A logo on a grey background

Description automatically generated

**NextPointLogix**

**Технічне завдання (ТЗ)**

**1. Вступ**

**1.1. Мета документа**  
Цей документ описує технічні вимоги до системи автоматизації роботи транспортно-логістичного відділу, яка буде розроблена як проект NextPointLogix. Мета системи - забезпечити ефективну організацію перевезення працівників на завод і назад, включаючи планування маршрутів, управління водіями, облік витрат пального і фінансові розрахунки.

**1.2. Мета проекту**  
Розробити інтегровану систему, яка включає мобільні додатки для працівників і водіїв, веб-додатки для логіст-оператора, фінансового менеджера і адміністратора системи, що забезпечить автоматизацію процесів, пов'язаних з транспортуванням працівників.

**2. Загальний опис**

**2.1. Опис системи**  
NextPointLogix є комплексною системою для автоматизації перевезень працівників, яка включає мобільні додатки для працівників та водіїв, а також веб-додатки для логіст-оператора, фінансового менеджера і адміністратора. Система дозволяє працівникам відмічати дні, коли їм потрібен транспорт, водіям - отримувати маршрути та координувати доставку, а логіст-операторам та фінансовим менеджерам - управляти маршрутами, обліковувати витрати та формувати звіти.

**2.2. Основні компоненти**

* Мобільний додаток для працівників
* Мобільний додаток для водіїв
* Веб-додаток для логіст-оператора
* Веб-інтерфейс для фінансового менеджера
* Веб-інтерфейс для адміністратора системи

**2.3. Технічний стек**

* **Front-end**: React Native (мобільні додатки), MS Power Apps (веб-додатки)
* **Back-end**: Python
* **База даних**: MySQL
* **Хостинг**: AWS
* **CI/CD**: GitHub

**3. Функціональні вимоги**

**3.1. Мобільний додаток для працівників**

* Реєстрація та авторизація
* Календар для відмітки днів потреби в транспорті
* Перегляд включення в маршрут та орієнтовний час прибуття

**3.2. Мобільний додаток для водіїв**

* Отримання та перегляд плану маршруту
* Перегляд точок посадки та висадки
* Оновлення статусу виконання маршруту

**3.3. Веб-додаток для логіст-оператора**

* Формування і редагування маршрутів
* Перегляд статистики та витрат
* Розподіл маршрутів між водіями

**3.4. Веб-інтерфейс для фінансового менеджера**

* Перегляд витрат на паливо
* Нарахування виплат водіям
* Формування звітів

**3.5. Веб-інтерфейс для адміністратора системи**

* Управління користувачами та ролями
* Налаштування доступу

**4. Нефункціональні вимоги**

**4.1. Платформа**

* Мобільні додатки для Android
* Веб-додатки для основних сучасних браузерів

**4.2. Зручність користування**

* Інтуїтивно зрозумілий інтерфейс для всіх користувачів

**4.3. Мобільний додаток для працівників і водіїв**

* Додатки повинні бути доступні на Google Play Store

**5. Нефункціональні вимоги**

**5.1. Продуктивність**

* Швидкий відгук на запити користувачів
* Здатність обробляти високі навантаження

**5.2. Безпека**

* Захист персональних даних
* Аутентифікація та авторизація користувачів

**5.3. Надійність**

* Висока доступність і відмовостійкість

**5.4. Масштабованість**

* Можливість масштабування системи при зростанні навантаження

**6. Інтеграційні вимоги**

**6.1. Інтеграція з бухгалтерськими програмами**

* Формування інвойсів для оплати послуг водіїв і пального
* Інтеграція з 1С або подібними системами

**6.2. Інші інтеграції**

* Інтеграція з платіжними шлюзами для обробки транзакцій (не обов'язково для першої версії)

**7. Вимоги до даних**

**7.1. Типи даних**

* Персональні дані користувачів
* Дані про маршрути і витрати

**7.2. Формат даних**

* Структуровані дані для бази даних MySQL
* Формати для обміну даними з бухгалтерськими системами

**8. Вимоги до розгортання та інфраструктури**

**8.1. Хостинг**

* AWS для хостингу серверів і бази даних

**8.2. Резервне копіювання та відновлення**

* Автоматичне резервне копіювання бази даних і файлів

**8.3. Безпека**

* Шифрування даних і управління доступом

**9. Тестові вимоги**

**9.1. Типи тестування**

* Функціональне, інтеграційне і навантажувальне тестування

**9.2. Критерії прийняття**

* Відповідність функціональним вимогам
* Успішне проходження всіх тестів

Опис основних компонентів

Інтерфейси користувачів

1. Мобільний додаток для працівників

**1. Екран авторизації:**

* **Логін та пароль:** Стандартний варіант авторизації.
* **Відновлення пароля:** Форма для відновлення доступу до облікового запису.
* **Пам’ятати мене:** Опція для автоматичного входу.

**2. Головний екран:**

* **Швидкий доступ:** До основних функцій (календар, повідомлення, профіль).
* **Екстренне повідомлення оператору**: Запит до оператора на скасування поїздки, що уже зафіксована.
* **Візуалізація останніх подій:** Короткий огляд найближчих поїздок або змін у розкладі.

