**ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**

**ІМЕНІ ІВАНА ФРАНКА**

Факультет прикладної математики та інформатики

Кафедра інформаційних систем

**КУРСОВА РОБОТА**

на тему:

**«АВТОМАТИЗАЦІЯ ЗБОРУ ТА ОБРОБКИ ІНФОРМАЦІЇ МУЛЬТИМЕДІЙНИМИ ЗАСОБАМИ»**

Студентки 3 курсу групи МПі-32

Напряму підготовки Інформатика

Спеціалізація: Інформатика

Йовбак А.В.

Керівник: Вовк В.Д.

Національна шкала \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кількість балів \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Оцінка ECTS \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Львів, 2019

**ЗМІСТ**

АНОТАЦІЯ 2

ВСТУП 3

1. ПОСТАНОВКА ЗАДАЧІ 5

1.1. Фізична постановка задачі 5

1.2. Програмна постановка задачі 7

2. МЕТОДИКА ВИРІШЕННЯ 10

3. РЕАЛІЗАЦІЯ 12

3.1. Мова розмітки даних HTML 12

3.2. Мова візуалізації CSS 13

3.3. Мова програмування JavaScript 14

3.4. Фрейморк ASP.NET CORE 15

3.5. База даних MS SQL 17

3.6. Інструмент Android Debag Bridge 19

4. АПРОБАЦІЯ 20

ПІДСУМКИ І ВИСНОВКИ 22

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 23

ДОДАТКИ 24

# АНОТАЦІЯ

У курсовій роботі було розроблено програмний продукт для прийняття даних лічильників шляхом зчитування вхідних sms-повідомлень від відповідних користувачів (зареєстрованих працівників), адміністрування системи, перегляду статусу та взаємодії між користувачем та адміністратором. Проведено аналіз та огляд існуючих реалізацій подібних систем. Проведено розробку та реалізовано веб-систему, що дозволяє здійснювати віддалене управляння. Проведено тестування програми на предмет ефективності та придатності до використання. Розроблено рекомендації щодо використання програмного продукту.

Розробка проводилась мовою програмування C# у середовищі Microsoft Visual Studio 2017.

# ВСТУП

В наш час неоціненно важливу роль займають інформаційні технології. Саме вони дали нові можливості для роботи і відпочинку, багато в чому полегшили працю і просто життя кожної сучасної людини. Теперішнє суспільство навряд чи можна уявити без Інтернету або засобів масової інформації.

Застосування комп’ютерів стало буденною справою, хоча ще зовсім недавно робоче місце, обладнане ним, було великою рідкістю. На сьогодні практично неможливо знайти сферу, в якій не використовуються можливості інформаційні технологій. Вони дуже швидко перетворилися на життєво важливий стимул розвитку не тільки світової економіки, а й інших сфер людської діяльності.

Зараз новітні технології впроваджуються на багатьох підприємствах, організаціях та різних органах влади, відбувається оперування великими кількостями даних, їх зберігання та опрацювання. Перед людьми постає задача : «Як організувати та автоматизувати цей процес? Як полегшити життя використавши всі можливості існуючих мультимедійних та інформаційних засобів?». Саме через це зараз великий попит мають системи, розроблені саме для спрощення збору інформації, а також її фільтрації, сортування та обробки.

Давайте відійдемо трохи назад, в ті часи коли до кожного в хату приходили державні працівники, записували дані ваших лічильників в журнал, потім відправляли їх для опрацювання. Це займало багато часу, часто бувало так, що дані губились, були записані неправильно... Одним словом процес був довготривалим та складним. Саме це і надихнуло мене на створення такої системи, яка б принесла користь та зробила життя кращим та яскравішим! Адже позбавившись рутинності, кожен буде мати набагато більше вільного часу для себе та здійснення своїх мрій.

Таким чином предметом роботи є розроблювана веб-система «PAL-System» для автоматизованого збору та адміністрування даних показників лічильників за газ, воду, та електроенергію. З використанням інструменту ADB, технології ASP.NET Core 2 та патернів MVC.

Метою роботи є створення високоякісного програмного продукту, який став би досить привабливою та конкурентоспроможною системою, яка б спрощувала процес збору даних та їх обробки для приватних підприємств та організацій.

# ПОСТАНОВКА ЗАДАЧІ

## **Фізична постановка задачі**

В будь якій галузі, яка проводить збір та оперування великими кількостями даних, потрібна автоматизація та спрощення основних процесів, якщо ж кількість інформації істотно велика, то це перетворюється просто в необхідність.

В таких ситуаціях на допомогу приходять системи, які і розраховані виконувати вище наведені вимоги. Як прикладом таких систем можуть бути :

* газова компанія 104.ua
* телекомунікаційна компанія «Укртелеком»
* головна енергорозподільча компанія Закарпаття - ПАТ "Закарпаттяобленерго" і т.д.

Саме цими веб-системами користується моє сім’я, адже там є можливість самостійно відправити дані показників, побачити статистику та інфографіку, що значно збільшує користь та зручність наведених систем.

В рамках цієї роботи було обрано розробку інформаційної веб-системи «PAL-System» для компанії PAL (**P**erfect **A**partment for **L**iving), яка займається побудовою комфортних будинків. Основна задача розролюваного продукту є спрощення процесу збору даних водо-, газо- та електролічильників, їх опрацювання та виведення інфографіки за вибраний період часу.

