дуже схожий на той, який зустрічається в типових програмах на C або Pascal. Це помітно знижує початкові зусилля при вивченні PHP. PHP - мова, що поєднує переваги Perl і Cі і спеціально націлений на роботу в Інтернеті, мова з універсальним (правда, за деякими застереженнями) і ясным синтаксисом.

І хоча PHP є досить молодого мовою, він знайшов таку популярність серед web-програмістів, що на даний момент є мало не найпопулярнішою мовою для створення web-додатків (скриптів).

### **Простота**

Сценарій PHP може складатися з 10 000 рядків або з одного рядка - все залежить від специфіки вашого завдання. Вам не доведеться довантажувати

бібліотеки, вказувати спеціальні параметри компіляції або що-небудь в цьому роді. Механізм PHP просто починає виконувати код після першої екрануючої послідовності (<?) і продовжує виконання до того моменту, коли він зустрине парну екранує послідовність (?>). Якщо код має правильний синтаксис, він виконується в точності так, як вказав програміст.

PHP - мова, яка може бути вбудований безпосередньо в html-код сторінок, які, в свою чергу будуть коректно оброблятися PHP -інтерпретатором. Ми можемо використовувати PHP для написання CGI-сценаріїв і позбутися від безлічі незручних операторів виведення тексту. Ми можемо залучати PHP для формування HTML-документів, позбувшись від безлічі викликів зовнішніх сценаріїв.

Велика розмаїтість функцій PHP позбавлять вас від написання багаторядкових призначених для користувача функцій на C або Pascal.

### **Ефективність**

Ефективність є виключно важливим фактором при програмуванні для багатокористувацьких середовищ, до числа яких належить і web.

Дуже важлива перевага PHP полягає в його «движку». «Движок» PHP не є ні компілятором, ні інтерпретатором. Він є транслює інтерпретатором. Такий пристрій «движка» PHP дозволяє обробляти сценарії з достатньо високою швидкістю.

За деякими оцінками, більшість PHP-сценаріїв (особливо не дуже великих розмірів) обробляються швидше аналогічних їм програм, написаних на Perl. Однак, щоб не робили розробники PHP, відкомпілювалися виконувані файли будуть працювати значно швидше - в десятки, а іноді і в сотні разів. Але продуктивність PHP цілком достатня для створення цілком серйозних web-

додатків. Детально про пристрій і характеристиках «движка» РНР можна ознайомитися тут.

## **Безпека**

РНР надає в розпорядження розробників і адміністраторів гнучкі і ефективні засоби безпеки, які умовно поділяються на дві категорії: засоби системного рівня і засоби рівня додатки.

- **Засоби безпеки системного рівня**

У РНР реалізовані механізми безпеки, що знаходяться під управлінням адміністраторів; при правильному налаштуванні РНР це забезпечує максимальну свободу дій і безпеку. РНР може працювати в так званому безпечному режимі (safe mode), який обмежує можливості застосування РНР користувачами по ряду важливих показників. Наприклад, можна обмежити максимальний час виконання і використання пам'яті (неконтрольований витрата пам'яті негативно впливає на швидкодію сервера). За аналогією з cgi-bin адміністратор також може встановлювати обмеження на каталоги, в яких користувач може переглядати і виконувати сценарії РНР, а також використовувати сценарії РНР для перегляду конфіденційної інформації на сервері (наприклад, файлу passwd).

- **Кошти безпеки рівня додатки**

У стандартний набір функцій РНР входить ряд надійних механізмів шифрування. РНР також сумісний з багатьма додатками незалежних фірм, що дозволяє легко інтегрувати його з захищеними технологіями електронної комерції (e-commerce). Інша перевага полягає в тому, що вихідний текст сценаріїв РНР можна переглянути в браузері, оскільки сценарій компілюється до його відправки за запитом користувача. Реалізація РНР на стороні сервера запобігає викрадення нетривіальних

сценаріїв користувачами, знань яких вистачає хоча б для виконання команди View Source.

## **Гнучкість**

Оскільки PHP є вбудовуваним (embedded) мовою, він відрізняється винятковою гнучкістю по відношенню до потреб розробника. Хоча PHP зазвичай рекомендується використовувати в поєднанні з HTML, він з таким же успіхом інтегрується і в JavaScript, WML, XML та інші мови. Крім того, добре структуровані додатки PHP легко розширюються в міру необхідності (втім, це відноситься до всіх основних мов програмування).

Немає проблем і з залежністю від браузерів, оскільки перед відправкою клієнту сценарії PHP повністю компілюються на стороні сервера. По суті, сценарії PHP можуть передаватися будь-яких пристроїв з браузерами, включаючи стільникові телефони, електронні записники, пейджери і портативні комп'ютери, не кажучи вже про традиційні ПК. Програмісти, що займаються допоміжними утилітами, можуть запускати PHP в режимі командного рядка.

Оскільки PHP не містить коду, орієнтованого на конкретний web-сервер, користувачі не обмежуються певними серверами (можливо, незнайомими для них). Apache, Microsoft IIS, Netscape Enterprise Server, Stronghold і Zeus - PHP працює на всіх перерахованих серверах. Оскільки ці сервери працюють на різних платформах, PHP в цілому є переносним незалежним мовою і існує на таких платформах, як UNIX, Solaris, FreeBSD і Windows 95/98 / NT / 2000 / XP / 2003.

Нарешті, засоби PHP дозволяють програмісту працювати з зовнішніми компонентами, такими як Enterprise Java Beans або COM-об'єкти Win32. Завдяки цим новим можливостям PHP займає гідне місце серед сучасних технологій і забезпечує масштабування проектів до необхідних меж.

**CMS** - це ПЗ на базі скриптів, яке дозволяє керувати вмістом ресурсу, змінювати його, переглядати і контролювати. Сьогоднішні системи мають широку функціональність і складаються з величезної кількості модулів, кожен з яких відповідає за свої елементи. Програмне забезпечення допомагає складати типові сайти з блоків подібно до конструктора. Для цього практично не потрібно навіть навичок програмування. Умовно CMS можна розділити на кілька частин:

- сховище баз даних, де знаходиться інформація про користувачів, наповненні сайту і інших важливих сутності;
- сховище елементів інтерфейсу, з якими безпосередньо взаємодіє користувач при перегляді сайту;
- візуальний редактор, що допомагає з легкістю створювати сторінки.

Крім того, множинні модулі дозволяють додати до сайту ті або інші додаткові функції.

CMS дозволяє:

- наповнювати сайт контентом, змінювати і адмініструвати ресурс, при цьому не будучи ІТ-фахівцем і не маючи серйозних навичок програмування;
- створювати нові сторінки в короткі терміни без зайвих витрат;
- оптимізувати зовнішній вигляд сайту і покращувати якість його наповнення.

Плюси CMS системи:

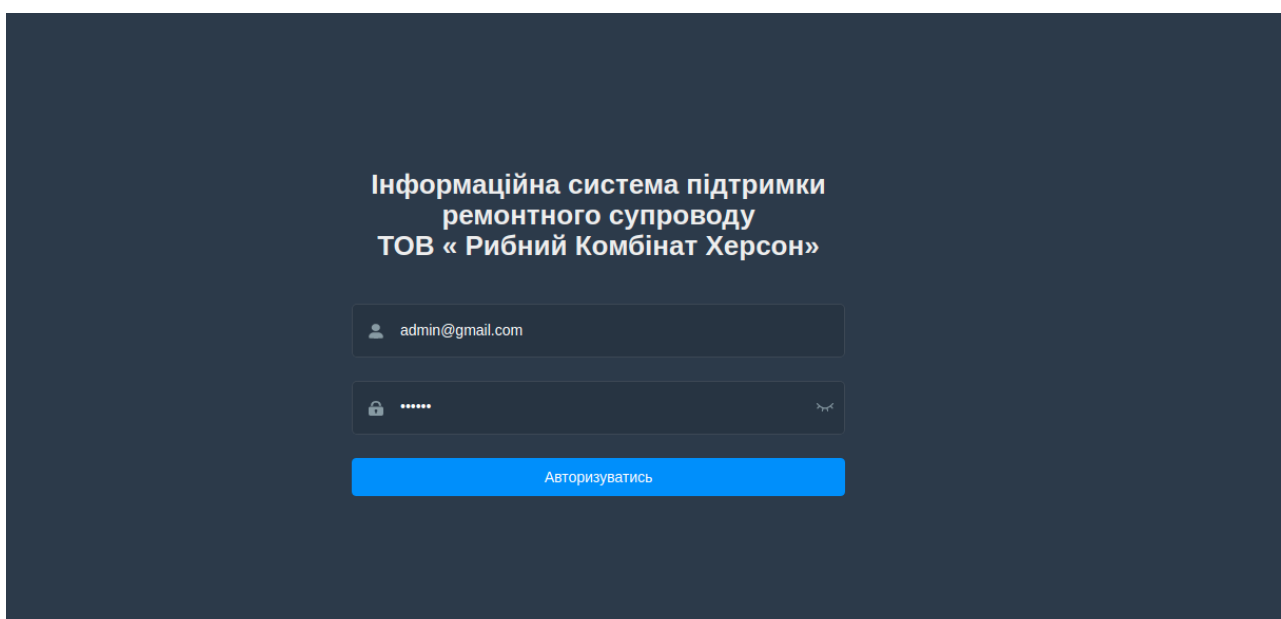
- відсутність необхідності в серйозних технічних знаннях, просунутих навичках програмування і верстки;
- висока швидкість створення сайту;
- зручність адміністрування та додавання нових елементів;
- можливість без зайвих складнощів створити гарний дизайн;
- просте наповнення, доступне навіть людям без спеціальних знань.

Мінуси CMS системи:

- Єдиним недоліком можна назвати складність створення унікального сайту з нестандартними функціями, але такі ресурси, як правило, потрібні компаніям з відповідними запитами (наприклад, працюючим в ІТ-сфері).

### 3.3. Інтерфейс користувача

Заходячи на веб-сторінку відділу одразу перед нами відкривається вікно авторизації(Рис3.3.1.)



(Рис3.3.1) Сторінка авторизація користувача

На мою думку сайт для відділу ремонтного супроводу має бути простий та зрозумілий, адже на заводі можуть працювати користувачі різного віку та з різним досвідом роботи с ПК або телефоном.

Далі в залежності від того хто авторизується відкривається наступне вікно. Якщо наприклад авторизувався головний механік заводу, який повинен бачити всі заявки, стан всіх заявок ,пріоритетність, дату отримання заявок ,автора цієї заявки. Переде головним механіком відділу відкривається головне меню,яке ми можемо бачити на Рис.3.3.2 .Де видно загальна кількість бригад, робочих,які

заявки зараз в обробці, а які в роботі, звіти по закупівлі пристроїв та завти по списанню.

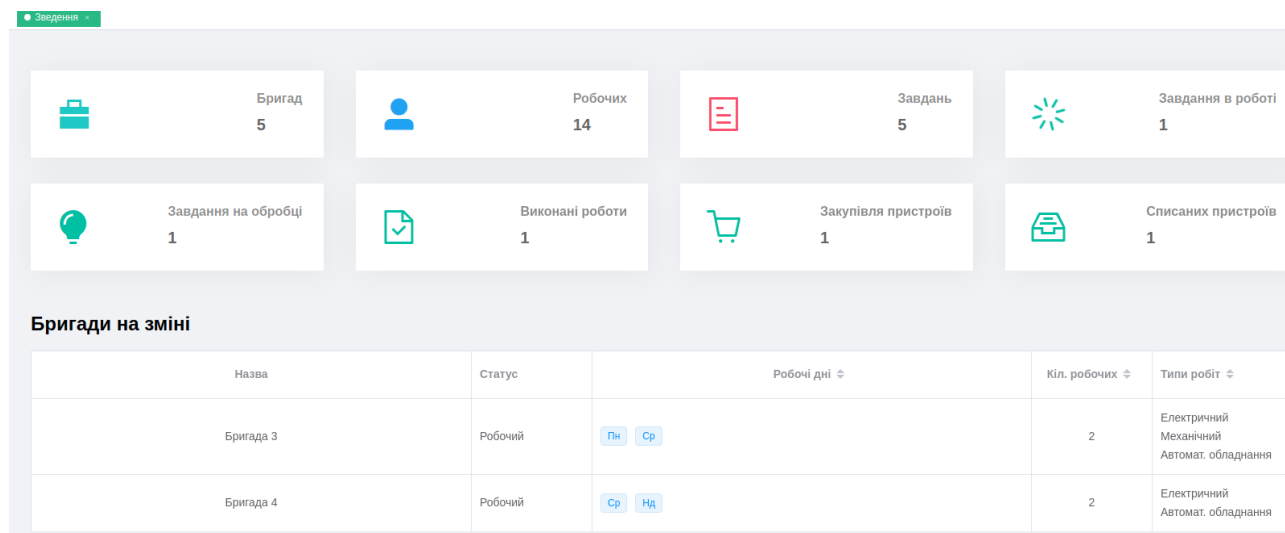


Рис3.3.2. Адмінська частина

Також зліва на сторінці механік може подивитись окремо статуси заявок, окремо бригад, працівників цеху на перейти на сторінку звітів. На Рис 3.3.3, ми можемо бачити саме сторінку для головного механіка, а зокрема перегляд всіх заявок, їх статусів, робітників та тд.