**3. Календар:**

* **Тижневий/місячний перегляд:** Можливість перемикання між різними видами календаря. Відобразити статус замовлення в плані редагування (можна редагувати, замовлення зафіксоване і недоступне для редагування, поїздка відбулась). Відобразити дні, на які посадка не замовлена.
* **Фільтри:** За типом події (робота, відпустка), датою, маршрутом.
* **Додавання подій:** Інтуїтивний інтерфейс для створення нових подій(відпуска, хвороба, відмова від замовлення). Автоматичне заповнення – поставити замовлення на всі робочі дні у наступному місяці, обовязкове ручне підтвердження. (можливо потрібен ще один екран). Групове скасування замовлень (хвороба, відпустка).
* **Редагування подій:** Можливість змінювати дату, час, тип події. Запит до оператора на скасування поїздки, що уже зафіксована.
* **Нагадування:** Пуш-сповіщення про наближення події. Пуш повідомлення за 15 хвилин до прибуття автобуса в точку посадки.

**4. Екран детальної інформації про день:**

* **Деталі маршруту:** Точка посадки, точка висадки, час у дорозі, номер автобуса.
* **Карта маршруту:** Візуальне представлення маршруту на карті.
* **Можливість поділитися інформацією:** Наприклад, відправити деталі маршруту колегам. Можливість зафіксувати координату посадки і відправити оператору.
* **Зворотний зв'язок:** Форма для зворотного зв'язку щодо якості обслуговування.

**5. Екран повідомлень:**

* **Різні типи повідомлень:** Системні повідомлення, повідомлення про зміни в розкладі, персональні повідомлення.
* **Архів повідомлень:** Збереження історії повідомлень для подальшого перегляду.

1. Мобільний додаток для водіїв
   1. **Екран авторизації:**
      * **Логін та пароль:** Стандартний варіант авторизації.
      * **Відновлення пароля:** Форма для відновлення доступу до облікового запису.
      * **Пам’ятати мене:** Опція для автоматичного входу.
   2. **Головний екран:**
      * **Швидкий доступ:** Іконки для переходу до календаря, повідомлень, профілю.
      * **Екстренне повідомлення оператору:** Кнопка для відправки запиту.
      * **Візуалізація останніх подій:** Список найближчих поїздок або змін у розкладі.
      * **Відображення статусу:** Я сьогодні на маршруті/Я сьогодні не на маршруті. Номер маршруту (коротка назва). Час початку (забів на першій точці посадки), Час прибуття на кінцеву точку.
      * **Кнопка відправки поточних координат** – «де я?». Вибір – кому відправити (оператору чи пасажирам, які перераховані у поточному маршруті).
   3. **Календар:**
      * **Перегляд:** Перемикачі для вибору тижневого або місячного перегляду.
      * **Фільтри:** Меню для фільтрації подій за типом, датою, маршрутом.
      * **Додавання подій:** Форма для створення нових подій з опцією автозаповнення.
      * **Редагування подій:** Інтерфейс для зміни деталей подій.
      * **Нагадування:** Налаштування для пуш-сповіщень про події.
   4. **Екран детальної інформації про день:**
      * **Деталі маршруту:** Відображення інформації про маршрут. Точки посадки пасажирів (адреса), ПІБ пасажирів, орієнтовний час посадки, контактні телефони. Можливість відмітити, що пасажир не прибув на точку посадки. Повинен бути видимий статус – яка точка посадки пройдена, яка наступна.
      * **Карта маршруту:** Інтерактивна карта з маршрутом. Можливість переходу в Google maps і навігація в реальному часі.
      * **Поділитися інформацією:** Кнопка для відправки деталей маршруту. Кнопка відправки поточних координат – «де я?»
      * **Дані про заправку палива:** Тип палива, кількість літрів, ціна, вартість. Назва та адреса АЗС, координати, номер автомобіля, ПІБ водія, дата, час, тип пального, ціна, кількість літрів, сума
      * **Зворотний зв’язок:** Форма для надсилання відгуків.
   5. **Екран заправки палива:**
      * Назва та адреса АЗС, координати, номер автомобіля, ПІБ водія, дата, час, тип пального, ціна, кількість літрів, сума.
      * Кнопка відправки даних до оператора.
      * Форма введення показів пробігу на спідометрі. Розрахунок пройденої відстані від попередньої заправки. Розрахунок витрати палива на 1 кілометр пройденого шляху.
   6. **Екран повідомлень:**
      * **Типи повідомлень:** Розділи для різних категорій повідомлень.
      * Кнопка відправки поточних координат – «де я?» (вибір : оператору та/або пасажирам, переліченим у маршруті).
2. Веб-додаток для логіст-оператора
3. **Екран авторизації:**
   * **Логін та пароль:** Стандартний варіант авторизації.
   * **Відновлення пароля:** Форма для відновлення доступу до облікового запису.
   * **Пам’ятати мене:** Опція для автоматичного входу.
4. **Головний екран:**
   * **Швидкий доступ:** Іконки для переходу до календаря, повідомлень, профілю.
   * **Екстренне повідомлення водіям:** Кнопка для відправки запиту або повідомлення водіям.
   * **Візуалізація останніх подій:** Список останніх подій, таких як зміни у маршрутах або повідомлення від водіїв.
   * **Відображення статусу маршрутів:** Список активних маршрутів із зазначенням стану виконання (наприклад, "В процесі", "Завершено").
   * **Кнопка відправки поточних координат водіїв:** Можливість запиту координат водіїв у реальному часі.
5. **Карта маршрутів:**
   * **Відображення заявок:** Карта з відображенням всіх пасажирів, які зробили заявку на вказану дату.
   * **Оптимальна кількість транспорту:** Інтерфейс, що показує оптимальну кількість транспорту, яку потрібно задіяти для виконання всіх посадок, розраховану АІ.
   * **Прийняття маршрутів:** Можливість прийняти оптимальні маршрути або ввести обмеження (кількість автомобілів, кількість точок на одному маршруті, максимальний час в дорозі).
   * **Параметри оптимізації:** Встановлення параметрів для оптимізації (найменша кількість транспорту, найменша витрата палива, найкоротша тривалість маршрутів).
   * **Ручне складання маршрутів:** Можливість ігнорувати пропозиції АІ і скласти маршрути вручну.
   * **Копіювання маршрутів:** Можливість скопіювати маршрути з попередніх дат або перенести всі маршрути за обраний період.
6. **Календар:**
   * **Перегляд:** Перемикачі для вибору тижневого або місячного перегляду.
   * **Фільтри:** Меню для фільтрації подій за типом, датою, маршрутом.
   * **Додавання подій:** Форма для створення нових подій з опцією автозаповнення.
   * **Редагування подій:** Інтерфейс для зміни деталей подій.
   * **Нагадування:** Налаштування для пуш-сповіщень про події.
7. **Екран детальної інформації про маршрут:**
   * **Деталі маршруту:** Відображення інформації про маршрут. Точки посадки пасажирів (адреса), ПІБ пасажирів, орієнтовний час посадки, контактні телефони.
   * **Карта маршруту:** Інтерактивна карта з маршрутом. Можливість переходу в Google Maps для навігації в реальному часі.
   * **Поділитися інформацією:** Кнопка для відправки деталей маршруту водіям або пасажирам.
   * **Дані про заправку палива:** Інформація про останні заправки: тип палива, кількість літрів, ціна, вартість, адреса АЗС.
   * **Зворотний зв’язок:** Форма для отримання відгуків від водіїв та пасажирів.
8. **Екран управління паливом:**
   * **Назва та адреса АЗС, координати, номер автомобіля, ПІБ водія, дата, час, тип пального, ціна, кількість літрів, сума.**
   * **Кнопка відправки даних:** Можливість відправки даних до фінансового менеджера.
   * **Форма введення показів пробігу:** Введення пробігу на спідометрі до початку маршруту та в кінці маршруту, розрахований показник пробігу на основі спідометра, розрахована довжина маршруту на основі карти.
   * **Прийняття показників:** Можливість прийняти за замовчуванням розрахований по карті пробіг або прийняти вручну показ по спідометру.
   * **Внесення коментаря:** Поле для введення коментаря (наприклад, "виникла необхідність забрати додаткового пасажира").
9. **Екран реєстрації наявних автобусів і доступних водіїв:**
   * **Доступні водії та автобуси:** Відображення списку доступних водіїв і автобусів на конкретний день.
   * **Календар доступності:** Календар з відображенням доступності водіїв і автобусів на різні дні.
10. **Екран повідомлень:**
    * **Типи повідомлень:** Розділи для різних категорій повідомлень (наприклад, від водіїв, від пасажирів, системні повідомлення).
    * **Кнопка відправки координат водіїв:** Можливість запиту поточних координат водіїв у реальному часі.
    * **Відправка повідомлень водіям:** Можливість відправки повідомлень водіям з оновленнями або інструкціями.