Основними елементами такої системи є користувачі, працівники, типи даних лічильників, а також самі дані, обробка над якими і буде здійснена.

*Користувачі* – це особи, які користуються послугами компанії PAL, тобто люди у яких беруть дані лічильників для опрацювання.

*Працівники* – особи, які відправляють показники смс повідомлення на гтелефон-приймач.

Дані про людські ресурси повинні містити таку інформацію як:

* ПІБ
* Тип показника (сторона, яка надає послуги)
* Номер особистого рахунку (сторона, яка приймає послуги)
* Email (сторона, яка приймає послуги)
* Номер мобільного телефону (сторона, яка надає послуги)

Тип показника містить в собі назву (газ, вода, електроенергія) та обмеження, які існують для відповідного тарифу користувача.

Самі дані для обробки (відправлені показники) поєднують в собі інформацію про працівника, який відправив дані лічильника, самі дані, номер особового рахунку жителя PAL та час відправлення показника.

Для найбільшої зручності всіх працівників компанії, відправка даних відбувається шляхом надсилання смс-повідомлень на номер головного телефону (+38\*\*\*\*\*\*\*014), з якого інформація зчитується і заносяться в базу даних. Сама система проводить обробку показників, їх сортування, класифікування та вивід інфографіки.

«PAL-System» повинна бути спроектованою і мати зручні з точки зору дизайну засоби навігації та відображення інформації. В ні не повинно бути сторонньої інформації, яка б відволікала працівників від роботи. А лише та інформація і той функціонал доступ до якого вони мають згідно зі своєю посадою та обов’язками.

## **Програмна постановка задачі**

Програмна постановка задачі полягає в програмуванні інтерфейсу інформаційної веб-системи та забезпечення повноцінного функціоналу, який відобразить всі переваги автоматизації та обробки даних мультимедійними засобами.

При розробці системи повинні використовуватись сучасні засоби проектування складних систем. А також сучасні засоби розробки, які дають змогу компонувати і організовувати можливість використовувати різних форматів файлів та мов програмування в одному проекті.

Для такої системи, яка б могла працювати в реальних умовах, для реальної компанії необхідно мати досить потужний сервер із достатньою кількістю оперативної пам’яті та жорстким диском, який вміщував в собі базу даних (БД) в які буде зберігатись вся необхідна для роботи інформація. Також важливим є швидкісний Інтернет з надійним каналом, щоб забезпечити доступ до веб-системи не лише з локальної мережі, а й з будь-якої точки планети де є більш-менш швидкий Інтернет. Неоціненним є фактор того, що телефон, на який приходять смс-повідомлення повинен завжди бути в полі з хорошим зв’язком.

Отже, враховуючи вище наведені вимоги і наші можливості, ми отримуємо наступний результат.

Параметри мультимедійного девайсу (в нашому випадку модель «головного телефону» - Xiomi Redmi 3S Prime):

* + - * CPU : Octa-core 1.4 Ггц ARM Cortex-A53
      * чіпсет : Qualcomm MSM8937 Snapdragon 430
      * пам’ять RAM : 3 Гб
      * операційна система : Android OS, v6.0.1 (Marshmallow)

Параметри комп’ютеру, на якому проводиться розробка веб-системи:

* процесор: AMD A6-4400M APU with Radeon(tm) HD Grsphics 2.70 GHz
* тип системи: 64-розрядна
* операційна система: Windows 8.1

З точки зору функціональних параметрів така система повинна підтримувати:

* багатокористувацький режим роботи;
* мати можливість зручного відображення інформації для користувача;
* можливість фільтрування даних за різними критеріями;
* можливість створювати нові типи інформаційного наповнення та використовувати їх в інших частинах системи;
* підтримувати функції додавання, редагування та видалення інформації;
* повинна мати в собі функціональність, яка опрацьовує дані та показує статистику та інфографіку;
* реалізовувати зчитування вхідних повідомлень, які задовольняють визначені критерії та їх додавання у базу даних у відповідному форматі.

Система, що розробляється повинна мати змогу оперувати зі значними масивами даних, використовуючи при цьому, якомога меншу кількість ресурсної потужності комп’ютера.

Всі перераховані вище вимоги до функціювання системи та дизайну можна реалізувати використовуючи сучасні середовища web-розробки. Для програмування системи використовуємо мову програмування C#, фреймворк ASP .NET CORE та середовище Microsoft Visual Studio. Для розробки User Interface використовуємо можливості мови розмітки HTML, таблиці стилів CSS та мови програмування JavaScript. JS використовується не тільки для покращення зовнішнього вигляду, а також для передачі даних з фронд-енд частини до бек-енду, використовуючи ajax-запити. Для організації баз даних використовуємо MS SQL Server. Зчитування системою вхідних повідомлень та їх запис в txt-файл відбувається інструментом Android Debug Bridge.

# 2. МЕТОДИКА ВИРІШЕННЯ

Оскільки об’єм виконуваної роботи є достатньо громіздким, потрібно детально розпланувати та обдумати методику вирішення даної задачі. Перш за все потрібно правильно оцінити свої можливості, ресурси та час.

«PAL-System» - система, яка перш за все повинна приносити корить та бути широко використовуваною. Тому одним з важливих факторів на який потрібно звернути увагу при розробці є «user friendly interface». Тобто наш продукт повинен бути зручний та зрозумілий для користувача.