Зведення × Список завдань ×

Назва Тип Пріоритет Статус Виконавець Пристрій 🔍 +

ID	Назва	Статус	Тип	Пріоритет	Пристрій	Виконавець	Автор	Дата ств.	Дата оновл.	Дії
> 1	Ремонт варильного котла	У обробці	Електричний	Високий	Варильний ко...	Не призначений	Панасюк Руслан	21 травня, 2021 р	21 травня, 2021 р	👁️ 🗑️
> 2	Ремонт холодильної системи	В роботі	Електричний	Високий	Холодильна у...	Панасюк Руслан	Кравченко Георгій	21 травня, 2021 р	21 травня, 2021 р	👁️ 🗑️
> 3	Ремонт сортувальної машини	Виконано	Автомат. обладнання	Середній	Сортувальна ...	Василенко Григ...	Мельниченко Я...	21 травня, 2021 р	21 травня, 2021 р	👁️ 🗑️
> 4	Ремонт системи витяжки	Виконано	Механічний	Середній	Система витя...	Пономарчук Іван	Крамаренко Ст...	21 травня, 2021 р	21 травня, 2021 р	👁️ 🗑️
> 5	Метрологічне забезпечення	Відкладено	Електричний	Низький	Терморегулят...	Не призначений	Кравченко Георгій	21 травня, 2021 р	21 травня, 2021 р	👁️ 🗑️

Всього 5 5на сторінці < 1 > Перейти 1

Рис.3.3.3.(1) Сторінка для головного механіка.



Рис.3.3.4.



Рис.3.3.5

На сторінці присутні такі кнопки, як зображені на Рис 3.3.4, перша для пошуку заявки по фільтрам, друга додавання нової заявки головним механіком заводу. На Рис.3.3.5. зображені дві кнопки для певних дій з заявкою, наприклад: видалити або перегляд в якому також можна редагувати заявку.

Якщо ж механік вибирає функцію додавання нової заявки перед ним відкривається вікно для оформлення заявки. Рис 3.3.6.

Створити завдання

Назва

Назва

Тип

Пріоритет

Вибрати Тип

Вибрати пріоритет

Виконавець

Пристрій

Вибрати виконавця

Вибрати пристрій

Опис

Закрити

Зберегти

Рис.3.3.6. Створення нової заявки

Інший варіант це перегляд вже існуючої заявки, де можлива заміна працівника або інший будь-який критерій. Рис 3.3.7.

Перегляд завдання

Назва

Тип

Пріоритет

Виконавець

Пристрій

Опис

Рис.3.3.7. Перегляд вже існуючої заявки

Можливі фільтри на головній сторінці:

Тип

- Електричний
- Механічний
- Автомат. обладнання

Рис.3.3.8. Фільтр за типом роботи в заявці

Пріоритет

- Низький
- Високий
- Середній

Рис.3.3.9. Фільтр за пріоритетністю

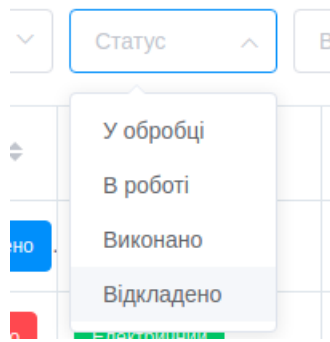


Рис.3.3.10. Фільтр за статусом заявки

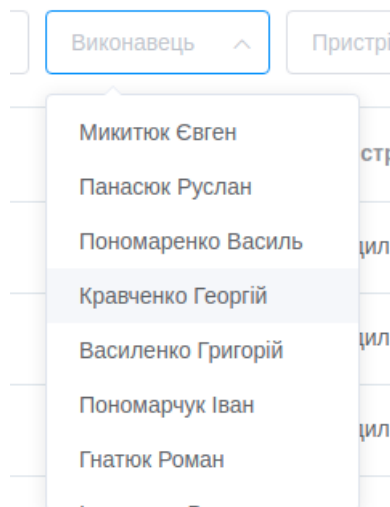


Рис.3.3.11. Фільтр за виконавцем

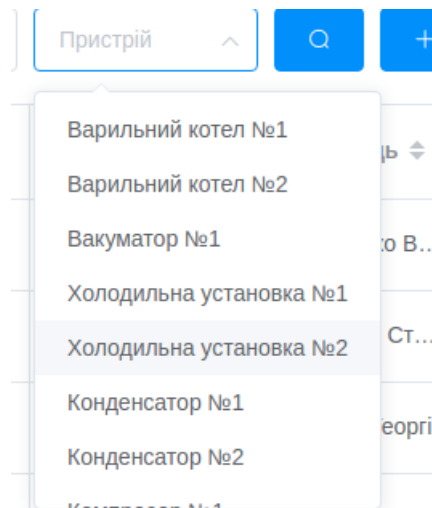


Рис.3.3.12. Фільтр за пристроєм для якого виконаний ремонт

Доступна ще можливість для головного механіка така, як перегляд всіх заявок, та перегляд історії конкретної вибраної заявки. Рис.3.3.13.

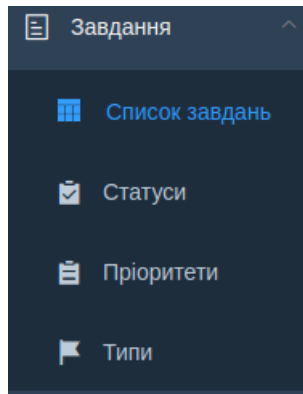


Рис.3.3.13. Кнопка для перегляду заявок

ID	Назва	Статус	Тип	Пріоритет	Пристрій	Виконавець	Автор	Дата ств.	Дата оновл.
1	Ремонт варильного котла	Виконано	Механічний	Високий	Терморегулят...	Гнатюк Роман	Іванченко Вале...	21 травня, 2021 р	21 травня, 2021 р

Id	Статус	Опис	Дата ств.
3	Виконано	Виконав всі роботи по цьому завданню	21 травня, 2021 р 18:00
2	В роботі		21 травня, 2021 р 15:00
1	У обробці		21 травня, 2021 р 13:00

Рис.3.3.13.(1)Вигляд сторінки для перегляду історії заявки.

На сторінці для головного також присутні сторінки статуси, перейшовши на неї можна спостерігати в якому статусі знаходяться всі заявки. Рис.3.3.14 та Рис.3.3.14(1).

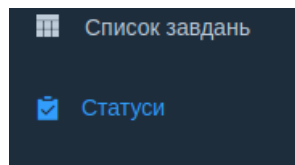


Рис.3.3.14. Сторінка перегляду статусів заявок

ID	Назва	Кільк. завдань	Дата створення	Дата оновлення	
1	У обробці	1	23 травня, 2021 р	23 травня, 2021 р	+ Додати
2	В роботі	1	23 травня, 2021 р	23 травня, 2021 р	🔍 🗑️
3	Виконано	2	23 травня, 2021 р	23 травня, 2021 р	🔍 🗑️
4	Відкладено	1	23 травня, 2021 р	23 травня, 2021 р	🔍 🗑️

Рис.3.3.14(1).Перегляд статусів заявок

Наступний етап це перегляд наших бригад та фронт їх работ. Також можливо додати нову бригаду з новими учасниками. Для цього присутні кнопки, які зображені на Рис.3.3.15, та приклад вигляду екрану для перегляду бригад на де ми можемо бачити бригаду , їх статус, робочі дні та спеціалізацію

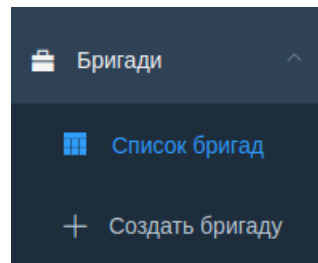


Рис.3.3.15. Кнопки для перегляду бригад

ID	Назва	Статус	Робочі дні	Кіл. робочих	Типи робіт	Дата ств.	Дата оновл.	Дії
1	Бригада 1	Вихідний	Пн Вт	2	Електричний Механічний Автомат. обладнання	23 травня, 2021 р	23 травня, 2021 р	
2	Бригада 2	Вихідний	Пн Вт	5	Електричний Механічний Автомат. обладнання	23 травня, 2021 р	23 травня, 2021 р	
3	Бригада 3	Робочий	Пн Ср	2	Електричний Механічний Автомат. обладнання	23 травня, 2021 р	23 травня, 2021 р	
4	Бригада 4	Робочий	Ср Чд	2	Електричний Автомат. обладнання	23 травня, 2021 р	23 травня, 2021 р	
5	Бригада 5	Вихідний	Чт Пт Сб	3	Електричний Механічний Автомат. обладнання	23 травня, 2021 р	23 травня, 2021 р	

Рис.3.3.15(1) Перегляд бригад, тип робіт та к-ть робочих.

Реалізація запиту для перегляду бригад реалізовано в SQL-запит наступним чином:

```
select brigades.name,
       bs.name          as `status`,
       GROUP_CONCAT(CONCAT(w.first_name, ' ', w.last_name)
                    SEPARATOR ',') as `workman`,
       GROUP_CONCAT(DISTINCT it.name SEPARATOR ',') as
`types`,
       brigades.work_days,
       COUNT(w.id)      as `count_workman`,
```

```

DATE_FORMAT(brigades.created_at, "%d %M, %Y")           as
`created_at`,
DATE_FORMAT(brigades.updated_at, "%d %M, %Y")          as
`update_at`
from brigades
left join workman w on brigades.id = w.brigade_id
left join workman_type wt on w.id = wt.workman_id
left join issue_types it on wt.type_id = it.id
left join brigade_statuses bs on brigades.status_id = bs.id
group by brigades.id

```

Можливий так названий мульти-фільтр для видів роботи між бригадами, тобто наприклад: знайти бригаду яка одночасно виконує механічний та електричний тип робіт. Рис.3.3.16.

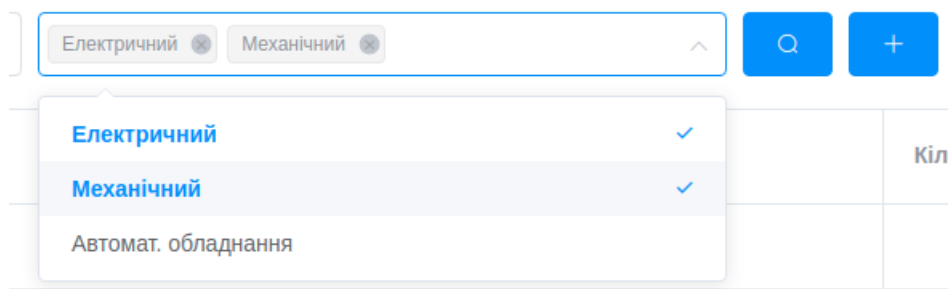


Рис.3.3.16.Мульти-фільтр.

Можливий перегляд кожної з цих бригад, та тільки для головного механіка можливо змінити учасника бригади, або ж робочі дня. Також головний має можливість додати нову бригаду. Рис.3.3.17 та Рис3.3.17.(1)

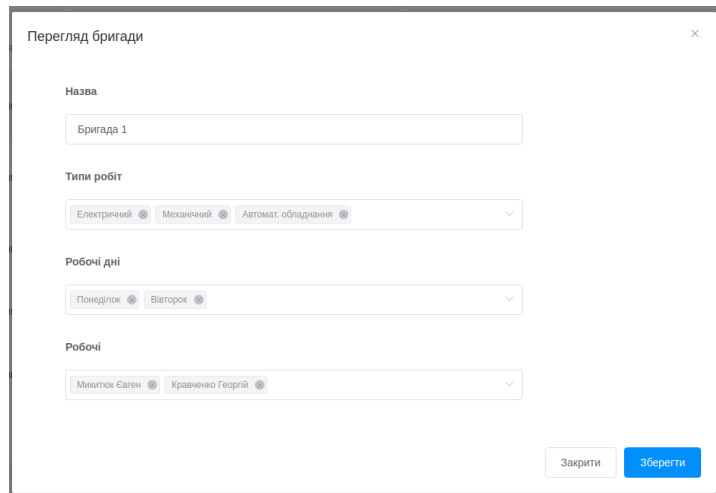


Рис 3.3.17. Перегляд бригади

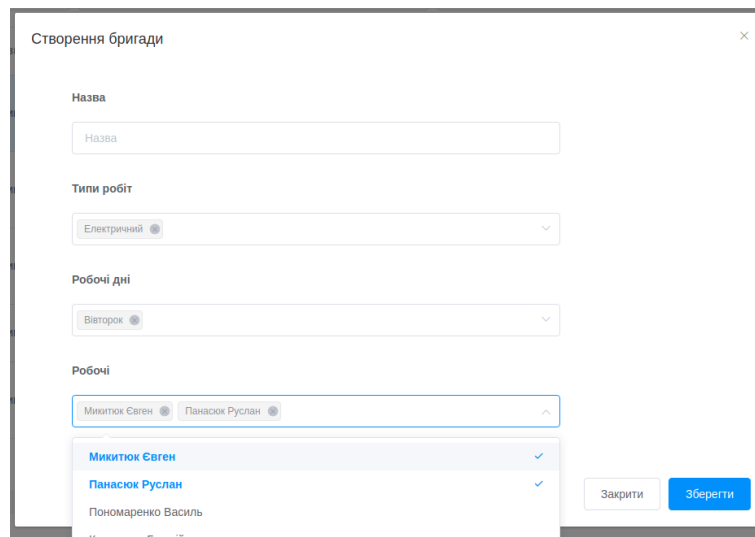


Рис.3.3.17.(1).Створення нової бригади.