**Додаткові функції:**

* **Алгоритм складання маршрутів:** Нейромережа автоматично складає маршрути на основі даних від працівників, з можливістю ручного коригування оператором.
* **Розрахунок витрат палива:** Нейромережа розраховує орієнтовну витрату палива на основі маршрутів.
* **Розрахунок часу:** Нейромережа розраховує орієнтовний час посадки пасажирів та час у дорозі.

Для головного екрану веб-додатку NextPointLogix, враховуючи логотип і тему, що відображають технологічність, ефективність і сучасний стиль, я пропоную наступний дизайн:

**Дизайн Головного Екрану**

**1. Верхня панель навігації (Header)**

* **Логотип NextPointLogix**: Розміщений у верхньому лівому куті. Логотип може бути інтерактивним і виконувати функцію кнопки повернення на головний екран.
* **Основне меню**:
  + **Календар**: Іконка або текстове посилання для переходу до календаря з подіями та маршрутами.
  + **Повідомлення**: Іконка дзвіночка або конверта для перегляду нових повідомлень і сповіщень.
  + **Профіль користувача**: Іконка профілю, яка відкриває меню з налаштуваннями облікового запису, зміною пароля та виходом з системи.

**2. Головний контент (Main Content)**

* **Швидкий доступ**:
  + Розміщення трьох великих кнопок (іконок) для швидкого доступу:
    - **Календар** (іконка календаря): Веде до детального перегляду подій та маршрутів.
    - **Повідомлення** (іконка дзвіночка): Відкриває екран повідомлень.
    - **Профіль** (іконка профілю): Відкриває екран налаштувань користувача.
* **Екстренне повідомлення водіям**:
  + Велика помітна кнопка з текстом "Екстренне повідомлення водіям" розташована посередині. Використовується для надсилання термінових повідомлень водіям.
* **Список останніх подій**:
  + Під швидким доступом розміщений горизонтальний або вертикальний список з останніми подіями (зміни в маршрутах, нові повідомлення від водіїв тощо).
  + Кожен елемент списку містить іконку, короткий опис події та час.
* **Статус маршрутів**:
  + Під списком подій розміщений блок зі статусом активних маршрутів. Кожен маршрут має іконку статусу ("В процесі", "Завершено", "Заплановано") та короткий опис.