Першим кроком було придумати хороший дизайн. Оцінивши функціонал системи, я продумала яким чином було б найкраще розташувати ті чи інші компоненти системи на веб сторінці, зрозуміло побудувати панель навігації та місце відображення статистичних даних. Також треба було добре продумати можливості фільтрації, сортування великих кількостей інформації на екрані.

Для написання хорошого інтерфейсу я буду використовувати такі загально відомі інструменти як:

* мова розмітки HTML
* таблиці каскадних стилів CSS (в т.ч. Bootstrap)
* мова програмування JavaScript
* бібліотеки JQuery.JS i Chart.JS

Після обрання ідеї, дизайну та визначившись з технологіями, які найкраще підійдуть для реалізації всіх потреб нашої системи – потрібно обдумати як саме реалізувати всю логіку нашої системи. Найбільш складним для мене було придумати як саме моя вебсистема буде зчитувати вхідні смс повідомлення з телефону. Після багаточисленних запитів у пошуковій системі я зупинилась на інструменті, який найкраще підходив для моєї задачі. Android debug Bridge - це такий консольний додаток за допомого якого можна здійснювати зв'язок з телефоном з операційною системою Android.

Наступним кроком я обдумала як саме і де будуть зберігатись дані. Для цього буде доречно побудувати модель даних, продумати зв’язки між ними. Зв'язок між нашою базою даних та самою системою ми будемо здійснювати через ASP.NET CORE. Також використовуючи цей фреймворк здійснюватиметься подальша реалізація функціоналу, об’єднання front-end i back-end частин.

# 3. РЕАЛІЗАЦІЯ

## **Мова розмітки даних HTML**

HTML або Hypertext Markup Language (укр.  мова розмітки гіпертекстових документів) — стандартна [мова розмітки](https://uk.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B2%D0%B0_%D1%80%D0%BE%D0%B7%D0%BC%D1%96%D1%82%D0%BA%D0%B8_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85) для створення [веб-сторінок](https://uk.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D1%82%D0%BE%D1%80%D1%96%D0%BD%D0%BA%D0%B0) і [веб-додатків](https://uk.wikipedia.org/w/index.php?title=%D0%92%D0%B5%D0%B1-%D0%B4%D0%BE%D0%B4%D0%B0%D1%82%D0%BA%D1%96%D0%B2&action=edit&redlink=1) . З [Cascading Style Sheets](https://uk.wikipedia.org/wiki/CSS) (CSS) і [JavaScript](https://uk.wikipedia.org/wiki/JavaScript), вона утворює тріаду основних технологій для [World Wide Web](https://uk.wikipedia.org/wiki/%D0%92%D1%81%D0%B5%D1%81%D0%B2%D1%96%D1%82%D0%BD%D1%94_%D0%BF%D0%B0%D0%B2%D1%83%D1%82%D0%B8%D0%BD%D0%BD%D1%8F)[1].

Саме використовуючи HTML і відбувається перший крок у розробці веб-системи - написання розмітки основних сторінок для виводу інформації та її менеджментом. Послідовність тегів і їх атрибутів задає структуру документа, інтерпретуючи мову розмітки, браузер «розуміє», про що сторінка і як повинен відображатися її контент (текст, картинки, відео).

До переліку основних блоків інформації для менеджера входили такі як :

* Users (Користувачі)
* Workers (Працівники)
* Units (Одиниці виміру даних лічильників)
* Catalog (Каталог відправлених показників)
* Infographics (Інфографіка за певний період часу)

При написанні розмітки використовувались такі семантичні теги як <footer>, <header>, <nav>. Також теги <a>, <table>, <button> та інші.

## **3.2. Мова візуалізації CSS**

CSS або Cascading Style Sheets ([укр.](https://uk.wikipedia.org/wiki/%D0%A3%D0%BA%D1%80%D0%B0%D1%97%D0%BD%D1%81%D1%8C%D0%BA%D0%B0_%D0%BC%D0%BE%D0%B2%D0%B0) каскадні таблиці стилів) — спеціальна [мова](https://uk.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B2%D0%B0_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F), що використовується для опису зовнішнього вигляду сторінок, написаних [мовами розмітки даних](https://uk.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B2%D0%B0_%D1%80%D0%BE%D0%B7%D0%BC%D1%96%D1%82%D0%BA%D0%B8_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85). Найчастіше CSS використовують для візуальної презентації сторінок, написаних на HTML[1].

Точне розташування об'єктів на сторінці відносно один одного є, мабуть, однією з найскладніших завдань для веб-майстра. Написати HTML-код сторінки так, щоб всі графічні зображення та текстові блоки були на своїх місцях не тільки на комп'ютері веб-майстра, а й у всіх відвідувачів сайту, – справжнє мистецтво. Каскадні таблиці стилів (Cascading Style Sheets), декларовані як засіб повного контролю над HTML-розміткою, подаються до світлі розглянутої проблеми, як хороший інструмент для організації точного розташування об'єктів на сторінці. CSS дозволяють перевизначити всі властивості будь-якого тега, призначаються за умовчанням. Стає можливим виконувати вирівнювання текстового блоку щодо сторінки і інших текстових блоків.