Сторінка робочих для перегляду та додавання нових працівників.

Рис.3.3.18 та Рис.3.3.18(1).

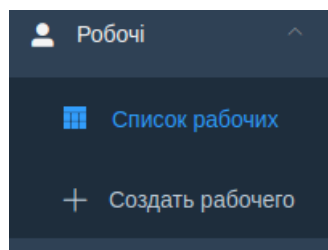


Рис.3.3.18. Кнопка робочих











ID	ПІБ	Email	Телефон	Типи робіт	Дата ств.	Дата оновл.	Дії
1	Микитюк Євген	test@gmail.com	+380630738041	Електричний Механічний	23 травня, 2021 р	23 травня, 2021 р	 
2	Панасюк Руслан	test1@gmail.com	+380630738042	Електричний	23 травня, 2021 р	23 травня, 2021 р	 
3	Пономаренко Василь	test2@gmail.com	+380630738045	Електричний Механічний	23 травня, 2021 р	23 травня, 2021 р	 
4	Кравченко Георгій	test3@gmail.com	+380630738044	Автомат. обладнання	23 травня, 2021 р	23 травня, 2021 р	 
5	Василенко Григорій	test4@gmail.com	+380630738046	Механічний Автомат. обладнання	23 травня, 2021 р	23 травня, 2021 р	 

Рис.3.3.18(1).Список працівників, дані про них.

Можливий доступ до змін інформації про робітників тільки для головного механіка. Вигляд вікна додавання нового робітника та редагування вже існуючого можна побачити на Рис.3.3.19 та Рис3.3.19(1).

Також подібний вигляд має сторінка перегляду робітників заводу, де інформація тільки про користувача, тобто пошта контактний номер, та відділ в якому працює.

Перегляд робочого ×

ПІБ

Email

Пароль

Телефон

Типи робіт

Рис3.3.19 Перегляд інформації.

Створення робочого ×

**ПІБ**

Назва

**Email**

Email

**Пароль**

Пароль

**Телефон**

Телефон

**Типи робіт**

Електричний

Електричний

Механічний

Автомат. обладнання

Рис.3.3.19(1).Додавання нового

Існує другий варіант авторизації, якщо заходить людина яка працює у іншому відділі та бажає залишити заявку для відділу ремонту, вони бачить список вже залишених ним заявок та їх статус. Також є можливість додати нову заявку, при цьому він не може вибрати виконавця заявки! Рис.3.3.20. та Рис.3.3.20(1)

ID	Назва	Статус	Тип	Пріоритет	Пристрій	Виконавець	Дата ств.	Дата оновл.
1	Ремонт варильного котла	У обробці	Електричний	Високий	Варильний ко...	Не призначений	21 травня, 2021 р	21 травня, 2021 р
2	Ремонт холодильної системи	В роботі	Електричний	Високий	Холодильна у...	Панасюк Руслан	21 травня, 2021 р	21 травня, 2021 р

Рис.3.3.20. Перегляд заявок.

створення завдання ×

**Назва**

Назва

**Тип**

**Пріоритет**

**Пристрій**

**Опис**

Рис.3.3.20(1).Додавання нової заявки без вибору виконавця.

Реалізація запиту для перегляду заявок та їх статусу наданий моєнт реалізовано в SQL-запит наступним чином:

```

select issues.id,
       issues.name,
       it.name           as `type_name`,
       ip.name          as `priority_name`,
       d.name           as `device_name`,
       (SELECT i.name
        from issue_history
         left join issue_statuses i on i.id = issue_history.status_id
        where issue_history.issue_id = issues.id
        order by issue_history.created_at
        LIMIT 1)        as `status`,
       CONCAT(u.first_name, ' ', u.last_name) as `user_name`,
       CONCAT(w.first_name, ' ', w.last_name) as `workman`
from issues
 left join workman w on issues.workman_id = w.id
 left join users u on issues.user_id = u.id
 left join issue_types it on it.id = issues.type_id
 left join issue_priorities ip on issues.priority_id = ip.id
 left join devices d on issues.device_id = d.id

```

При авторизації учасником бригади, він може бачити заявки які закріплені за ним, її пріоритетність та стан заявки який він може самостійно змінювати. Рис.3.3.21 та Рис.3.3.21(1).

ID ↕	Назва ↕	Статус ↕	Тип ↕	Пріоритет ↕	Пристрій ↕	Автор ↕	Дата ств. ↕	Дата оновл. ↕	Дії
1	Ремонт варильного котла	У обробці	Електричний	Високий	Варильний ко...	Панасюк Руслан	21 травня, 2021 р	21 травня, 2021 р	
2	Ремонт холодильної системи	В роботі	Електричний	Високий	Холодильна у...	Кравченко Георгій	21 травня, 2021 р	21 травня, 2021 р	

Рис.3.3.21. Вигляд заявок для конкретного учасника бригади.

Рис.3.3.21.(1).Зміна учасником бригади інформації про стан заявки.

На Рис.3.3.21(1) ми чітко можемо бачити різницю в можливостях для головного механіка та для члену бригади. Учасник бригади НЕ МАЄ МОЖЛИВОСТІ змінювати щось в заявці, окрім статусу, який потім відображається в історії нашої заявки, та надати додаткову інформацію про списання або закупівля нового обладнання.

Сторінка формування звітів є в доступі тільки для головного механіка. Звіти, які він може сформувавати за певний період: звіти про списане обладнання, звіти виконаних замовлень та звіт про закупівлю обладнання. Рис.3.3.22 ,Рис.3.3.22(1)та Рис.3.3.22(2)

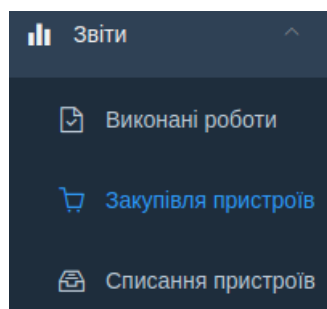


Рис.3.3.22. Перехід на сторінку звітів

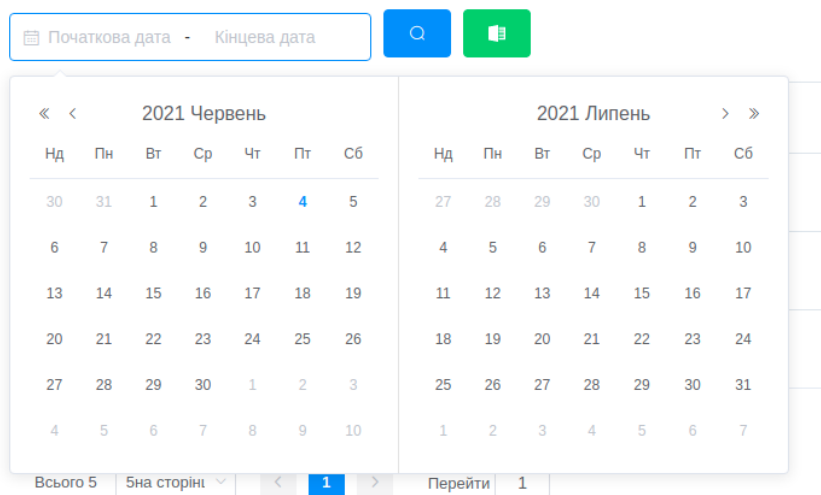


Рис.3.3.22(1)Вибір дати для формування звіту за період.

ID	Пристрій	Серійний номер	Автор
1	Сортувальна машина №1	GTd-ga-2644-32	Мельниченко Ярослав
2	Терморегулятор №1	QWS-232-fdf-309264432	Кравченко Георгій
3	Сортувальна машина №1	309-264-43-2ASD	Мельниченко Ярослав

Всього 5    Бна сторінці    < 1 >    Перейти 1

Рис.3.3.22(2)Сторінка сформованого звіту

### 3.4.Проектування БД

Для проектування бази даних в першу чергу була розроблена логічна модель БД в програмному засобі AllFusion Erwin Data Modeler.

Після чого на основі логічної моделі БД була створена фізична модель БД, яка на відміну від логічної вже проектується з розрахунком того, під яку СУБД проектується БД, в нашому випадку це MS SQL Server 2008 (Додаток Г). На наступному кроці фізична модель БД була згенерована в БД в СУБД MS SQL Server 2008.

Фізична модель була згенерована в БД в СУБД MS SQL Server 2008.Також була створена логічна та фізична схема БД в Erwin.(Додаток В).

### 3.4. Техніко-економічне обґрунтування

#### 3.4.1. Визначення розміру оплати праці

##### 4.1. Вид системи:

Інформаційна система ремонтного супроводу ТОВ «Рибний комбінат Херсон»

##### 4.2. Розрахунки проводяться для кожної стадії розробки системи.

Визначаються такі стадії:

- Ескізний проект (передпроектне дослідження);
- Технічне завдання;
- Технічний проект;
- Робочий проект;
- Впровадження.

##### 4.3. Ступінь новизни розроблюваних задач – В.

##### 4.4. Група складності алгоритму за їх характеристикою:

Алгоритми пріоритетності, звітності, статистики.

4.5. Визначається вид інформації, яка використовується, на основі аналізу вхідної та вихідної інформації функціональної моделі системи.

Таблиця 4. Визначення виду інформації.

Вид інформації	Позначення	Кількість наборів даних
Кількість видів змінної інформації	ЗІ	m=9
Кількість видів нормативно-довідкової інформації	НДІ	n=6
Кількість баз даних	БД	p=1
Обробка в режимі реального часу	РЧ	Так
Забезпечення телекомунікаційної обробки даних і управління віддаленими об'єктами	ТОУ	Ні

4.6. Визначаються витрати часу на розробку ескізного проекту (передпроектного дослідження) T1 і технічного завдання T2 за даними:

Ескізний проект,  $T_1 = 53$ .

Технічне завдання  $T_2 = 42$ .

4.7. Визначаються витрати часу на стадіях «технічний проект», «робочий проект» і «впровадження».

– **Визначення витрат часу для стадії “технічний проект” (T3).**

$$k_{\Pi} = \frac{k_1 * m + k_2 * n + k_3 * p}{m + n + p}$$

$$K_{\Pi} = (1 * 9 + 0.67 * 6 + 2.08 * 1) / (9 + 6 + 1) = 0.94735$$

$$T_3 = T_{B3} * k_{\Pi} * k_O$$

$$T_3 = 96 * 0.94735 * 1.26 = 114.156$$

– **Визначення витрат часу на стадії «робочий проект» (T4).**

$$k_{\Pi} = \frac{k_1 * m + k_2 * n + k_3 * p}{m + n + p}$$

$$K_{\Pi} = (1.1 * 9 + 0.24 * 6 + 0.48 * 1) / (9 + 6 + 1) = 0.73875$$

$$T_4 = T_{B4} * k_{\Pi} * k_O * k_C$$

$$T_4 = 176 * 0.73875 * 1.32 * 1 = 171.6264$$

– **Визначення витрат часу на стадії «впровадження» (T5).**

$$k_{\Pi} = \frac{k_1 * m + k_2 * n + k_3 * p}{m + n + p}$$

$$K_{\Pi} = (1.1 * 9 + 0.58 * 6 + 0.24 * 1) / (9 + 6 + 1) = 0.85125$$

$$T_5 = T_{B5} * k_{\Pi} * k_O * k_C$$

$$T_5 = 64 * 0.85125 * 1.21 * 1 = 65.9208$$

4.8. Для визначення загальних витрат часу на розробку системи використовується:

$$T_{\Sigma} = T_1 + T_2 + T_3 + T_4 + T_5 = 53 + 42 + 114.156 + 171.63 + 65.92 = 467.703 \text{ (годин)}$$

4.9. Визначення чисельності виконавців:

$$\text{Ч} = \frac{T_{\Sigma}}{\Phi}$$

$\Phi=90$  днів

$\text{Ч}=467.703/90\approx 5.196$  виконавців.