**3. Бічна панель (Sidebar)**

* **Меню навігації**:
  + Список основних розділів додатку:
    - **Карта маршрутів**
    - **Управління паливом**
    - **Заявки на перевезення**
    - **Реєстрація автобусів і водіїв**
    - **Налаштування**
* **Кнопка запиту координат**:
  + У нижній частині бічної панелі велика кнопка для запиту поточних координат водіїв у реальному часі.

**4. Футер (Footer)**

* **Інформаційний текст**:
  + Коротке повідомлення про права захищеність, посилання на політику конфіденційності та технічну підтримку.
* **Іконки соціальних мереж** (якщо доречно): Відображаються іконки з посиланнями на соціальні мережі або додаткові ресурси компанії.

**Тема та стиль:**

* **Кольори**: Використання холодних відтінків синього та зеленого, які підкреслюють технологічність і надійність (наприклад, градієнт від темно-синього до світло-зеленого).
* **Шрифти**: Модерновий, чистий шрифт для тексту. Можна використовувати Open Sans або Roboto для заголовків та основного тексту.
* **Градієнти та тіні**: Легка тінь під елементами та плавні градієнти для кнопок надають сучасного вигляду.
* **Анімації**: М'які анімації при наведенні на кнопки, плавний перехід між екранами.

1. **Веб-інтерфейс для фінансового менеджера**
2. **Екран авторизації:**
   * **Логін та пароль:** Стандартний варіант авторизації.
   * **Відновлення пароля:** Форма для відновлення доступу до облікового запису.
   * **Пам’ятати мене:** Опція для автоматичного входу.
3. **Головний екран:**
   * **Швидкий доступ:** Іконки для переходу до календаря, звітів, налаштувань.
   * **Огляд фінансових даних:** Відображення основних фінансових показників (наприклад, загальні витрати, витрати на паливо, доходи від перевезень).
   * **Кнопка створення звітів:** Можливість швидкого переходу до створення нових фінансових звітів.
4. **Екран звітів:**
   * **Типи звітів:** Різні категорії звітів (витрати на паливо, витрати на обслуговування транспорту, доходи).
   * **Фільтри:** Фільтрація звітів за датою, типом витрат, маршрутом.
   * **Створення звіту:** Форма для створення нового звіту з можливістю вибору параметрів (наприклад, період, категорія витрат).
   * **Експорт звітів:** Можливість експорту звітів у різних форматах (PDF, Excel).
   * **Графіки та діаграми:** Візуалізація даних у вигляді графіків та діаграм для полегшення аналізу.
5. **Екран обліку палива:**
   * **Історія заправок:** Відображення історії заправок з деталізацією (назва та адреса АЗС, кількість літрів, ціна, загальна вартість, номер автомобіля, ПІБ водія, дата, час).
   * **Розрахунок витрат палива:** Автоматичний розрахунок витрат палива на основі даних про маршрути та пробіг.
   * **Внесення даних вручну:** Можливість вручну вносити дані про заправки та пробіг, з коментарями для додаткової інформації.
   * **Аналіз витрат:** Інструменти для аналізу витрат палива, включаючи порівняння з попередніми періодами.
6. **Екран управління бюджетом:**
   * **Планування бюджету:** Форма для планування бюджету на різні категорії витрат (паливо, обслуговування транспорту, заробітна плата водіїв).
   * **Контроль витрат:** Відображення фактичних витрат у порівнянні з запланованими, з виділенням перевищень та економії.
   * **Налаштування сповіщень:** Можливість налаштування сповіщень про перевищення бюджету або інші важливі фінансові події.
7. **Екран платежів:**
   * **Інвойси:** Відображення всіх інвойсів, створених для оплати послуг водіїв та палива.
   * **Статус платежів:** Відстеження статусу платежів (оплачено, очікує на оплату, прострочено).
   * **Створення інвойсів:** Форма для створення нових інвойсів з можливістю вибору параметрів (отримувач, сума, дата).
   * **Інтеграція з бухгалтерськими програмами:** Автоматичне формування інвойсів на основі даних з системи та їх інтеграція з бухгалтерськими програмами (наприклад, 1С).
8. **Екран обліку доходів:**
   * **Реєстрація доходів:** Введення даних про доходи від перевезень, включаючи деталі (дата, маршрут, сума).
   * **Аналіз доходів:** Інструменти для аналізу доходів, включаючи порівняння з попередніми періодами та прогнозування.
   * **Звіти про доходи:** Формування звітів про доходи з можливістю експорту у різних форматах.
9. **Екран обліку пробігу:**
   * **Дані про пробіг:** Відображення даних про пробіг, внесених вручну логістом на основі даних спідометра.
     + **Кілометраж на спідометрі на початок дня.**
     + **Довжина та час маршруту, розраховані за картою.**
     + **Різниця між розрахованим та фактичним пробігом.**
     + **Дані про заправки палива, якщо були.**
     + **Коментар від логіста.**
     + **Кілометраж.**
   * **Прийняття або відхилення даних:** Можливість прийняти або відхилити дані, внесені логістом, з зазначенням причини.
   * **Звітування додаткових витрат:** Форма для внесення додаткових витрат (наприклад, додаткове пальне і мастила, ремонт, евакуація, заміна автобуса на маршруті, непередбачувані витрати).
10. **Екран повідомлень:**
    * **Типи повідомлень:** Розділи для різних категорій повідомлень (наприклад, від водіїв, від бухгалтерії, системні повідомлення).
    * **Сповіщення про фінансові події:** Автоматичні сповіщення про важливі фінансові події (наприклад, надходження нових інвойсів, перевищення бюджету).

