

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ
КАФЕДРА ПРИКЛАДНОЇ МАТЕМАТИКИ

ПОЯСНЮВАЛЬНА ЗАПИСКА

до курсового проекту з дисципліни
Бази даних та інформаційні системи
на тему

Сайт Логістичної компанії

Виконали:

студенти групи КМ-82-1

Бурлаченко Я.Б.

Буслаєв В.О.

Курилко М.О.

Марченко В.В.

Керівник:

Ковальчук-Химюк Л.О.

Зміст

Резюме	3
Вступ.....	5
Стейкхолдери	5
Мета.....	5
Вимоги.....	7
Функціональні вимоги.....	7
Нефункціональні вимоги.....	8
Архітектурно значущі якісні атрибути.....	9
Modifiability	9
Availability.....	9
Performance	10
Представлення.....	12
Компоненти системи.....	15
База даних	15
Серверна частина	16
Клієнтська частина	16
Контейнер	17
Структура декомпозиції робіт	18
Результати перевірки відповідності вимогам	21
Додаток 1. Посилання.....	23

Резюме

Темою курсового проекту було обрано розробку логістичної служби, яка, на відміну від уже існуючих, рахуватиме вартість не за масою відправлення, а за відстанню між пунктами відправлення та призначення.

У даній системі буде реалізація створення сервісу у вигляді веб-застосунку для здійснення замовлень. А також початок роботи сервісу до новорічних свят – найоптимальніший момент виходу на ринок коли карантин скоріш за все буде ще актуальним, а попит на відправку тільки збільшиться. Усе це призведе до досягнення глобальної цілі – створення альтернативи існуючим компаніям у ніші бюджетних логістичних перевезень. Це дозволить користувачам зекономити на відправленні.

Для розробки обраної системи стали ключовими три підстави. Перша: попит на логістичні послуги, який створили одночасно самоізоляція та необхідність часто обмінюватися негабаритними речами та документами. Друга – це існування ніші, яку доцільно було б зайняти, адже попит на перевезення перевищує існуючі на ринку пропозиції. І третьою стала невелика кількість компаній які б орієнтувалися на низовий сегмент ринку, враховуючи економічну ситуацію в країні.

Дане програмне забезпечення розробляється згідно з навчальним планом підготовки бакалаврів.

Яких результатів вдалось досягнути?

В результаті виконання проекту було розроблену інформаційну веб систему для організації логістичного процесу.

Створено сайт, де користувачі можуть зареєструватися для того, щоб користуватися послугами компанії, автентифікуватися та почати обмінюватися відправленнями з іншими користувачами, а в особистому кабінеті створено можливість вивантаження даних у вигляді файлу.

Робітники компанії є окремою групою користувачів у яких є можливість додавати нові відділення та перегляди замовлення які проходять через відділення, в якому він працює.

Для реалізації процесу доставки було введено обов'язкового пакування усіх відправлень у фірмові упаковки, які займають відомий об'єм у машині. Тобто відомо, скільки пакувань яких розмірів може бути доставлено за один раз. Для реалізації алгоритму доставки було взято за основу задачу про рюкзак, яку було розв'язано методами динамічного програмування за наступним планом.

Створення словників, де ключі – номери відправлень, а значення представлено кортежем з об'єму відправлення та його цінність. Цінність вираховується як різниця загальної кількості не відправлених посилок та порядкового номера. Тобто більш цінними будуть відправлення, які довше не доставлялися.

Вступ

Стейкхолдери

Business Consultant Team

Concerns: допомога у створенні системи для досягнення поставлених цілей.

Views: система, що дозволить досягти поставлених бізнес цілей.

Project Manager:

Concerns: реалізація проекту в строк

Views: вчасно зроблений проект, в якому виконані всі технічні та функціональні вимоги.

Design Engineer:

Concerns: створення WEB-UI, дизайн документів користувачів (виписки, квитанції, накладні)

Views: дизайн, який буде простим і зручним для користувачів.

DB Engineer:

Concerns: розробка архітектури БД для збереження даних користувачів та відправлень.

Views: розгорнута база даних яка задовольняє функціональним та технічним вимогам.