Використовуючи CSS в стилізації системи, я зробила її вид цілісним та привабливим. Приклади стилізації можна побачити в додатках.

## **Мова програмування JavaScript**

[JavaScript](https://developer.mozilla.org/uk/docs/Glossary/JavaScript) ("JS" скорочено) це повноцінна [динамічна мова програмування](https://developer.mozilla.org/uk/docs/Glossary/%D0%94%D0%B8%D0%BD%D0%B0%D0%BC%D1%96%D1%87%D0%BD%D0%B0_%D0%BC%D0%BE%D0%B2%D0%B0_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F), яка, у застосуванні до [HTML](https://developer.mozilla.org/uk/docs/Glossary/HTML) документу, може надати динамічну інтерактивність на веб-сайтах (наприклад: ігри, реакція на натиски кнопок, введеня даних через форми, динамічна зміна стилів, анімація). Вона була винайдена Бренданом Eйхом, співзасновником проекту Mozilla, the Mozilla Foundation, та Mozilla Corporation.

JavaScript має надзвичайно багато застосувань, починаючи з малого: створити "каруселі", галереї зображень, динамічні макети сторінок, відповіді на натиски кнопок, тощо, і закінчуючи створенням ігор, 2D та 3D графіки, складних застосунків з використанням баз даних та багато іншого.

JavaScript доволі компактна та гнучка мова, що робить її одною з найбільш популярних мов програмування веб застосунків. Розробники забезпечили велике розмаїття інструментів, що доповнюють основу мови JavaScript, які відкривають величезну кількість додаткового функціоналу з мінімальними зусиллями.

Використовуючи JS я робила передачу даних зі своєї веб-сторінки в методи контролерів. Також створювала різні EventHandler-и для створення гарних анімацій переходів та ефектів.

Chart.js - популярна бібліотека з відкритим кодом, яка допомагає нам створювати дані в веб-додатках. Саме її я і вирішила виористати для побудови графіків та виведення даних лаконічно та інформаційно для користувача.

## **Фрейморк ASP.NET CORE**

ASP.NET Core є кроссплатформенним, високопродуктивним середовищем з відкритим вихідним кодом для створення сучасних хмарних додатків, підключених до Інтернету. Іншими словами, це дуже зручний та новітній фреймворк, який чудово підходить для розрабки системи «PAL-System».

ASP.NET Core надає наступні переваги:

* Єдине рішення для створення призначеного для користувача веб-інтерфейсу і веб-API.
* Надає зручні умови для тестування.
* Razor Pages робить створення сторінок простішим і ефективнішим.
* Можливість розробки і запуску в ОС Windows, macOS і Linux.
* Відкритий вихідний код і орієнтація на співтовариство.
* Інтеграція сучасних клієнтських платформ і робочих процесів розробки.
* Хмарна система конфігурації на основі середовища.
* Спрощений високопродуктивний модульний конвеєр HTTP-запитів.
* Можливість розміщення в IIS, Nginx, Apache, Docker або у власному процесі.
* Паралельне управління версіями додатка, орієнтоване на .NET Core.
* Інструментарій, що спрощує процес сучасної веб-розробки

Microsoft Visual Studio 2017 із встановленим додатком бібліотек для роботи у середовищі ASP.NET CORE має готові стартові шаблони для створення програмних веб-додатків. При створенні такого проекту формується основний «кістяк», який слугує базовим підґрунтям для створення нашої системи.

Після вибору необхідних параметрів та підключення усіх необхідних бібліотек маємо стартовий проект, який слугуватиме відправним пунктом у розробці. Таким чином уже створено усі необхідні файли конфігурації і можемо зосередитись на створені структури, дизайну та інтерфейсів користувача та адміністратора. На рисунку 4.1. зображена структура завершеного проекту.

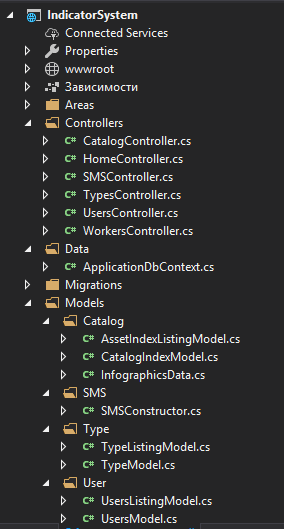
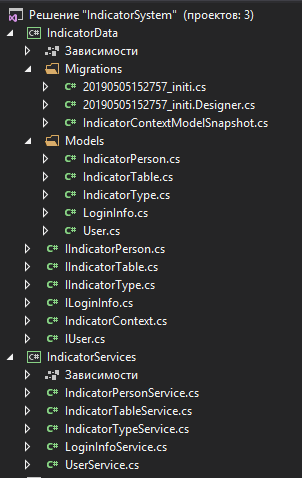
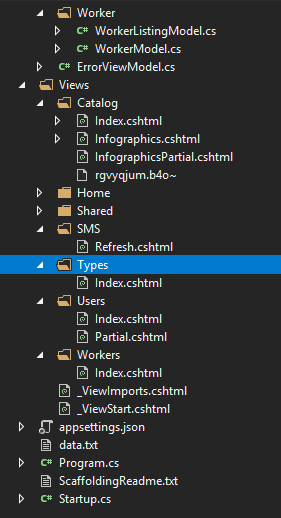


Рисунок 4.1.