**Додаткові функції:**

* **Автоматизація задач:** Нейромережа автоматично розраховує витрати та доходи на основі введених даних.
* **Інтеграція з бухгалтерськими програмами:** Можливість автоматичного обміну даними з бухгалтерськими програмами для формування звітів та інвойсів.

1. Веб-інтерфейс для адміністратора системи

Головна база даних

Основні таблиці.

 **Users (Користувачі)**

* UserID (PK)
* Username
* Password
* Role (Driver, Operator, Financial Manager, Admin)
* ContactInfo

 **Vehicles (Транспортні засоби)**

* VehicleID (PK)
* VehicleType
* LicensePlate
* FuelType
* Capacity
* Status (Available, In Use, Under Maintenance)

 **Drivers (Водії)**

* DriverID (PK)
* UserID (FK)
* LicenseNumber
* AssignedVehicleID (FK)

 **Routes (Маршрути)**

* RouteID (PK)
* OperatorID (FK)
* RouteName
* StartLocation
* EndLocation
* TotalDistance
* TotalTime
* Status (Planned, Active, Completed)

 **Passengers (Пасажири)**

* PassengerID (PK)
* Name
* ContactInfo

 **Bookings (Замовлення)**

* BookingID (PK)
* PassengerID (FK)
* RouteID (FK)
* PickupLocation
* DropoffLocation
* PickupTime
* DropoffTime
* Status (Confirmed, Canceled)

 **FuelRecords (Облік палива)**

* FuelRecordID (PK)
* VehicleID (FK)
* DriverID (FK)
* Date
* FuelAmount
* FuelCost
* OdometerReading

 **Expenses (Витрати)**

* ExpenseID (PK)
* RouteID (FK)
* Description
* Amount
* Date

 **Invoices (Інвойси)**

* InvoiceID (PK)
* DriverID (FK)
* Date
* Amount
* Status (Pending, Paid)

 **Logs (Логи)**

* LogID (PK)
* UserID (FK)
* Action
* Timestamp

Діаграма

digraph ERD {

rankdir=LR;

node [shape=record];

Users [label="{Users|UserID : int (PK)\lUsername : varchar\lPassword : varchar\lRole : varchar\lContactInfo : varchar\l}"];

Vehicles [label="{Vehicles|VehicleID : int (PK)\lVehicleType : varchar\lLicensePlate : varchar\lFuelType : varchar\lCapacity : int\lStatus : varchar\l}"];

Drivers [label="{Drivers|DriverID : int (PK)\lUserID : int (FK)\lLicenseNumber : varchar\lAssignedVehicleID : int (FK)\l}"];

Routes [label="{Routes|RouteID : int (PK)\lOperatorID : int (FK)\lRouteName : varchar\lStartLocation : varchar\lEndLocation : varchar\lTotalDistance : float\lTotalTime : float\lStatus : varchar\l}"];

Passengers [label="{Passengers|PassengerID : int (PK)\lName : varchar\lContactInfo : varchar\l}"];

Bookings [label="{Bookings|BookingID : int (PK)\lPassengerID : int (FK)\lRouteID : int (FK)\lPickupLocation : varchar\lDropoffLocation : varchar\lPickupTime : datetime\lDropoffTime : datetime\lStatus : varchar\l}"];

FuelRecords [label="{FuelRecords|FuelRecordID : int (PK)\lVehicleID : int (FK)\lDriverID : int (FK)\lDate : datetime\lFuelAmount : float\lFuelCost : float\lOdometerReading : int\l}"];

Expenses [label="{Expenses|ExpenseID : int (PK)\lRouteID : int (FK)\lDescription : varchar\lAmount : float\lDate : datetime\l}"];

Invoices [label="{Invoices|InvoiceID : int (PK)\lDriverID : int (FK)\lDate : datetime\lAmount : float\lStatus : varchar\l}"];

Logs [label="{Logs|LogID : int (PK)\lUserID : int (FK)\lAction : varchar\lTimestamp : datetime\l}"];

Users -> Drivers [label="1 to 1"];

Users -> Logs [label="1 to many"];

Vehicles -> Drivers [label="1 to 1"];

Drivers -> FuelRecords [label="1 to many"];

Routes -> Bookings [label="1 to many"];

Routes -> Expenses [label="1 to many"];

Drivers -> Invoices [label="1 to many"];

Passengers -> Bookings [label="1 to many"];

Vehicles -> FuelRecords [label="1 to many"];

Routes -> Invoices [label="1 to many"];

}

План опису БД.