Technical Writer:

Concerns: створення документації системи

Views: шаблон пояснювальної записки

Мета

Business drivers

- Самоізоляція та необхідність часто обмінюватися негабаритними речами та документами – створили *неабиякий попит на логістичні послуги*.
- Попит на перевезення перевищує існуючі на ринку пропозиції - *утворюється ніша, яку доцільно зайняти*.
- Невелика кількість компаній які б орієнтувалися на низовий сегмент ринку, враховуючи економічну ситуацію в країні.

Business goals

- Створити альтернативу існуючим компаніям в ніші бюджетних логістичних перевезень – дозволить користувачам зекономити на відправленні

Business objectives

- *Створити сервіс у вигляді веб-застосунку для здійснення замовлень.*
- *Почати роботу сервісу до новорічних свят – найоптимальніший момент виходу на ринок коли карантин скоріш за все буде ще актуальним, а попит на відправку тільки збільшиться.*

Вимоги

Функціональні вимоги

Для реалізації поставлених цілей проекту програмне забезпечення має задовольняти наступним умовам:

- Система має дозволяти користувачам реєструватися за умови заповнення e-mail адреси та обраного паролю.
- Система має дозволяти користувачам автентифікуватися за допомогою e-mail адреси та обраного паролю.
- В системі має бути реалізована сторінка «Особистого кабінету користувача» на якій буде змога переглядати відправлення в яких користувач є Відправником або Одержувачем, перевірити статус відправлення та отримати накладну з інформацією про відправлення.
- Система має дозволяти створювати відправлення при отриманні наступних даних:
 - e-mail відпраника та отримувача;
 - номер відділення отримання;
 - розмір пакування відправлення;
- Після формування замовленню має бути присвоєно індивідуальний номер та прорахована вартість послуги, яка вираховується з відстані та розміру відправлення;
- Варіанти розміру пакування (S, M, L, XL) та номери відділень з адресами опис яких має знаходитися на окремих сторінках застосунку.
- Створити сторінки обслуговування для працівників відділень:
 - Сторінка додавання нового відділення;
 - Сторінка перегляду усіх надходжень до конкретного відділення.

Нефункціональні вимоги

1. Система має дозволяти змінювати дані користувача такі як пароль чи логін без втрати даних - *Quality attribute requirements*.
2. Система має бути доступною різним користувачам для перегляду або внесення інформації - *Quality attribute requirements*.
3. Реалізовано дві чи більше ролей з можливістю створення користувачів з різними ролями - *Обмеження*.
4. Паролі користувачів мають бути захешіровані - *Обмеження*.
5. Система має валідувати принаймні 60% інформації - *Обмеження*.
6. В системі має бути використано REST API 2 рівня - *Обмеження*.

Архітектурно значущі якісні атрибути

Modifiability

Сценарій	Значення
Source	Кінцевий користувач
Stimulus	Хочемо змінити особисті дані користувача
Artifact	Платформа
Environment	Під час виконання, час компіляції, час складання, час проектування
Response	Робити зміни без впливу на інші функціональності
Response Measure	Повідомлення про кількість змінних рядків

Цей атрибут було обрано, щоб розширити можливості користувача. Наприклад, йому вже не зручно відвідувати відділення яке обрав при реєстрації і тому він може змінити його.

Availability

Сценарій	Значення
Source	Всередині системи
Stimulus	Несправність: пропуск, аварія, час, відповідь

Artifact	Процесори системи, канали зв'язку, постійне сховище, процеси
Environment	Будь-яка операція
Response	Система повинна виявляти подію і може бути недоступною протягом заздалегідь визначеного інтервалу, де інтервал залежить від критичності системи, після чого продовжувати працювати у звичайному режимі
Response Measure	Інтервал часу, коли система повинна бути доступною

Система має бути доступною. Тому при проведенні технічного .

Performance

Сценарій	Значення
Source	Надсилання інформації
Stimulus	Поява періодичних подій
Artifact	Система
Environment	Нормальний режим
Response	Обробка подразників
Response Measure	Затримка, кінцевий термін, пропускну здатність, швидкість пропусків, втрата даних