## **База даних MS SQL**

Бази даних для сайтів дають змогу зберігати інформацію, що виглядає як зв'язані між собою таблиці. Саме в БД зберігаються вся необхідна та корисна інформація для функціонування сайту (клієнтські дані, прайс-лист, список товарів).

Щоб створити запит до бази даних часто використовують Structured Query Language. SQL дає змогу додавати, редагувати та видаляти інформацію, що міститься у таблицях. Під час програмування сайтів використовують різні системи управління БД. До основних СУБД, відносять:

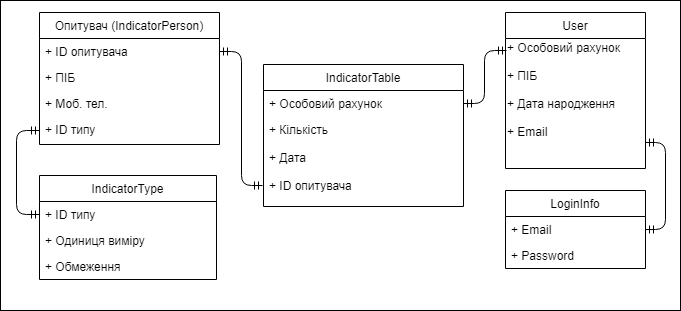
* об'єктно-реляційна система управління базами даних Oracle Database;
* вільна система управління базами даних PostgreSQL;
* система керування базами даних Microsoft SQL Сервер;
* вільна система управління базами даних MySQL;

Такі системи управління відрізняються централізованою обробкою запитів, забезпечують надійність, доступність та безпеку БД.

Під час реалізації продуту я використовувала Microsoft SQL Server , комерційну [систему керування базами даних](https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%BA%D0%B5%D1%80%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85), що розповсюджується корпорацією [Microsoft](https://uk.wikipedia.org/wiki/Microsoft). Мова, що використовується для запитів — [Transact-SQL](https://uk.wikipedia.org/wiki/Transact-SQL), створена спільно Microsoft та Sybase. Transact-SQL є реалізацією стандарту [ANSI](https://uk.wikipedia.org/wiki/%D0%90%D0%BC%D0%B5%D1%80%D0%B8%D0%BA%D0%B0%D0%BD%D1%81%D1%8C%D0%BA%D0%B8%D0%B9_%D0%BD%D0%B0%D1%86%D1%96%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D0%B8%D0%B9_%D1%96%D0%BD%D1%81%D1%82%D0%B8%D1%82%D1%83%D1%82_%D1%81%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%96%D0%B2) / [ISO](https://uk.wikipedia.org/wiki/%D0%9C%D1%96%D0%B6%D0%BD%D0%B0%D1%80%D0%BE%D0%B4%D0%BD%D0%B0_%D0%BE%D1%80%D0%B3%D0%B0%D0%BD%D1%96%D0%B7%D0%B0%D1%86%D1%96%D1%8F_%D0%B7%D1%96_%D1%81%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D0%B8%D0%B7%D0%B0%D1%86%D1%96%D1%97) щодо структурованої мови запитів [SQL](https://uk.wikipedia.org/wiki/SQL) із розширеннями. Використовується як для невеликих і середніх за розміром баз даних, так і для великих баз даних масштабу підприємства. Багато років вдало конкурує з іншими системами керування базами даних [5].

На рисунку 5.1 зображено модель даних нашої системи та звязки між її елементами. Можна побачити, що в нас 5 основних таблиць (опитувачі, іншими словами працівники, користувачі, вхідні дані користувача, типи показників і загальна таблиця з показниками).

*Рисунок 5.1*



## **Інструмент Android Debag Bridge**

ADB(Android debug Bridge - відладка Android мосту) - це інструмент, який встановлюється разом з Android-SDK і дозволяє керувати пристроєм на базі ОС Android. Іншими словами, це такий консольний додаток для персонального комп’ютера, який використовується для налагодження Android пристроїв, включаючи емулятори. Працює за принципом клієнт-сервер на всіх андроїд-пристроях, де ця функціональність не була навмисно заблокована виробником.

В свої системі я використовувала цей інструмент для зв’язку з телефоном приймачем. Саме завдяки спеціально визначених команд я доступалась до телефону, пересилала дані у форматі txt на комп’ютер та видаляла непотрібні файли.

Нижче наведено приклади таких команд.