1. **Докладний опис сутностей і атрибутів**:
   * Опис кожного атрибута (тип даних, довжина, можливі значення).
   * Які атрибути є обов'язковими, а які - необов'язковими.
2. **Зв'язки між сутностями**:
   * Типи зв'язків (один-до-одного, один-до-багатьох, багато-до-багатьох).
   * Кардинальності (мінімальна та максимальна кількість зв'язків для кожної сутності).
3. **Первинні та зовнішні ключі**:
   * Які атрибути є первинними ключами.
   * Які атрибути є зовнішніми ключами і на які сутності вони посилаються.
4. **Індекси**:
   * Які атрибути потрібно індексувати для оптимізації запитів.
5. **Тригери та збережені процедури**:
   * Які тригери та збережені процедури потрібно створити для забезпечення цілісності даних і автоматизації певних процесів.
6. **Валідаційні правила**:
   * Правила валідації для різних атрибутів (наприклад, формат електронної пошти, діапазон значень для числових полів).
7. **Додаткові бізнес-правила**:
   * Специфічні правила та обмеження, які необхідно врахувати при проектуванні бази даних.
8. **Права доступу та безпека**:
   * Хто матиме доступ до яких даних і які права (читання, запис, оновлення, видалення) матиме кожен тип користувача.
9. **Приклади даних**:
   * Зразки даних для кожної сутності для кращого розуміння структури та взаємодії даних.
10. **Опис бізнес-процесів**:

* Опис основних бізнес-процесів, які система повинна підтримувати, щоб розробник міг краще зрозуміти контекст використання бази даних.

1. **Міграція даних**:

* Якщо є потреба в міграції даних зі старої системи, надайте інформацію про те, як це буде виконано.

1. **Технічні вимоги**:

* Вимоги до продуктивності, обсягів даних, резервного копіювання та відновлення.

**1. Докладний опис сутностей і атрибутів**

**1.1. Сутність: Водій (Driver)**

* **ID Водія (driver\_id)**: Унікальний ідентифікатор водія (ціле число, автозаповнення, первинний ключ)
* **Ім'я (first\_name)**: Ім'я водія (рядок, обов'язковий)
* **Прізвище (last\_name)**: Прізвище водія (рядок, обов'язковий)
* **Дата народження (date\_of\_birth)**: Дата народження водія (дата, необов'язковий)
* **Телефон (phone\_number)**: Контактний номер телефону (рядок, обов'язковий)
* **Електронна пошта (email)**: Адреса електронної пошти (рядок, необов'язковий)
* **Ліцензія (license\_number)**: Номер водійської ліцензії (рядок, обов'язковий)
* **Адреса (address)**: Домашня адреса водія (рядок, необов'язковий)

**1.2. Сутність: Автобус (Bus)**

* **ID Автобуса (bus\_id)**: Унікальний ідентифікатор автобуса (ціле число, автозаповнення, первинний ключ)
* **Номерний знак (license\_plate)**: Номерний знак автобуса (рядок, обов'язковий)
* **Модель (model)**: Модель автобуса (рядок, обов'язковий)
* **Рік випуску (year\_of\_manufacture)**: Рік випуску автобуса (ціле число, обов'язковий)
* **Тип палива (fuel\_type)**: Тип палива (рядок, обов'язковий)
* **Витрата палива (fuel\_consumption)**: Витрата палива на 100 км (десяткове число, обов'язковий)

**1.3. Сутність: Пасажир (Passenger)**

* **ID Пасажира (passenger\_id)**: Унікальний ідентифікатор пасажира (ціле число, автозаповнення, первинний ключ)
* **Ім'я (first\_name)**: Ім'я пасажира (рядок, обов'язковий)
* **Прізвище (last\_name)**: Прізвище пасажира (рядок, обов'язковий)
* **Телефон (phone\_number)**: Контактний номер телефону (рядок, обов'язковий)
* **Електронна пошта (email)**: Адреса електронної пошти (рядок, необов'язковий)
* **Адреса (address)**: Адреса проживання (рядок, необов'язковий)

**1.4. Сутність: Маршрут (Route)**

* **ID Маршруту (route\_id)**: Унікальний ідентифікатор маршруту (ціле число, автозаповнення, первинний ключ)
* **Назва (route\_name)**: Назва маршруту (рядок, обов'язковий)
* **Початкова точка (start\_location)**: Початкова точка маршруту (рядок, обов'язковий)
* **Кінцева точка (end\_location)**: Кінцева точка маршруту (рядок, обов'язковий)
* **Відстань (distance)**: Відстань маршруту в кілометрах (десяткове число, обов'язковий)
* **Орієнтовний час (estimated\_time)**: Орієнтовний час у дорозі (рядок, обов'язковий)

**1.5. Сутність: Розклад (Schedule)**

* **ID Розкладу (schedule\_id)**: Унікальний ідентифікатор розкладу (ціле число, автозаповнення, первинний ключ)
* **Дата (date)**: Дата виконання маршруту (дата, обов'язковий)
* **Час відправлення (departure\_time)**: Час відправлення (час, обов'язковий)
* **Час прибуття (arrival\_time)**: Час прибуття (час, обов'язковий)
* **ID Водія (driver\_id)**: Посилання на водія (ціле число, зовнішній ключ, обов'язковий)
* **ID Автобуса (bus\_id)**: Посилання на автобус (ціле число, зовнішній ключ, обов'язковий)
* **ID Маршруту (route\_id)**: Посилання на маршрут (ціле число, зовнішній ключ, обов'язковий)