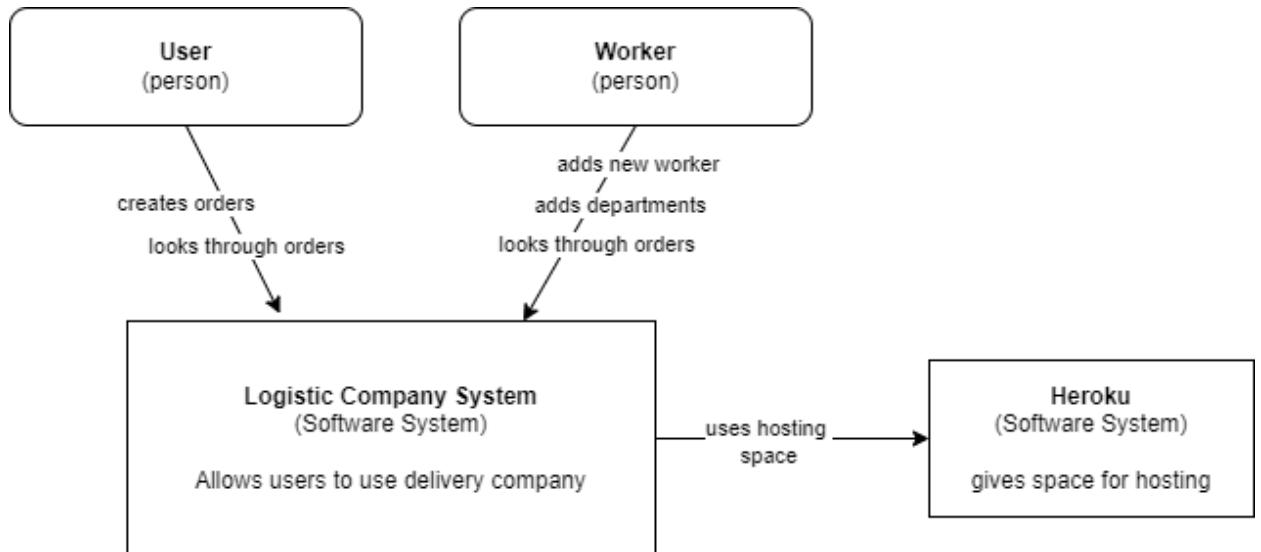
Система має бути інформативною і при виникненні проблем користувач має бути проінформованим про них, або бачити результат своїх дій.

Представлення

В ході розробки даного проекту для представлення архітектури було обрано модель під назвою С4.