* Емуляція натискання кнопки розблокування, іншими словами ввімкнення телефону



* Видалення файлу за вказаним шляхом



* Копіювання файлу з вказаного шляху до вказаної папки



# АПРОБАЦІЯ

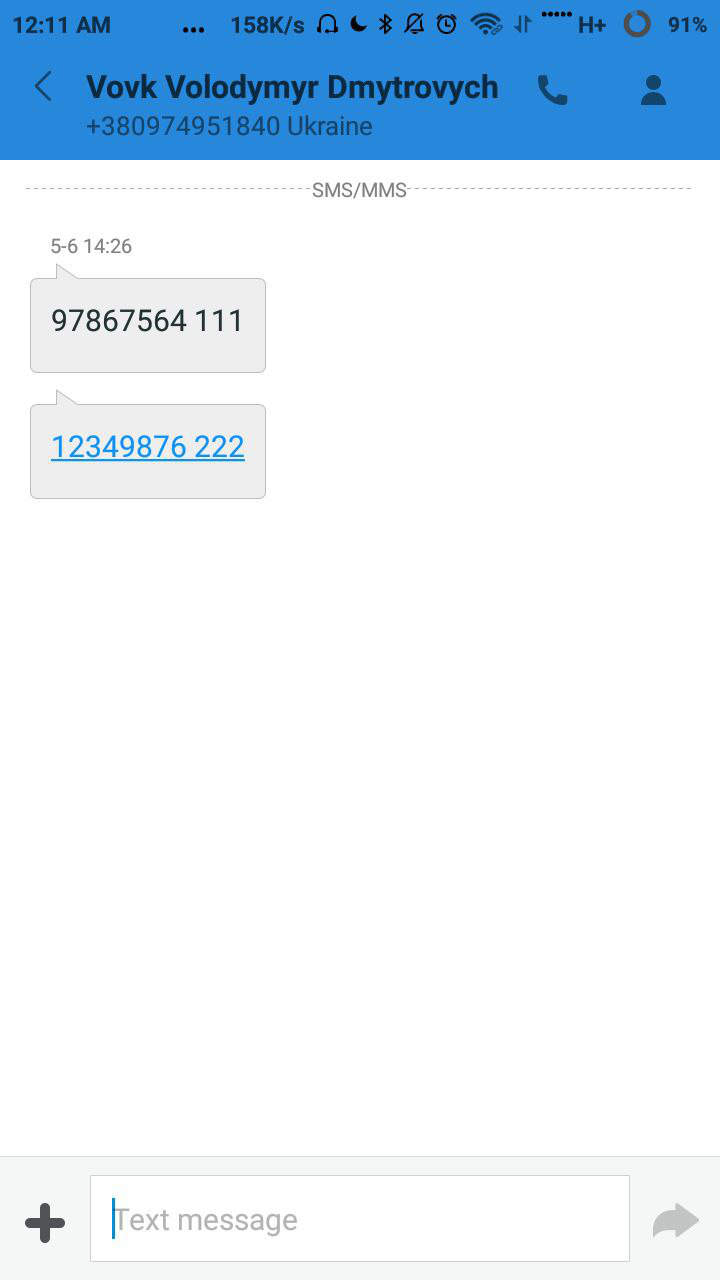
Безперечно однією з невід’ємних части написання веб-системи – є її апробація. Потрібно перевірити весь функціонал і, виявивши помилки або невідповідності, вирішити їх.

Перше з чим прийшлось стикнутись при апробації – це некоректне відображення даних. Якщо детальніше – це відображення великої кількості інформації на одній сторінці. І для того, щоб переглянути весь обсяг даних потрібно багато прокручувати сторінку, що є не дуже хорошою практикою і приносить дискомфорт для користувача.

Ось як наприклад на сторінці «Infographics» виводять всі дані про користувачів, працівників і всіх типів даних лічильників(в нашому випадку вода, газ та електроенергія). Очевидно, що така велика кількість інфлрмації не коректно розміщена на сторінці роздратує користувача. Тому було прийнято рішення розбити вивід даних на 3 блоки (Газ, Електроенергія та Вода).

Нижче буде описано покрокове проведення апробації автоматизації зчитування, обробки та виведення даних розробленої системи.

КРОК 1. Відправлення смс-повідомлень на телефон-приймач.



КРОК 2. Вхід в систему, перехід на сторінку каталогу (Catalog), оновлення сторінки (кнопка Refresh).



Очікуваний результат виконання цих двох кроків: виведення зчитаних даних у переліку відправлених показників. Як ми бачимо відправлені повідомлення були зчитані, додані в бд і виведені в вікні браузеру.

Також була проведена перевірка таких функціонал системи як

* додавання та видалення користувачів/працівників
* виведення коректної інформації
* правильний вивід опрацьованої інформації в інфографіці
* валідація даних

# ПІДСУМКИ І ВИСНОВКИ

Метою даної роботи була розробка веб-системи «PAL-System» для автоматизованого збору та адміністрування даних показників лічильників за газ, воду, та електроенергію.

В курсовій роботі було досліджено та проаналізовано найбільш важливу і актуальну інформацію щодо розробки web-додатків, а саме основні принципи створення web-додатків, їх структуру і функціональність, взаємодію основних компонентів. Також було реалізовано процес збору даних та їх обробки в спрощеному та автоматизованому вигляді. Також були розглянуті новітні та найбільш перспективні web-технології, які з успіхом вже використовуються користувачами по всьому світі.