**1.6. Сутність: Заправка (Fueling)**

* **ID Заправки (fueling\_id)**: Унікальний ідентифікатор заправки (ціле число, автозаповнення, первинний ключ)
* **Дата заправки (fueling\_date)**: Дата заправки (дата, обов'язковий)
* **Кількість літрів (liters)**: Кількість залитого палива в літрах (десяткове число, обов'язковий)
* **Вартість (cost)**: Вартість заправки (десяткове число, обов'язковий)
* **ID Автобуса (bus\_id)**: Посилання на автобус (ціле число, зовнішній ключ, обов'язковий)
* **ID Водія (driver\_id)**: Посилання на водія (ціле число, зовнішній ключ, обов'язковий)

**1.7. Сутність: Користувач (User)**

* **ID Користувача (user\_id)**: Унікальний ідентифікатор користувача (ціле число, автозаповнення, первинний ключ)
* **Ім'я користувача (username)**: Логін користувача (рядок, обов'язковий)
* **Пароль (password)**: Пароль користувача (рядок, обов'язковий)
* **Роль (role)**: Роль користувача в системі (рядок, обов'язковий, можливі значення: "водій", "логіст", "фінансовий менеджер", "адміністратор")

**1.8. Сутність: Витрати (Expenses)**

* **ID Витрати (expense\_id)**: Унікальний ідентифікатор витрати (ціле число, автозаповнення, первинний ключ)
* **Дата (date)**: Дата витрати (дата, обов'язковий)
* **Сума (amount)**: Сума витрати (десяткове число, обов'язковий)
* **Тип витрати (expense\_type)**: Тип витрати (рядок, обов'язковий, можливі значення: "пальне", "ремонт", "інше")
* **Опис (description)**: Опис витрати (рядок, необов'язковий)
* **ID Водія (driver\_id)**: Посилання на водія (ціле число, зовнішній ключ, необов'язковий)
* **ID Автобуса (bus\_id)**: Посилання на автобус (ціле число, зовнішній ключ, необов'язковий)

**2. Зв'язки між сутностями**

**2.1. Водій (Driver) і Розклад (Schedule)**

* **Зв'язок**: Один водій може бути призначений на багато розкладів, але кожен розклад має лише одного водія.
* **Тип зв'язку**: Один до багатьох (1

)

* **Зовнішній ключ**: driver\_id в таблиці Schedule посилається на driver\_id в таблиці Driver.

**2.2. Автобус (Bus) і Розклад (Schedule)**

* **Зв'язок**: Один автобус може бути призначений на багато розкладів, але кожен розклад має лише один автобус.
* **Тип зв'язку**: Один до багатьох (1

)

* **Зовнішній ключ**: bus\_id в таблиці Schedule посилається на bus\_id в таблиці Bus.

**2.3. Маршрут (Route) і Розклад (Schedule)**

* **Зв'язок**: Один маршрут може бути призначений на багато розкладів, але кожен розклад має лише один маршрут.
* **Тип зв'язку**: Один до багатьох (1

)

* **Зовнішній ключ**: route\_id в таблиці Schedule посилається на route\_id в таблиці Route.

**2.4. Автобус (Bus) і Заправка (Fueling)**

* **Зв'язок**: Один автобус може мати багато заправок, але кожна заправка пов'язана лише з одним автобусом.
* **Тип зв'язку**: Один до багатьох (1

)

* **Зовнішній ключ**: bus\_id в таблиці Fueling посилається на bus\_id в таблиці Bus.

**2.5. Водій (Driver) і Заправка (Fueling)**

* **Зв'язок**: Один водій може мати багато заправок, але кожна заправка пов'язана лише з одним водієм.
* **Тип зв'язку**: Один до багатьох (1

)

* **Зовнішній ключ**: driver\_id в таблиці Fueling посилається на driver\_id в таблиці Driver.

**2.6. Водій (Driver) і Витрати (Expenses)**

* **Зв'язок**: Один водій може мати багато витрат, але кожна витрата пов'язана лише з одним водієм.
* **Тип зв'язку**: Один до багатьох (1

)

* **Зовнішній ключ**: driver\_id в таблиці Expenses посилається на driver\_id в таблиці Driver.

**2.7. Автобус (Bus) і Витрати (Expenses)**

* **Зв'язок**: Один автобус може мати багато витрат, але кожна витрата пов'язана лише з одним автобусом.
* **Тип зв'язку**: Один до багатьох (1

)

* **Зовнішній ключ**: bus\_id в таблиці Expenses посилається на bus\_id в таблиці Bus.

**2.8. Користувач (User) і Водій (Driver)**

* **Зв'язок**: Один користувач може мати одного водія (у разі, якщо користувач є водієм), але один водій не може бути пов'язаний з декількома користувачами.
* **Тип зв'язку**: Один до одного (1:1)
* **Зовнішній ключ**: user\_id в таблиці Driver посилається на user\_id в таблиці User.