Рівень 1

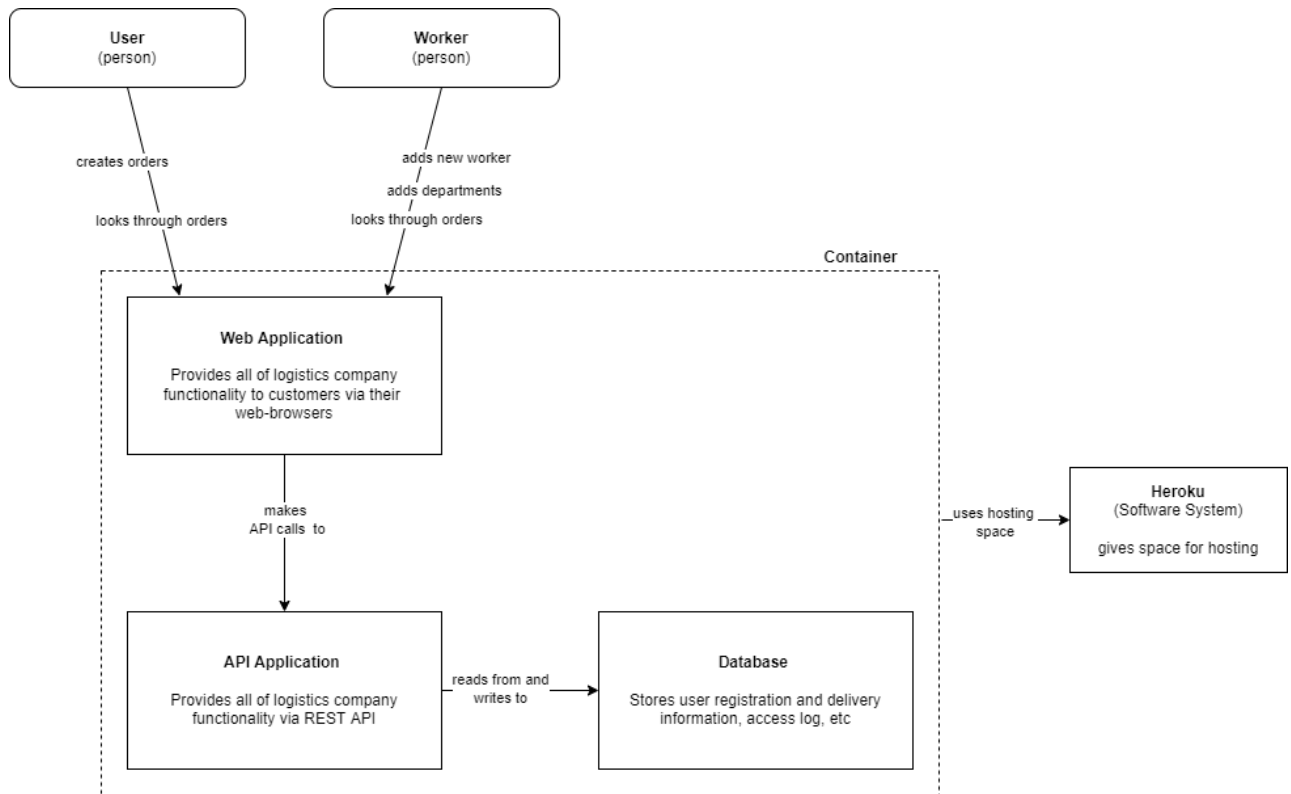
Context diagram



Показує взаємодію системи з навколишнім середовищем.