1.10. Оплата праці виконавців підраховується за формулою:

$$V'_1 = \text{Ч} * \text{М} * \text{ЗП}_{\text{ПР}} = 6 * 3 * 10\,000 = 180\,000 \text{ грн}$$

### 3.4.2 .Витрати, пов'язані з розробкою програми на ПК

1. Розрахунок річного фонду часу роботи ПК в годинах:

$$T_{\text{ПК}} = T_{\text{ОП}} - (6 * 8 + 5 * 12) = 2000 - (6*8+5*12) = 1892 \text{ години}$$

$$T'_{\text{ПК}} = T_{\text{ПК}} * \frac{R}{T_{\text{ОП}}} = 1892 * (440/2000) = 416 \text{ годин}$$

1.10. Оплата праці виконавців підраховується формулою:

$$V'_1 = \text{Ч} * \text{М} * \text{ЗП}_{\text{ПР}} = 6 * 3 * 12\,000 = 216\,000 \text{ грн}$$

2. Поточні витрати на експлуатацію  $V1''$ :

$$Ц_{\text{ПК}} = Ц_{\text{р}} * (1 + k_{\text{УН}})$$

$$Ц_{\text{ПК}}=15\,000 * (1+0.12)=16800 \text{ гривень.}$$

$$З_{\text{АМ}} = \frac{Ц_{\text{ПК}}}{N_{\text{А}}}$$

$$\text{Зам} = 16800/5=3\,360 \text{ гривень}$$

Витрати на електроенергію, споживану ПК, визначаються за наступною формулою, де потужність ПК,  $P_{\text{ПК}} = 0.4$  кВт, фонд корисного часу роботи ПК,  $T_{\text{ПК}} = 425.7$  год, вартість 1 кВт електроенергії для підприємств,  $Ц_{\text{ЕЛ}} = 2.01$  грн/кВт, коефіцієнт інтенсивного використання ПК,  $A = 0.9$ .

$$З_{\text{ЕЛ}} = P_{\text{ПК}} * T_{\text{ПК}} * Ц_{\text{ЕЛ}} * A = 643.65 \text{ грн}$$

ЗР – витрати на поточний ремонт і технічне обслуговування ПК визначаються як 6% від балансової вартості ПК, ЦПК.

$$З_p = Ц_{ПК} * 0.06$$

$$З_p = 16\,800 * 0.06 = 1008 \text{ грн}$$

ЗМАТ – непрямі витрати, пов'язані з експлуатацією, визначаються як 5% від балансової вартості ПК ЦПК.

$$З_{МАТ} = Ц_{ПК} * 0.05$$

$$З_{МАТ} = 16800 * 0.05 = 840 \text{ грн}$$

Отже, загальні витрати на розробку програмного забезпечення комп'ютерної системи розраховуються за формулою 5.3.13 і складуть:

$$V_1'' = З_{ОП} + З_{АМ} + З_{ЕЛ} + З_p + З_{МАТ}$$

$$V_1'' = 5000 + 3360 + 643.65 + 1008 + 840 = 10\,815.65$$

$$V_1 = V_1' + V_1''$$

$$V_1 = 216\,000 + 10\,815.65 = 226\,861.65 \text{ гривень.}$$

Так як є потреба в купівлі ПК і встановлені на підприємстві то витрати на придбання і установку ПК V2 становитимуть:

$$V_2 = 21280 \text{ грн}$$

4. Ці витрати залежать від стану приміщення, де буде встановлюватися ПК.

Так як пристосоване приміщення є, тому:

$$V_3 = 0 \text{ грн}$$

5. В середньому навчання персоналу триватиме 1 місяць, тому можна вважати, що:

$$V_4 = 3000 \text{ грн}$$

5. Загальна вартість розробки і впровадження системи вираховується за формулою:

$$V_{\Sigma} = V_1 + V_2 + V_3 + V_4$$

$$V = 226\,861.65 + 21\,280 + 0 + 3\,000 = 251\,141.65 \text{ гривень}$$

Оскільки норма амортизаційних втрат для комп'ютерних систем  $HA = 5$ , то для обрахування річного економічного ефекту слід брати до розгляду величину:

$$V_p = \frac{V_{\Sigma}}{H_A}$$

$$V_p = 246\,519.86 / 5 = 50\,228.33 \text{ гривень}$$

Таблиця 5. Основні джерела прибутку від впровадження комп'ютерної системи і порядок його підрахунку.

№	Джерела прибутку	Порядок підрахунку прибутку	Сума за рік, грн
1.	Можливість виконувати більшу кількість замовлень	За рахунок збільшення співробітників для ремонту кількості можливих заявок збільшується	23 000
2.	Збільшення якості продукції	За рахунок можливості прямого швидкого укладенням поламок працівниками, якість та швидкість виконаних замовлень на заводі збільшиться.	5200
Разом			28200

Коефіцієнт економічної ефективності розробки вираховується за:

$$K_{\text{ЕФ}} = \frac{\Pi_p}{V_p}$$

$$K_{\text{ЕФ}} = 81\,500 / 50\,228.33 = 1.62$$

Термін окупності розробки визначається за формулою:

$$T_{\text{ок}} = 1 / 1.62 = 0.61$$

Термін окупності системи буде становити 5 місяців.

## **РОЗДІЛ 4. ОХОРОНА ПРАЦІ**

### **4.1. ЗАГАЛЬНА ІНФОРМАЦІЯ ПО ОХОРОНУ ПРАЦІ**

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 в процесі трудової діяльності.

Основними завданнями в галузі охорони праці є підготовка, прийняття та реалізація заходів, які спрямовані на:

- забезпечення належних, безпечних і здорових умов праці;
- утримання в належному стані виробничого устаткування, будівель і споруд;
- пропаганду охорони праці;
- облік, аналіз та оцінку стану умов і безпеки праці;
- забезпечення страхування працівників від нещасного випадку на виробництві та профзахворювання.

### **4.2. Охорона праці користувачів ПК**

Головними елементами робочого місця того хто працює з ПК є письмовий стіл та крісло. Основними робочим положенням є положення сидячи. Робоче місце для виконання робіт у положенні організується відповідно до ГОСТ 12.2.032-78. Робоча поза сидячи викликає мінімальне стомлення. Раціональне планування робочого місця передбачає чіткий порядок і сталість розміщення предметів, засобів праці і документації. Те, що потрібно для виконання робіт частіше, розташоване в зоні легкої досяжності робочого простору.

Моторне поле – простір робочого місця, в якому можуть здійснюватися рухові дії людини. Максимальна зона досяжності рук – це частина моторного поля робочого місця, обмеженого дугами, описуваними максимально витягнутими руками при русі їх у плечовому суглобі.

Оптимальна зона - частина моторного поля робочого місця, обмеженого о дугами, описуваними передпліччями при русі в ліктьових суглобах з опорою в точці ліктя і з відносно нерухомим плечем. Зони досяжності рук у горизонтальній площині:

А- зона максимальної досяжності;

б - зона досяжності пальців при витягнутій руці;

в - зона легкої досяжності долоні;

г - оптимальний простір для грубої ручної роботи;

д - оптимальний простір для тонкої ручної роботи.

Розглянемо оптимальне розміщення предметів праці і документації в зонах досяжності рук:

ДІСПЛЕЙ розміщується в зоні а (у центрі);

КЛАВІАТУРА - у зоні г / д;

СИСТЕМНИЙ БЛОК розміщується в зоні б(зліва);

ПРИНТЕР знаходиться в зоні а (праворуч);

ДОКУМЕНТАЦІЯ в зоні легкої досяжності долоні-в(ліворуч) література і документація, необхідна при роботі; у висувних ящиках столу - література, невикористовувана постійно.

При проектування письмового столу слід врахувати наступне:

- висота столу повинна бути вибрана з урахуванням можливості сидіти вільно, в зручній позі, при необхідності спираючись на підлокітники;

- нижня частина столу повинна бути сконструйована так, щоб співробітник міг зручно сидіти, не був змушений підгинати ноги;

- конструкція столу повинна передбачати наявність висувних ящиків (не менше 3 для зберігання документації, канцелярії, особистих речей)..

Параметри робочого місця вибираються у відповідності з антропометричними характеристиками. При роботі в положенні сидячи рекомендуються такі параметри простору: ширина не менше 700 мм, глибина не менше 400 мм, висота робочої поверхні столу над підлогою 700-750 мм.

Оптимальними розмірами столу є:

висота 710 мм;

довжина столу 1300 мм;

ширина столу 650 мм.

Поверхня для листа повинна мати не менше 40 мм в глибину і не менше 600 мм завширки. Під робочою поверхнею повинно бути передбачено простір для ніг:

- висота не менше 600 мм,

- ширина не менше 500 мм,

- глибина не менше 400 мм.

Важливим елементом робочого місця за ПК є крісло. Воно виконується відповідно до ГОСТ 21.889-76. При проектуванні крісла виходять з того, що при будь-якому робочому положенні програміста його поза повинна бути фізіологічно правильно обгрунтованою, тобто положення частин тіла повинно бути оптимальним.

Для задоволення вимог фізіології, що впливають з аналізу положення тіла людини в положенні сидячи, конструкція робочого сидіння повинна відповідати таким основним вимогам:

- ✓ припускати можливість зміни положення тіла, тобто забезпечувати вільне переміщення корпусу і кінцівок тіла один щодо одного;
- ✓ допускати регулювання висоти в залежності від зростання працюючої людини(у межах від 400 до 550мм),мати злегка увігнуту поверхню, мати невеликий нахил назад.

Створення сприятливих умов праці і правильне естетичне оформлення робочих місця на виробництві має велике значення як для полегшення праці, так і для підвищення його привабливості, позитивно впливає на продуктивність праці

Забарвлення приміщень і меблів повинні сприяти створенню сприятливих умов для здорового сприйняття, гарного настрою. У службових приміщеннях, у яких виконується одноманітна розумова робота, що вимагає значної нервової напруги і великого зосередження, фарбування повинна бути спокійних тонів - малонасичені відтінки холодного зеленого або блакитного кольорів.

#### **4.3. Заходи техніки безпеки на заводі**

Під технікою безпеки мається на увазі комплекс заходів технічного та організаційного характеру, спрямованих на створення безпечних умов праці та запобігання нещасним випадкам на виробництві.

На рибзаводі є спеціальна служба безпеки, підпорядкована головному інженеру, що розробляє заходи, які мають забезпечити робітникам безпечні умови роботи, контролює стан техніки безпеки на виробництві та стежить за новими робітниками заводу, тобто вони мають бути навчені безпечним прийомам роботи.

Проводяться інструктажі (первинний, позаплановий, повторний, цільовий) з техніки безпеки та охорони праці згідно нормативів, передбачених законодавством.

Існують технічні та індивідуальні засоби безпеки.

### Індивідуальні.

Відповідно до роду виконуваних робіт видається спецодяг, який після закінчення терміну використання списується. До такого спецодягу належать: гумові рукавиці, гумові чоботи, спеціальний головний убір, комбінезон, фартух з нагрудником.

### Технічні.

Безпека технологічного обладнання забезпечується правильним вибором принципів дії, конструктивних систем, матеріалів, робочих процесів.

Крім цього, безпека праці забезпечується:

- використанням у конструкціях технологічного обладнання спеціальних захисних засобів;
- дотриманням ергономічних вимог;
- включенням вимог безпеки в технічну документацію з монтажу, експлуатації, ремонту машин та обладнання.

## **4.4. Пожежна безпека відділу**

Забороняється зберігання різних матеріалів, готової продукції біля стін будівель і споруд, а також у протипожежних розривах між будівлями та устаткуванням. Допоміжні матеріали в цехах зберігаються в коморах у кількостях, обумовлених цеховими виробничими інструкціями.

Цехи, майстерня, адміністративні приміщення мають первинні засоби пожежогасіння: вогнегасник, пісок, відро та лопата . Відповідальні за пожежну безпеку особи відправляють на перезарядку вогнегасники (у міру використання, періодично через 5 років, при відсутності пломб і табличок і т.п.).

У кожному цеху, поверсі побутових та адміністративних будинків вивішені схеми евакуації людей на випадок пожежі, таблички із зазначенням прізвищ відповідальних за пожежну безпеку осіб.

Заборонено складування сторонніх предметів, металу на електричних щитах, обладнанні, у вентиляційних камерах. Всі роботи, пов'язані із застосуванням відкритого вогню (електро-і газозварювальні) у підрозділах і на території дозволено проводити тільки за наявності спеціального наряду-допуску (дозволу) на виробництво вогневих робіт.

## **ВИСНОВОК**

Під час виконання кваліфікаційної роботи, була досліджена робота ремонтного відділу рибзаводу. В роботі виконана та протестована інформаційна система для ТОВ «Рибний Комбінат Херсон». Додатково була створена інструкція для користувача для даної інформаційної системи для полегшення впровадження її для співробітників.

В результаті, був створений сайт для ремонтного відділу заводу. За допомогою мови Web-програмування - PHP було розроблено інтерфейс користувача сайту. Він прискорює всю роботу ,тобто полегшує розподіл роботи між бригадами та співробітниками , також для розставлення коректної пріоритетності виконання заявок на ремонт.