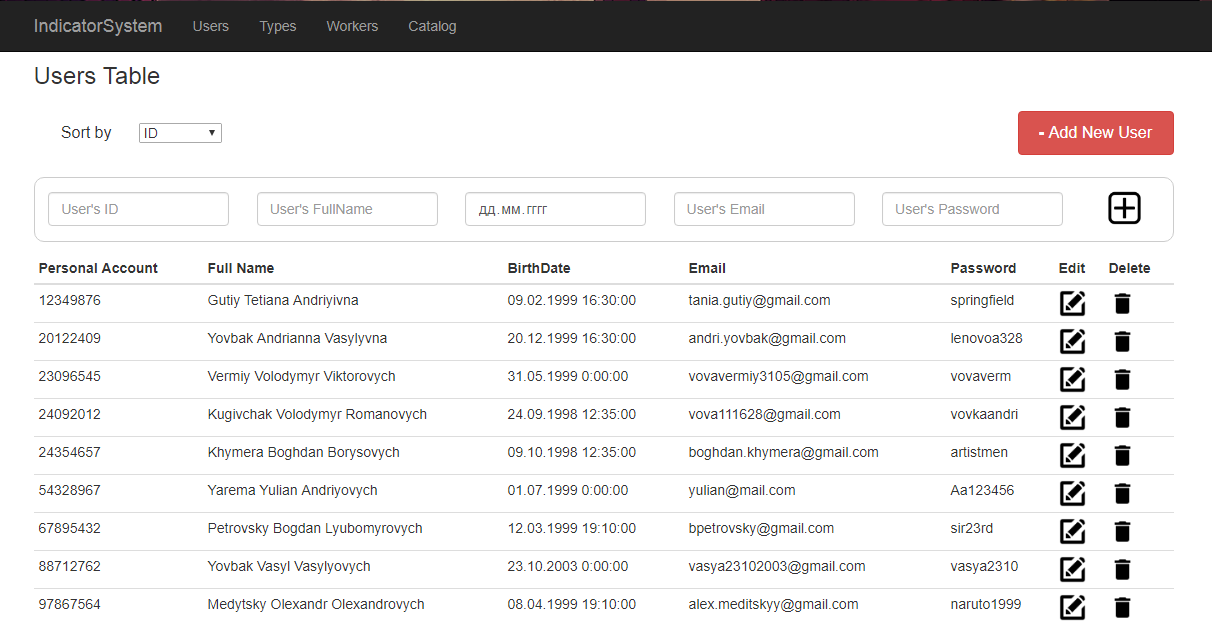
В процесі розробки web-системи адміністрування та оформлення роботи на практиці були закріплені теоретичні знання, вдосконалені навички програмування, проведено оформлення технічної документації в текстових та графічних редакторах.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

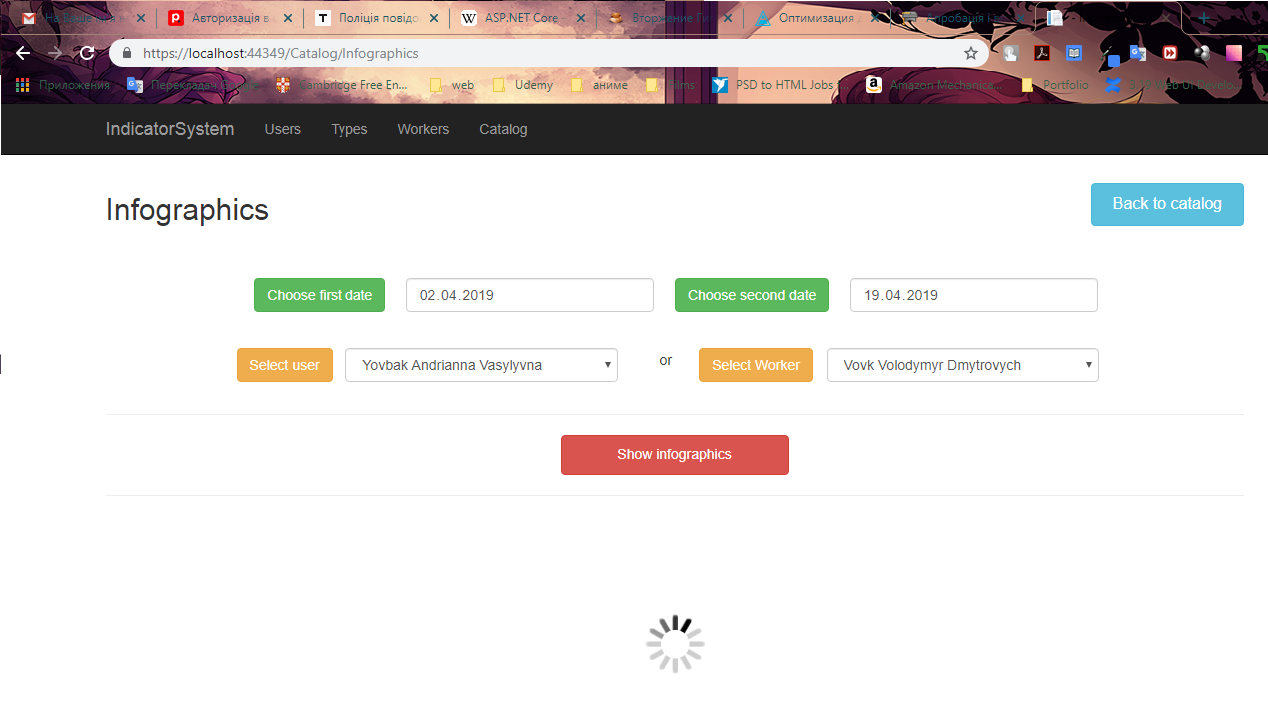
1. HTML [Електронний ресурс]: Режим доступу: https://uk.wikipedia.org/wiki/ HTML
2. CSS [Електронний ресурс]: Режим доступу: https://uk.wikipedia.org/wiki/CSS
3. JavaScript [Електронний ресурс]: Режим доступу: <https://developer.mozilla.org/uk/docs/Learn/Getting_started_with_the_web/JavaScript_basics>
4. ASP.NET Core [Електронний ресурс]: Режим доступу: <https://docs.microsoft.com/ru-ru/aspnet/core/?view=aspnetcore-2.2>
5. Microsoft SQL Server [Електронний ресурс]: Режим доступу: https://uk.wikipedia.org/wiki/Microsoft\_SQL\_Server
6. Android Debug Bridge (adb)  |  Android Developers [Електронний ресурс]: Режим доступу: <https://developer.android.com/studio/command-line/adb>
7. Chart.js | Open source HTML5 Charts for your website [Електронний ресурс]: Режим доступу: <https://www.chartjs.org/>
8. Bootstrap · The most popular HTML, CSS, and JS library in the world. [Електронний ресурс]: Режим доступу: <https://getbootstrap.com/>
9. htmlbook.ru | Для тех, кто делает сайты [Електронний ресурс]: Режим доступу: <http://htmlbook.ru/>
10. «HTML & CSS: Design and Build Web Sites» / Джон Дюкетт, 2011
11. «JavaScript and JQuery: Interactive Front-End Web Development» / Джон Дюкетт, 2013
12. «[Pro ASP.NET Core MVC 2, Seventh Edition](http://agile.tests-ua.com/Resources/bb4f7cd5-ed35-4f82-b689-72afe2f47d00)» / Freeman A. , 2017