Рівень 2

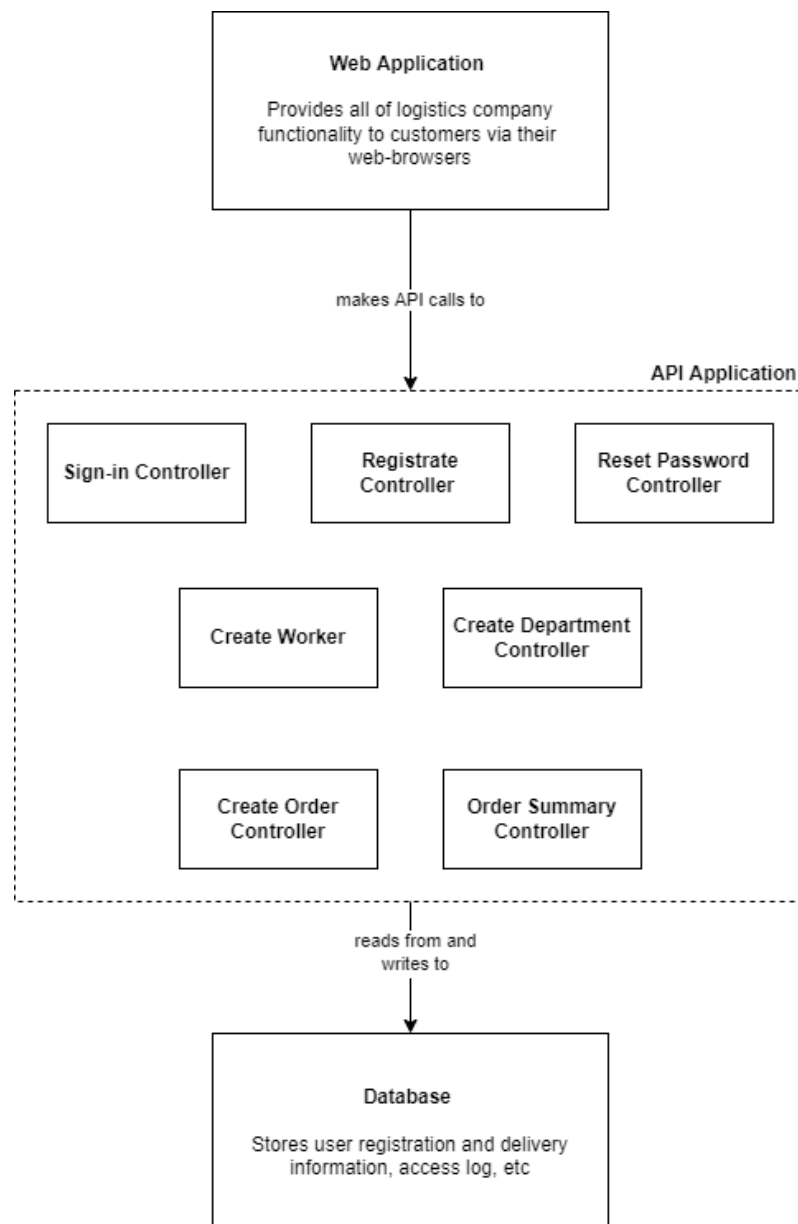
Container Diagram



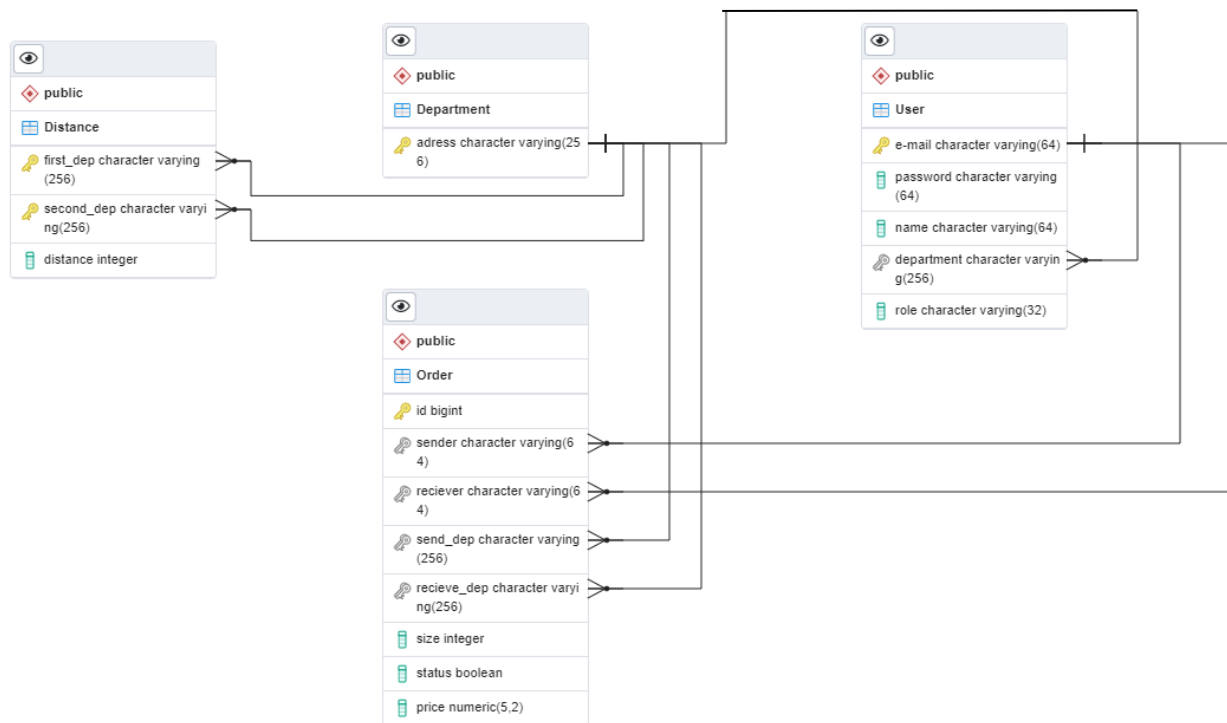
Показує взаємодію компонентів всередині системи.

Рівень 3

Component Diagram



Показує складові компонента серверної частини.



Показує взаємодію складових бази даних.

Компоненти системи

Система складається з чотирьох компонентів: бази даних, серверної частини, клієнтської частини та контейнеру.

База даних

Опис	Збереження інформації
Технологічний стек	PostgreSQL
Пов'язані компоненти	Серверна частина, клієнтська частина

Покриті функціональні вимоги	Реєстрація , Створення замовлення
------------------------------	---

Серверна частина

Опис	Обробка запитів користувача
Технологічний стек	SQLAlchemy, Flask, psycopg2
Пов'язані компоненти	Клієнтська частина, база даних
Покриті функціональні вимоги	Реєстрація , Автентифікація , Особистий кабінет , Створення замовлення , Формування замовлення , Інформація про компанію , Візуальне представлення

Клієнтська частина

Опис	Взаємодія клієнта із веб-застосунком
Технологічний стек	html, CSS
Пов'язані компоненти	Серверна частина, база даних