Даний сайт є універсальним для задач зберігання інформації про робітників, обладнання на заводі, накладні, заявок, звітів а отже при певному коригуванні структури БД може застосовуватися на підприємствах різного профілю.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. <https://atlant2003.ru/sluzhba-glavnogo-inzhenera-na-proizvodstve/>
2. [http://www.dut.edu.ua/en/uploads/1\\_842\\_23775847.pdf](http://www.dut.edu.ua/en/uploads/1_842_23775847.pdf)
3. <http://trudova-ohrana.ru/primery-dokumentov/primery-polozhenij-po-predpriyatiju-/1750-polozhenie-ob-otdele-glavnogo-mehanika.html>
4. МЕТОДОЛОГІЯ IDEF0 [Електронний ресурс] – Режим доступу до ресурсу: [https://stud.com.ua/87184/ekonomika/metodologiya\\_idef0](https://stud.com.ua/87184/ekonomika/metodologiya_idef0).
5. Законодавство України про охорону праці .
6. Про охорону праці [Електронний ресурс] – Режим доступу до ресурсу: <https://zakon.rada.gov.ua/laws/show/2694-12>.
7. [https://neumeika.ru/pravila\\_raboty\\_za\\_kompyuterom.html](https://neumeika.ru/pravila_raboty_za_kompyuterom.html)
8. <https://profiteh.ua/ru/pozharnaya-bezopasnost-na-predpriyatii-pravila-i-organizatsiya/>
9. <https://www.php.net/manual/ru/index.php>
10. [https://youcontrol.com.ua/ru/catalog/company\\_details/42771160/](https://youcontrol.com.ua/ru/catalog/company_details/42771160/)
11. <https://core.ac.uk/reader/86920640>
12. ГОСТ Р 50948, 49-96. Загальні ергономічні вимоги і вимоги безпеки і її параметри для ЕОМ.
13. <https://wiki.rookee.ru/cms/>
14. <https://www.php.net/>
15. <https://www.activetraffic.ru/wiki/cms/>