# ДОДАТКИ

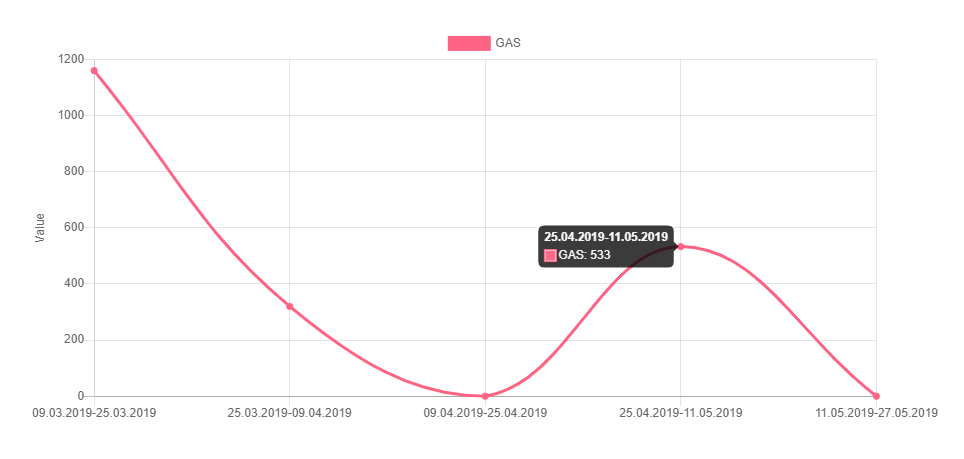
1. Сторінка «Users»



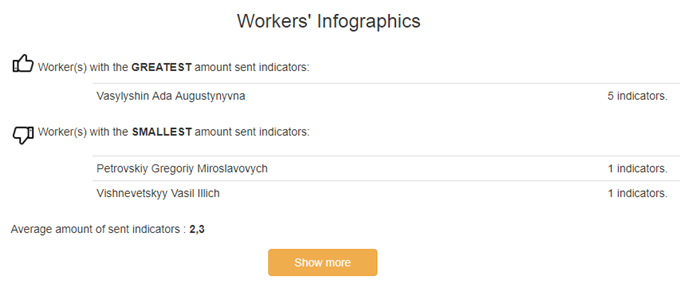
1. Сторінка «Infographics» (передзагрузка даних)



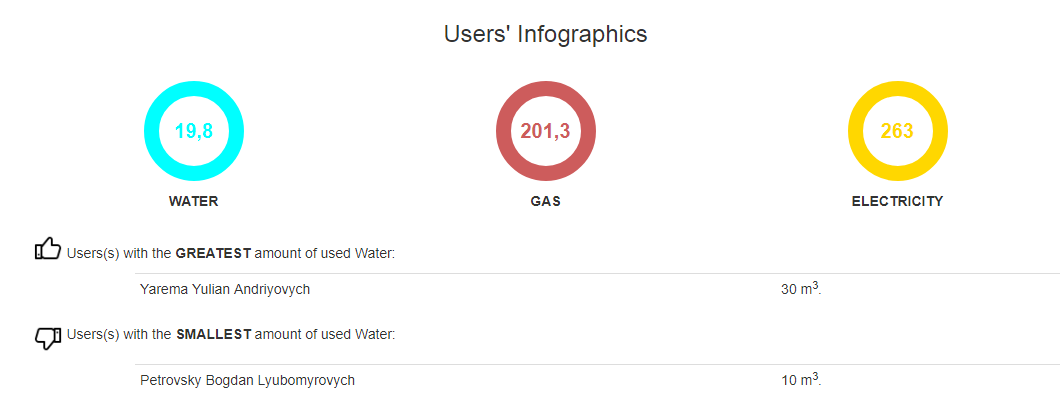
1. Сторінка «Infographics (Елементи)
   1. Приклад побудови графіку з допомогою бібліотеки Chart.JS



* 1. Статистика працівників



* 1. Статистика користувачів (по значеннях показників)



* 1. Перегляд розширеного списку даних