Покриті функціональні вимоги	Особистий кабінет , Інформація про компанію , Візуальне представлення
------------------------------	---

Контейнер

Опис	Забезпечення ізолюваного середовища для проекту
Технологічний стек	Docker Container
Пов'язані компоненти	Серверна частина, клієнтська частина, база даних
Покриті функціональні вимоги	Створення ізолюваного середовища для правильного формування проекту та налаштування взаємодії його компонентів

Структура декомпозиції робіт

Завдання	Відповідальний	Початкова оцінка часу (год)	Реально витрачений час (год)
Дизайн веб-застосунку			
• Сторінка Особистого кабінету Користувача	Курилко М.О.	2	2
• Сторінка Реєстрації		3	5
• Сторінка автентифікації		3	3
• Сторінка Особистого кабінету працівника компанії		2	2
• Сторінка формування відправлення		3	3
• Головна сторінка застосунку		2	1
• Сторінка перегляду та додавання відділень		2	3
• Сторінка додавання працівника		3	3
Дизайн БД			
• Таблиця Відправлень	Буслаєв В.О.	0.5	0.5
• Таблиця Відділень		0.5	0.5
• Таблиця Користувачів		0.5	0.5
• Написати запити для заповнення таблиць		0.5	0.5
• Створити Логічну та Реляційну діаграму БД		0.5	1

Функціональність			
• Створення Docker Container	Буслаєв В.О.	2	3
• Деплой застосунку на Heroku		1	4
• Реалізувати реєстрацію користувачів	Марченко В.В.	4	5
• Реалізувати процедуру автентифікації		4	6
• Реалізувати відображення інформації в особистому кабінеті		3	3
• Реалізувати збереження інформації про відправлення		2	1,5
• Реалізувати обрахунок вартості послуги		0,5	0,1
• Розгорнути систему на Heroku		2	4
• Реалізувати вивантаження даних у вигляді документу для завантаження		2	5
• Реалізувати збереження і відображення Розмірів відправлень		2	1
Тестування			
• Перевірити дизайн	Бурлаченко Я.Б.	1	1
• Перевірити коректність відображення даних з БД		1.5	2

<ul style="list-style-type: none"> Перевірити коректність роботи функціональної частини 		1.5	2
<ul style="list-style-type: none"> Перевірити відповідність виконання умов ТЗ 		1	1
Документація			
<ul style="list-style-type: none"> Написати документацію у вигляді пояснювальної записки 		11	13.5

Результати перевірки відповідності вимогам

У результаті перевірки відповідності вимогам з'ясовано:

- система дозволяє користувачам реєструватися за умови заповнення e-mail адреси, обраного паролю, імені та відділення, з якого користувач планує робити відправлення;
- система дозволяє користувачам автентифікуватися за допомогою e-mail адреси та обраного паролю;
- в системі реалізована сторінка «Особистого кабінету користувача» на якій є змога переглядати відправлення в яких користувач є Відправником або Одержувачем, перевірити статус відправлення та отримати накладну з інформацією про відправлення;
- система дозволяє створювати відправлення при отриманні наступних даних: e-mail отримувача, відділення отримання, розмір пакування відправлення;
- після формування замовленню присвоєно індивідуальний номер та прорахована вартість послуги, яка вираховується з відстані та розміру відправлення та виробничих коефіцієнтів;
- у системі не реалізовано окремі сторінки перегляду розмірів пакування (S, M, L, XL) та відділень з адресами, проте усі відділення та розміри пакування, доступні користувачам на формах створення відправлення та реєстрації;
- створено сторінки обслуговування для працівників відділень: додавання нового відділення та перегляду усіх надходжень до конкретного відділення;
- система має дозволяти змінювати дані користувача, в тому числі, пароль, однак логін змінювати не можна.
- система є доступною різним користувачам для перегляду, та, на жаль, внесення інформації доступне неодноразово.

- реалізовано дві ролі користувачів, при чому робітники компанії можуть додавати нових робітників до системи, але першого робітника має додати адміністратор бази даних.
- паролі користувачів захешировано.
- валідація зроблена методами html, решта не валідується за відсутністю необхідності.

Додаток 1. Посилання

- Посилання на GitHub: https://github.com/mrvladkino27/DBIS_Logistic
- Посилання на WEB-систему: <http://logistics-cumpany.herokuapp.com/>