## ДОДАТОК А

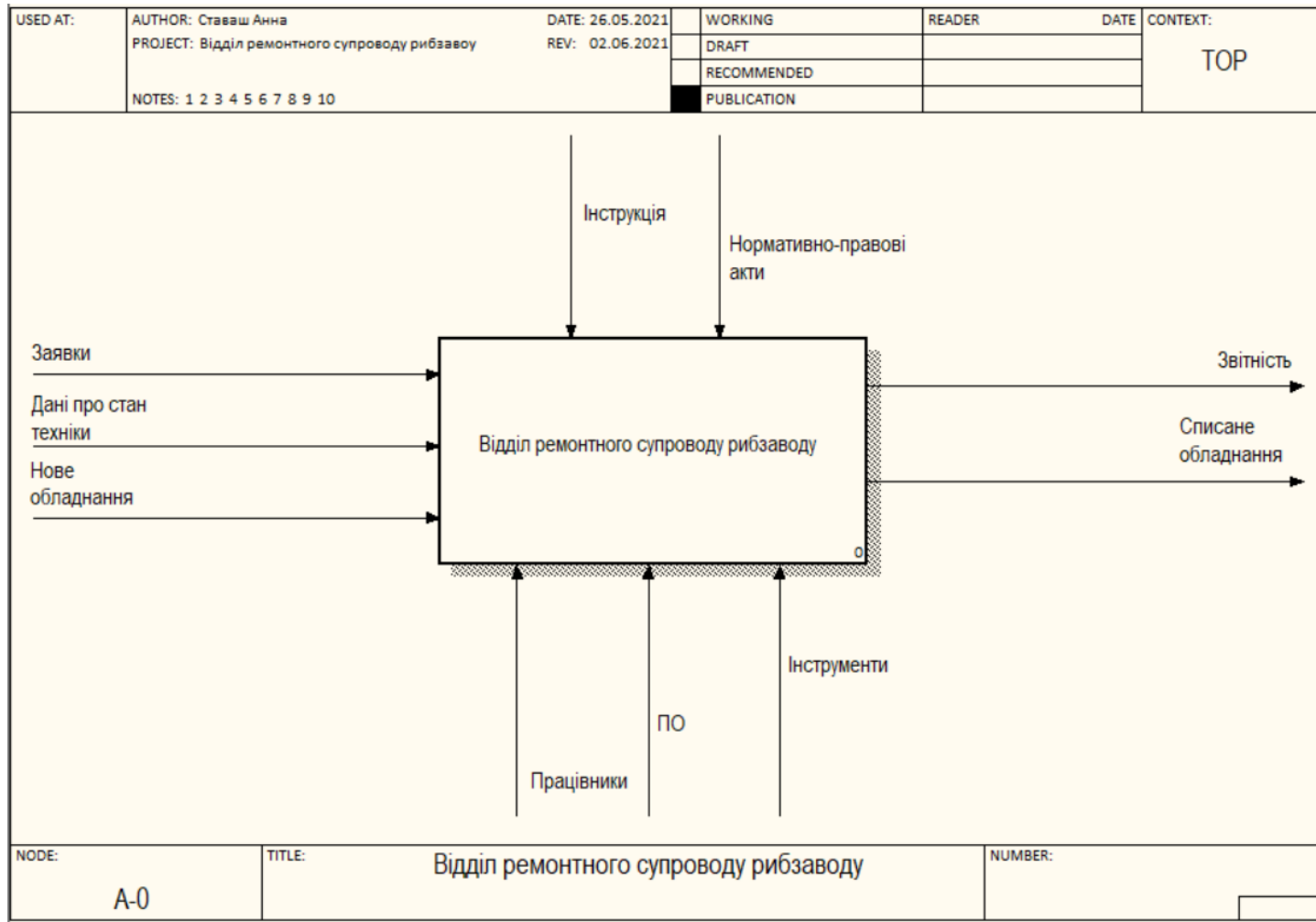


Рис. А.1 – Модель «AS-IS». Контексна діаграма

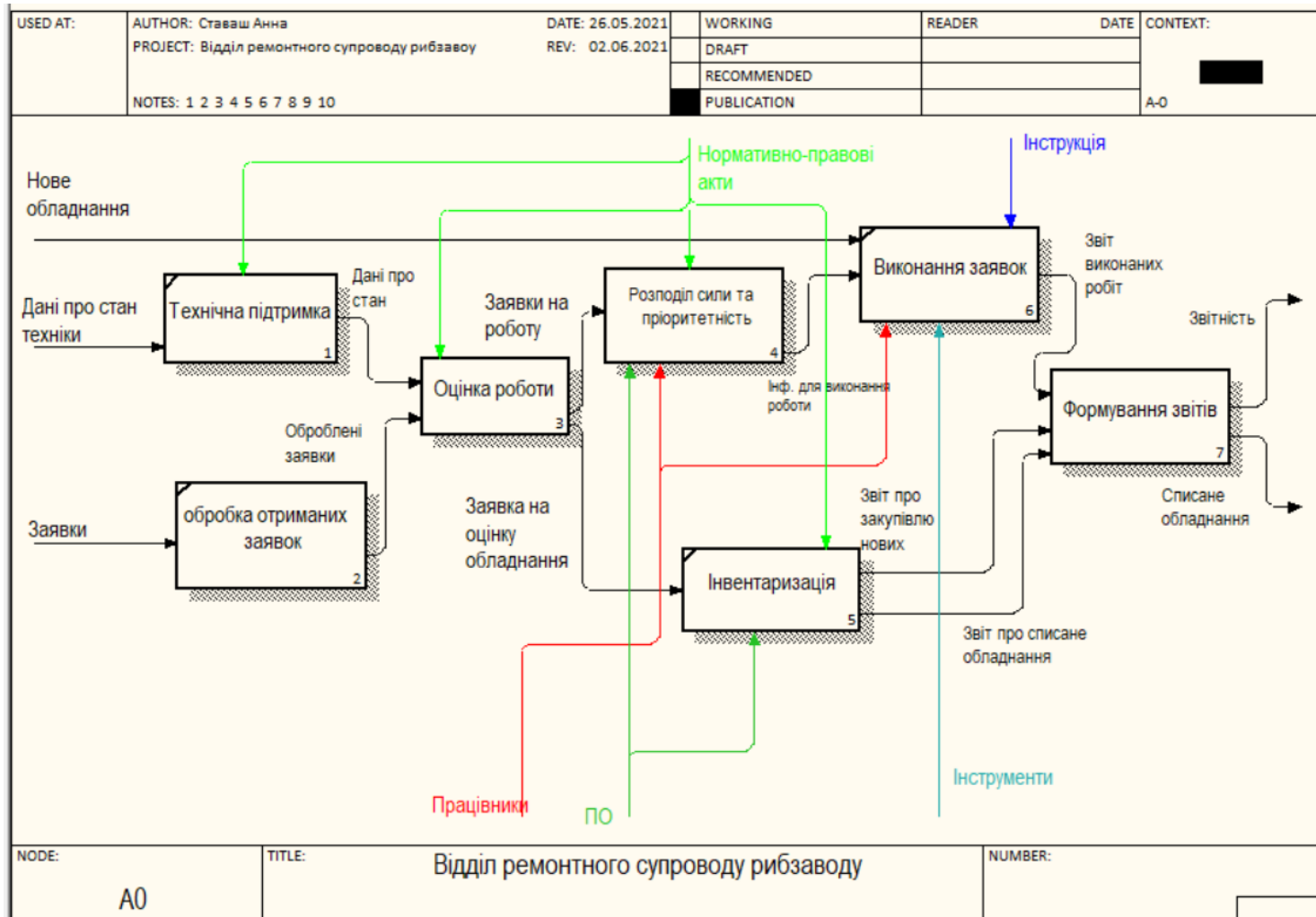


Рис. А.2 – Модель «AS-IS». Функціональна модель діяльності

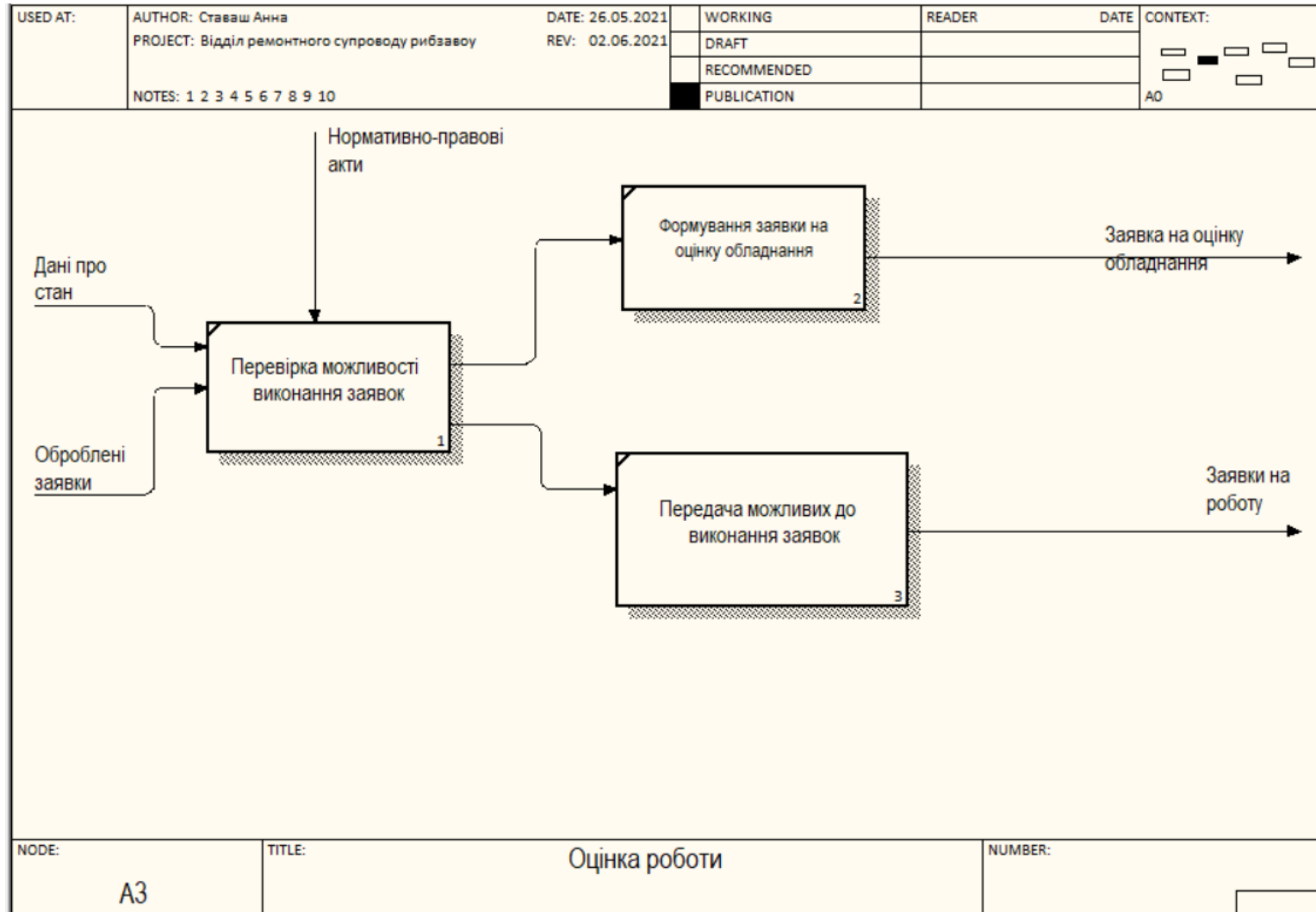


Рис. А.3 – Модель «AS-IS». Діаграма декомпозиції II рівня

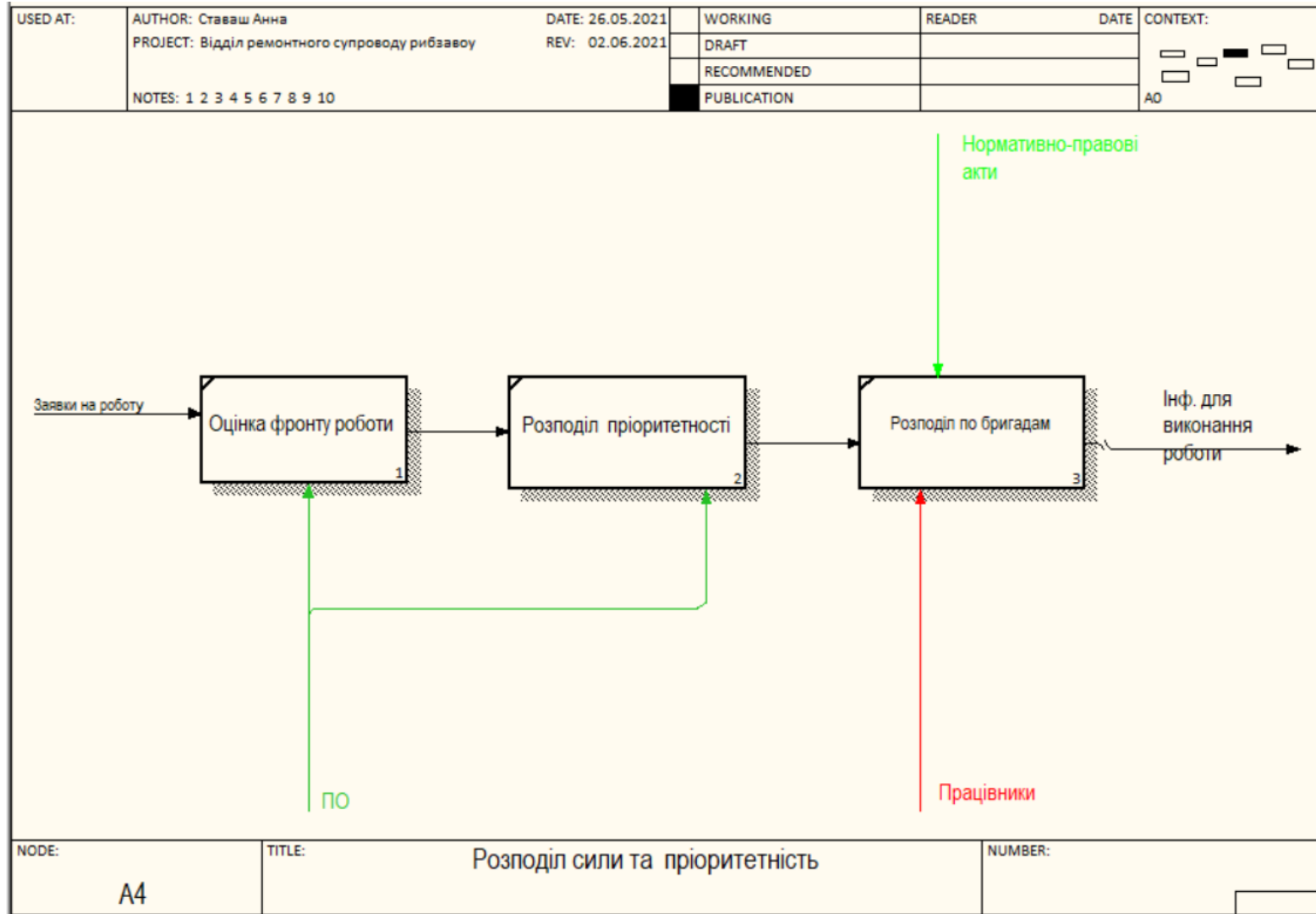


Рис. А.4 – Модель «AS-IS». Діаграма декомпозиції II рівня

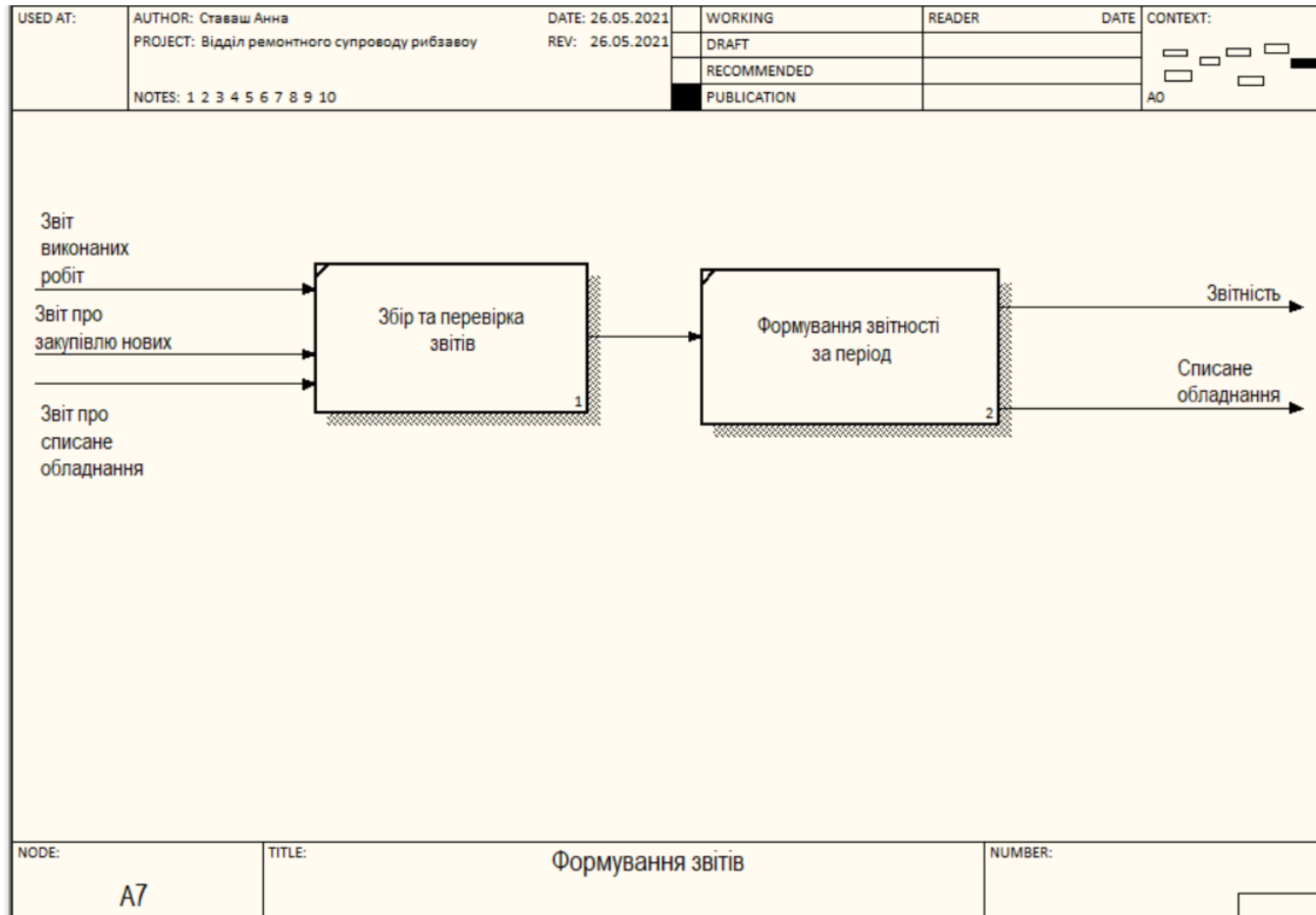


Рис. А.5 – Модель «AS-IS». Діаграма декомпозиції II рівня

## ДОДАТОК Б

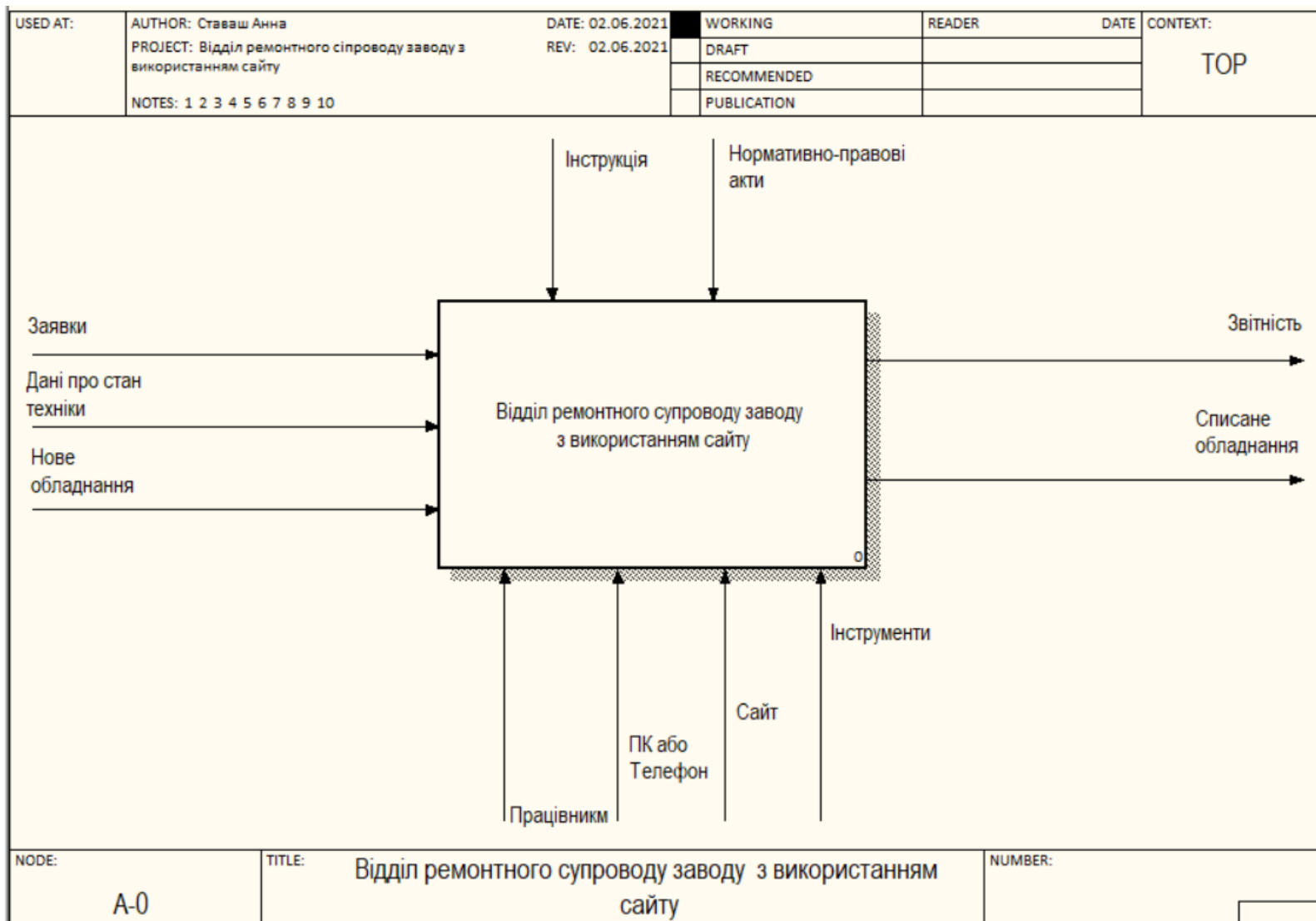


Рис. Б.1 – Модель «ТО-ВЕ». Контекстна діаграма

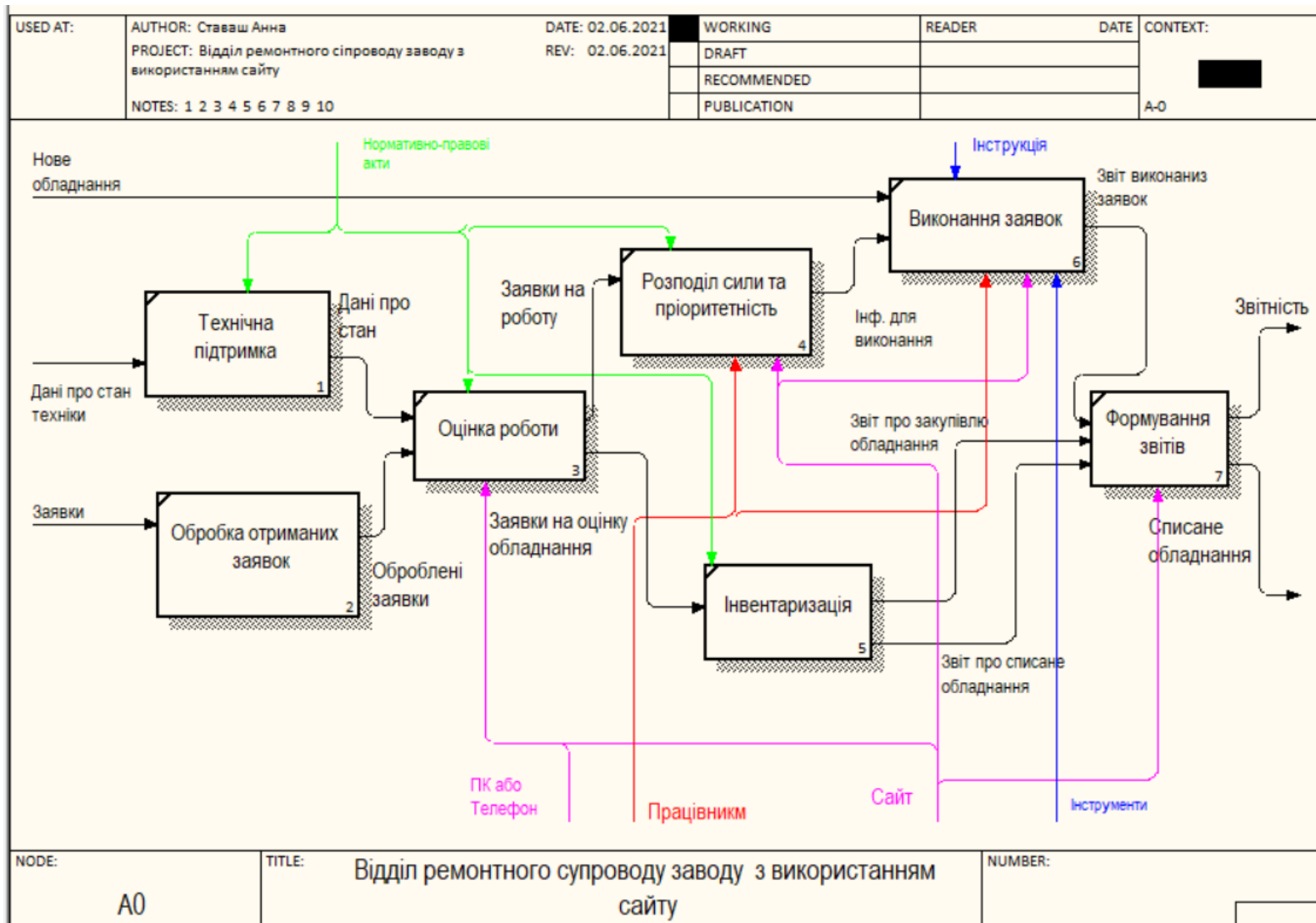


Рис. Б.2 – Модель «ТО-VE». Функціональна модель діяльності

## ДОДАТОК В

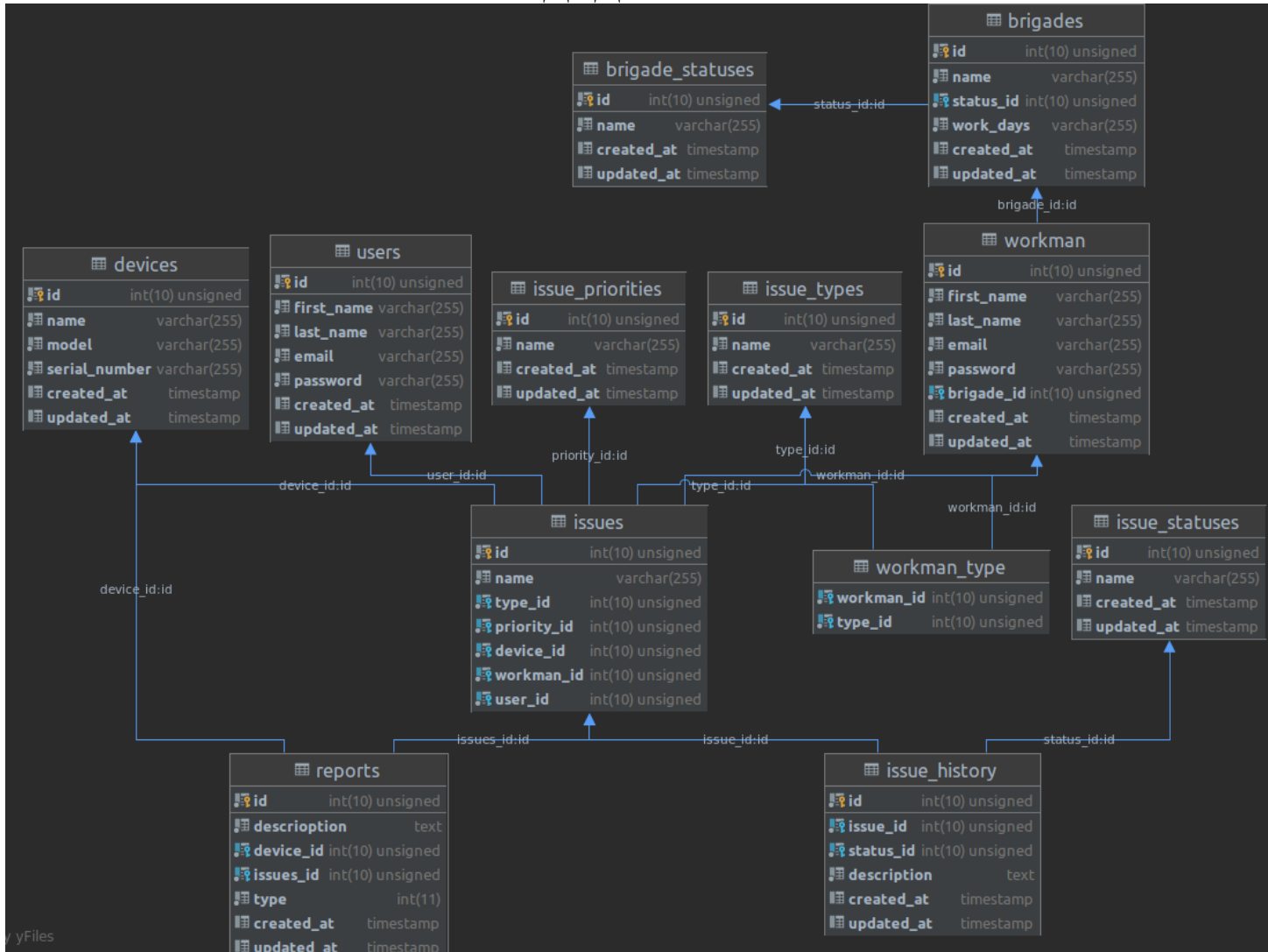


Рис.В.1. Схема БД у MS SQL Server 2014

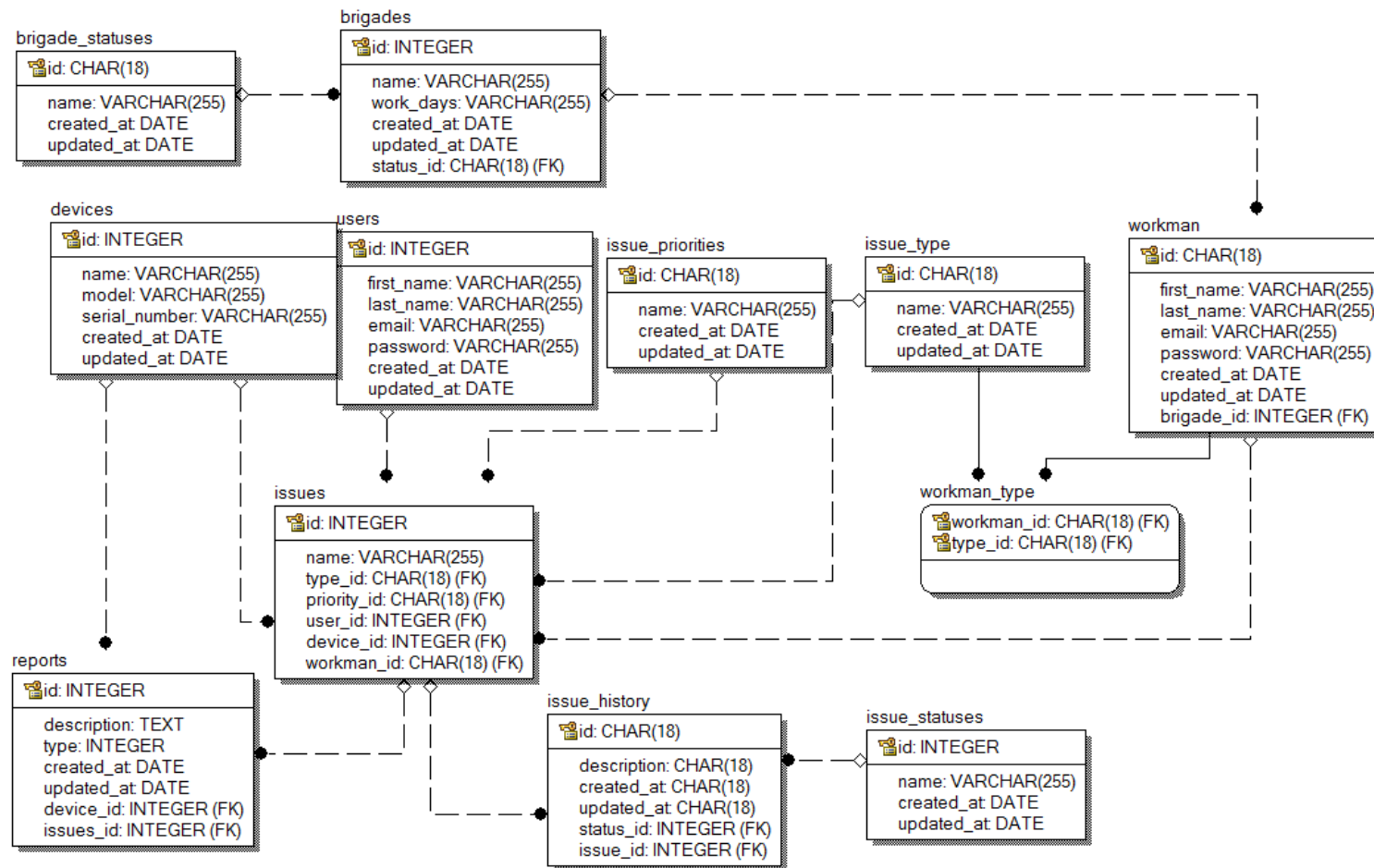


Рис.В.2. Логічна модель БД

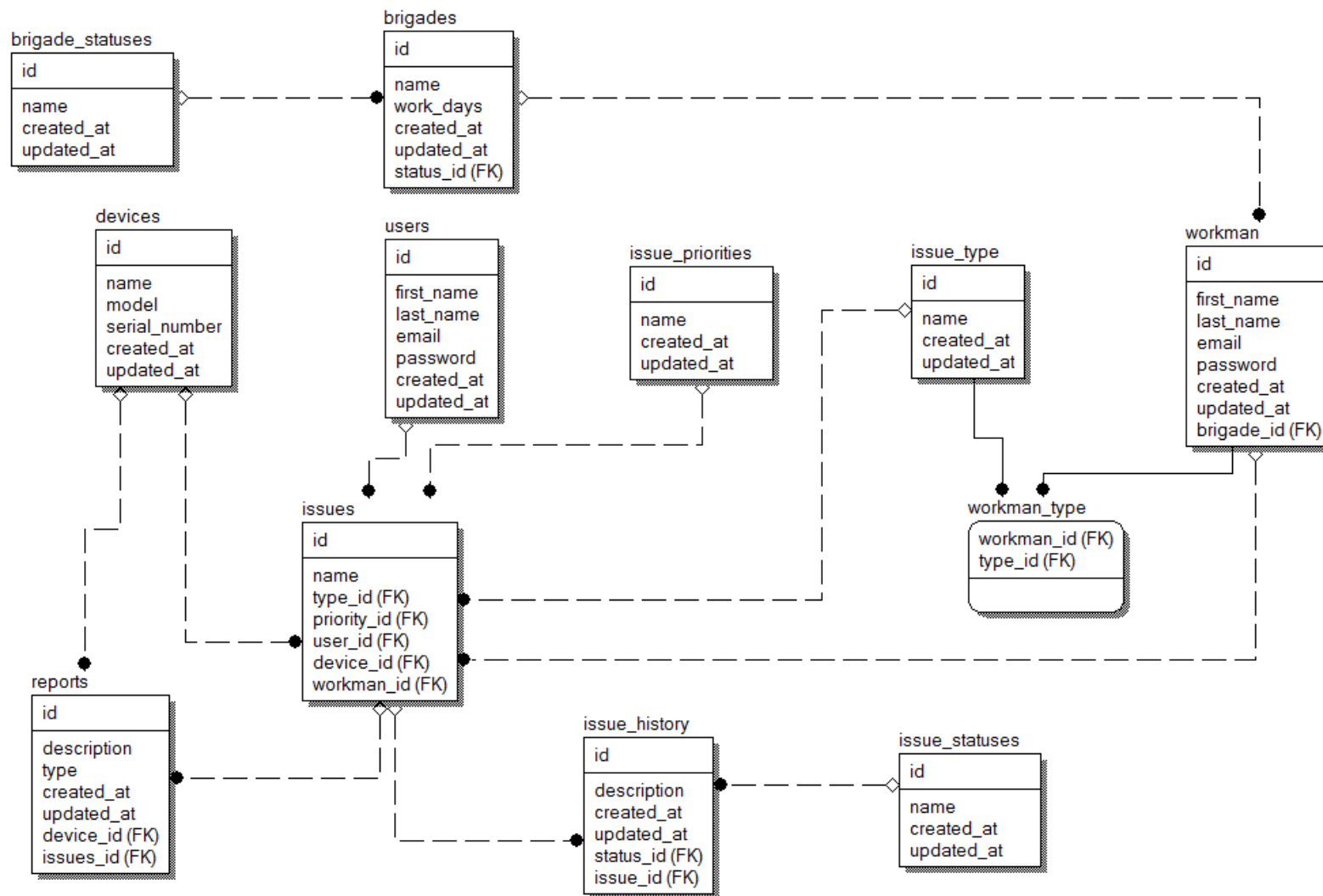


Рис.В.3. Фізична модель БД.

## ДОДАТОК Г

**Фрагменти коду програми, який обробляє запит отримання завдань, запрошувати їх з бази даних:**

```
<?php
namespace App\Http\Controllers;
use Symfony\Component\HttpFoundation\Request;
/**
 * Class IssueController
 * @package App\Http\Controllers
 */
class IssueController
{
    /**
     * @var IssueRepository
     */
    private $issueRepository;
    /**
     * IssueController constructor.
     * @param IssueRepository $issueRepository
     */
    public function __construct(IssueRepository $issueRepository)
    {
        $this->issueRepository = $issueRepository;
    }
    /**
     * @param Request $request
     *
     * @return
     * \Illuminate\Contracts\Routing\ResponseFactory|\Illuminate\Foundation\Application|\I
     *lluminate\Http\Response
```

```

    */
    public function getTableData(Request $request)
    {
        $issues = $this->issueRepository->getTableDataFromDB(
            $request->get('filter'),
            $request->get('order')
        );
        $result = [
            'items' => $issues->mapItemFromDB(),
            'count' => $issues->count()
        ];
        return response(json_decode($result), 200);
    }
}
<?php

```

```

namespace App\Http\Controllers;
use Illuminate\Database\Eloquent\Model;
/**
 * Class Issue
 * @package App\Http\Controllers
 */
class Issue extends Model
{
    /**
     * @var string
     */

```

```

public $table = 'issues';

/**
 * @var string[]
 */
public $fillable = [
    'name',
    'type_id',
    'priority_id',
    'device_id',
    'user_id',
    'workman_id'
];

public $casts = [
    'user_id' => 'int',
    'workman_id' => 'int',
    'type_id' => 'int',
    'device_id' => 'int'
];

/**
 * @return \Illuminate\Database\Eloquent\Relations\BelongsTo
 */
public function user()
{
    return $this->belongsTo(User::class);
}

/**

```

```

* @return \Illuminate\Database\Eloquent\Relations\BelongsTo
*/
public function workman()
{
    return $this->belongsTo(Workman::class);
}
/**
* @return \Illuminate\Database\Eloquent\Relations\BelongsTo
*/
public function type()
{
    return $this->belongsTo(IssueType::class);
}

/**
* @return \Illuminate\Database\Eloquent\Relations\BelongsTo
*/
public function priority()
{
    return $this->belongsTo(IssuePriority::class);
}

/**
* @return \Illuminate\Database\Eloquent\Relations\BelongsTo
*/
public function device()
{
    return $this->belongsTo(Device::class);
}

```

```

}
/**
 * @return \Illuminate\Database\Eloquent\Relations\HasMany
 */
public function issueHistory()
{
    return $this->hasMany(IssueHistory::class);
}
}

```

### **Пошук заявки:**

```

select issues.id,
        issues.name,
        it.name           as `type_name`,
        ip.name           as `priority_name`,
        d.name            as `device_name`,
        (SELECT i.name
        from issue_history
            left join issue_statuses i on i.id = issue_history.status_id
        where issue_history.issue_id = issues.id
        order by issue_history.created_at
        LIMIT 1)          as `status`,
        CONCAT(u.first_name, ' ', u.last_name) as `user_name`,
        CONCAT(w.first_name, ' ', w.last_name) as `workman`
from issues
        left join workman w on issues.workman_id = w.id
        left join users u on issues.user_id = u.id

```

```
left join issue_types it on it.id = issues.type_id
```

```
left join issue_priorities ip on issues.priority_id = ip.id
```

```
left join devices d on issues.device_id = d.id
```

### Пошук бригади:

```
select brigades.name,  
  
       bs.name                               as `status`,  
  
       GROUP_CONCAT(CONCAT(w.first_name, ' ', w.last_name) SEPARATOR ',')  
as `workman`,  
  
       GROUP_CONCAT(DISTINCT it.name SEPARATOR ',')           as `types`,  
  
       brigades.work_days,  
  
       COUNT(w.id)                                           as `count_workman`,  
  
       DATE_FORMAT(brigades.created_at, "%d %M, %Y")         as `created_at`,  
  
       DATE_FORMAT(brigades.updated_at, "%d %M, %Y")        as `update_at`  
  
from brigades  
  
       join workman w on brigades.id = w.brigade_id  
  
       join workman_type wt on w.id = wt.workman_id  
  
       join issue_types it on wt.type_id = it.id  
  
       join brigade_statuses bs on brigades.status_id = bs.id  
  
group by brigades.id
```

### Код БД

```
- Host: localhost Database: crm_diplom
```

```
-----
```

```
-- Server version 5.7.34
```

```
/*!40101 SET  
@OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
```

```
/*!40101 SET  
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
```

```

/*!40101 SET @OLD_COLLATION_CONNECTION=@ @COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;
/*!40103 SET @OLD_TIME_ZONE=@ @TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@ @UNIQUE_CHECKS,
UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@ @FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@ @SQL_MODE,
SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@ @SQL_NOTES, SQL_NOTES=0 */;
--
-- Table structure for table `brigade_statuses`
--
DROP TABLE IF EXISTS `brigade_statuses`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `brigade_statuses` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `name` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `created_at` timestamp NULL DEFAULT NULL,
  `updated_at` timestamp NULL DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_unicode_ci;
/*!40101 SET character_set_client = @saved_cs_client */;
--
-- Table structure for table `brigades`

```

```

--
DROP TABLE IF EXISTS `brigades`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `brigades` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `name` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `status_id` int(10) unsigned NOT NULL,
  `work_days` json NOT NULL,
  `created_at` timestamp NULL DEFAULT NULL,
  `updated_at` timestamp NULL DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `brigades_status_id_foreign` (`status_id`),
  CONSTRAINT `brigades_status_id_foreign` FOREIGN KEY (`status_id`)
REFERENCES `brigade_statuses` (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_unicode_ci;
/*!40101 SET character_set_client = @saved_cs_client */;
--
-- Table structure for table `devices`
--
DROP TABLE IF EXISTS `devices`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `devices` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `name` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `model` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,

```

```

`serial_number` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
`created_at` timestamp NULL DEFAULT NULL,
`updated_at` timestamp NULL DEFAULT NULL,
PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=16 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_unicode_ci;
/*!40101 SET character_set_client = @saved_cs_client */;
--
-- Table structure for table `issue_history`
--
DROP TABLE IF EXISTS `issue_history`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `issue_history` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `issue_id` int(10) unsigned NOT NULL,
  `status_id` int(10) unsigned NOT NULL,
  `description` text COLLATE utf8mb4_unicode_ci NOT NULL,
  `created_at` timestamp NULL DEFAULT NULL,
  `updated_at` timestamp NULL DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `issue_history_issue_id_foreign` (`issue_id`),
  KEY `issue_history_status_id_foreign` (`status_id`),
  CONSTRAINT `issue_history_issue_id_foreign` FOREIGN KEY (`issue_id`)
REFERENCES `issues` (`id`),
  CONSTRAINT `issue_history_status_id_foreign` FOREIGN KEY (`status_id`)
REFERENCES `issue_statuses` (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_unicode_ci;

```

```

/*!40101 SET character_set_client = @saved_cs_client */;
--
-- Table structure for table `issue_priorities`
--
DROP TABLE IF EXISTS `issue_priorities`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `issue_priorities` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `name` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `created_at` timestamp NULL DEFAULT NULL,
  `updated_at` timestamp NULL DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_unicode_ci;
/*!40101 SET character_set_client = @saved_cs_client */;
--
-- Table structure for table `issue_statuses`
--
DROP TABLE IF EXISTS `issue_statuses`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `issue_statuses` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `name` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `created_at` timestamp NULL DEFAULT NULL,
  `updated_at` timestamp NULL DEFAULT NULL,
  PRIMARY KEY (`id`)

```

```
) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_unicode_ci;
```

```
/*!40101 SET character_set_client = @saved_cs_client */;
```

```
--
```

```
-- Table structure for table `issue_types`
```

```
--
```

```
DROP TABLE IF EXISTS `issue_types`;
```

```
/*!40101 SET @saved_cs_client = @@character_set_client */;
```

```
/*!40101 SET character_set_client = utf8 */;
```

```
CREATE TABLE `issue_types` (  
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
```

```
  `name` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
```

```
  `created_at` timestamp NULL DEFAULT NULL,
```

```
  `updated_at` timestamp NULL DEFAULT NULL,
```

```
  PRIMARY KEY (`id`)
```

```
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_unicode_ci;
```

```
/*!40101 SET character_set_client = @saved_cs_client */;
```

```
--
```

```
-- Table structure for table `issues`
```

```
--
```

```
DROP TABLE IF EXISTS `issues`;
```

```
/*!40101 SET @saved_cs_client = @@character_set_client */;
```

```
/*!40101 SET character_set_client = utf8 */;
```

```
CREATE TABLE `issues` (  
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
```

```
  `name` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
```

```
  `type_id` int(10) unsigned NOT NULL,
```

```

`priority_id` int(10) unsigned NOT NULL,
`device_id` int(10) unsigned NOT NULL,
`workman_id` int(10) unsigned NOT NULL,
`user_id` int(10) unsigned NOT NULL,
PRIMARY KEY (`id`),
KEY `issues_type_id_foreign` (`type_id`),
KEY `issues_priority_id_foreign` (`priority_id`),
KEY `issues_device_id_foreign` (`device_id`),
KEY `issues_workman_id_foreign` (`workman_id`),
KEY `issues_user_id_foreign` (`user_id`),
CONSTRAINT `issues_device_id_foreign` FOREIGN KEY (`device_id`)
REFERENCES `devices` (`id`),
CONSTRAINT `issues_priority_id_foreign` FOREIGN KEY (`priority_id`)
REFERENCES `issue_priorities` (`id`),
CONSTRAINT `issues_type_id_foreign` FOREIGN KEY (`type_id`)
REFERENCES `issue_types` (`id`),
CONSTRAINT `issues_user_id_foreign` FOREIGN KEY (`user_id`)
REFERENCES `users` (`id`),
CONSTRAINT `issues_workman_id_foreign` FOREIGN KEY (`workman_id`)
REFERENCES `workman` (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_unicode_ci;
/*!40101 SET character_set_client = @saved_cs_client */;
--
-- Table structure for table `migrations`
--
DROP TABLE IF EXISTS `migrations`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;

```

```

CREATE TABLE `migrations` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `migration` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `batch` int(11) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_unicode_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `reports`
--

DROP TABLE IF EXISTS `reports`;

/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;

CREATE TABLE `reports` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `description` text COLLATE utf8mb4_unicode_ci NOT NULL,
  `device_id` int(10) unsigned NOT NULL,
  `issues_id` int(10) unsigned NOT NULL,
  `type` int(11) NOT NULL,
  `created_at` timestamp NULL DEFAULT NULL,
  `updated_at` timestamp NULL DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `reports_device_id_foreign` (`device_id`),
  KEY `reports_issues_id_foreign` (`issues_id`),

```

```

    CONSTRAINT `reports_device_id_foreign` FOREIGN KEY (`device_id`)
REFERENCES `devices` (`id`),

    CONSTRAINT `reports_issues_id_foreign` FOREIGN KEY (`issues_id`)
REFERENCES `issues` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_unicode_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `users`
--

DROP TABLE IF EXISTS `users`;

/*!40101 SET @saved_cs_client = @@character_set_client */;

/*!40101 SET character_set_client = utf8 */;

CREATE TABLE `users` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `first_name` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `last_name` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `email` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `password` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `created_at` timestamp NULL DEFAULT NULL,
  `updated_at` timestamp NULL DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=17 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_unicode_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Table structure for table `workman`

```

```

--
DROP TABLE IF EXISTS `workman`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `workman` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `first_name` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `last_name` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `email` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `password` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL,
  `brigade_id` int(10) unsigned NOT NULL,
  `created_at` timestamp NULL DEFAULT NULL,
  `updated_at` timestamp NULL DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `workman_brigade_id_foreign` (`brigade_id`),
  CONSTRAINT `workman_brigade_id_foreign` FOREIGN KEY (`brigade_id`)
REFERENCES `brigades` (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=17 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_unicode_ci;
/*!40101 SET character_set_client = @saved_cs_client */;
--
-- Table structure for table `workman_type`
--
DROP TABLE IF EXISTS `workman_type`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `workman_type` (
  `workman_id` int(10) unsigned NOT NULL,

```

```

`type_id` int(10) unsigned NOT NULL,
KEY `workman_type_workman_id_foreign` (`workman_id`),
KEY `workman_type_type_id_foreign` (`type_id`),
CONSTRAINT `workman_type_type_id_foreign` FOREIGN KEY (`type_id`)
REFERENCES `issue_types` (`id`),
CONSTRAINT `workman_type_workman_id_foreign` FOREIGN KEY
(`workman_id`) REFERENCES `workman` (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_unicode_ci;

/*!40101 SET character_set_client = @saved_cs_client */;

/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;

/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;

/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;

/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;

/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

```