**3.1. Таблиця User**

* **Первинний ключ**: user\_id
* **Зовнішні ключі**: немає

**3.2. Таблиця Driver**

* **Первинний ключ**: driver\_id
* **Зовнішні ключі**:
  + user\_id посилається на user\_id в таблиці User

**3.3. Таблиця Bus**

* **Первинний ключ**: bus\_id
* **Зовнішні ключі**: немає

**3.4. Таблиця Route**

* **Первинний ключ**: route\_id
* **Зовнішні ключі**: немає

**3.5. Таблиця Schedule**

* **Первинний ключ**: schedule\_id
* **Зовнішні ключі**:
  + driver\_id посилається на driver\_id в таблиці Driver
  + bus\_id посилається на bus\_id в таблиці Bus
  + route\_id посилається на route\_id в таблиці Route

**3.6. Таблиця Fueling**

* **Первинний ключ**: fueling\_id
* **Зовнішні ключі**:
  + bus\_id посилається на bus\_id в таблиці Bus
  + driver\_id посилається на driver\_id в таблиці Driver

**3.7. Таблиця Expenses**

* **Первинний ключ**: expense\_id
* **Зовнішні ключі**:
  + driver\_id посилається на driver\_id в таблиці Driver
  + bus\_id посилається на bus\_id в таблиці Bus

**4. Індекси**

**4.1. Таблиця User**

* **Первинний індекс**: PRIMARY (user\_id)
* **Додаткові індекси**:
  + INDEX (username)
  + INDEX (email)

**4.2. Таблиця Driver**

* **Первинний індекс**: PRIMARY (driver\_id)
* **Додаткові індекси**:
  + INDEX (user\_id)

**4.3. Таблиця Bus**

* **Первинний індекс**: PRIMARY (bus\_id)
* **Додаткові індекси**:
  + INDEX (license\_plate)

**4.4. Таблиця Route**

* **Первинний індекс**: PRIMARY (route\_id)
* **Додаткові індекси**:
  + INDEX (route\_name)

**4.5. Таблиця Schedule**

* **Первинний індекс**: PRIMARY (schedule\_id)
* **Додаткові індекси**:
  + INDEX (driver\_id)
  + INDEX (bus\_id)
  + INDEX (route\_id)

**4.6. Таблиця Fueling**

* **Первинний індекс**: PRIMARY (fueling\_id)
* **Додаткові індекси**:
  + INDEX (bus\_id)
  + INDEX (driver\_id)

**4.7. Таблиця Expenses**

* **Первинний індекс**: PRIMARY (expense\_id)
* **Додаткові індекси**:
  + INDEX (driver\_id)
  + INDEX (bus\_id)

**5. Тригери та збережені процедури**

**5.1 Тригери**

Тригери будуть використовуватися для автоматичного виконання певних дій у базі даних, коли відбуваються визначені події (наприклад, вставка, оновлення або видалення записів). Нижче наведено перелік тригерів, які повинні бути реалізовані в системі NextPointLogix:

1. **Trigger: UpdateVehicleStatus**
   * **Подія**: Оновлення запису у таблиці Vehicles
   * **Опис**: Автоматично оновлює статус транспортного засобу у таблиці Vehicles після завершення маршруту.
   * **Дія**:

sql

Copy code

CREATE TRIGGER UpdateVehicleStatus

AFTER UPDATE ON Routes

FOR EACH ROW

BEGIN

IF NEW.Status = 'Completed' THEN

UPDATE Vehicles

SET Status = 'Available'

WHERE VehicleID = NEW.VehicleID;

END IF;

END;

1. **Trigger: LogFuelEntry**
   * **Подія**: Вставка запису у таблицю FuelRecords
   * **Опис**: Автоматично додає запис у таблицю Logs після додавання запису про заправку палива.
   * **Дія**:

sql

Copy code

CREATE TRIGGER LogFuelEntry

AFTER INSERT ON FuelRecords

FOR EACH ROW

BEGIN

INSERT INTO Logs (UserID, Action, Timestamp)

VALUES (NEW.DriverID, 'Added fuel record', NOW());

END;

1. **Trigger: InvoiceStatusUpdate**
   * **Подія**: Оновлення запису у таблиці Invoices
   * **Опис**: Автоматично додає запис у таблицю Logs після зміни статусу інвойсу.
   * **Дія**:

sql

Copy code

CREATE TRIGGER InvoiceStatusUpdate

AFTER UPDATE ON Invoices

FOR EACH ROW

BEGIN

IF NEW.Status != OLD.Status THEN

INSERT INTO Logs (UserID, Action, Timestamp)

VALUES (NEW.DriverID, CONCAT('Invoice status changed to ', NEW.Status), NOW());

END IF;

END;

Додаткові тригери

Перелік тригерів, приведений вище, покриває основні автоматизаційні вимоги для системи NextPointLogix. Однак, в залежності від деталей проекту, можуть бути потрібні додаткові тригери для обробки специфічних ситуацій чи додаткових функцій.

Ось кілька можливих додаткових тригерів, які можуть бути корисні для системи:

1. **Trigger: UpdateDriverStatus**
   * **Подія**: Оновлення запису у таблиці Routes
   * **Опис**: Автоматично оновлює статус водія у таблиці Drivers після завершення маршруту.
   * **Дія**:

sql

Copy code

CREATE TRIGGER UpdateDriverStatus

AFTER UPDATE ON Routes

FOR EACH ROW

BEGIN

IF NEW.Status = 'Completed' THEN

UPDATE Drivers

SET Status = 'Available'

WHERE DriverID = NEW.DriverID;

END IF;

END;

1. **Trigger: LogLoginAttempt**
   * **Подія**: Вставка запису у таблицю LoginAttempts
   * **Опис**: Автоматично додає запис у таблицю Logs після спроби входу користувача.
   * **Дія**:

sql

Copy code

CREATE TRIGGER LogLoginAttempt

AFTER INSERT ON LoginAttempts

FOR EACH ROW

BEGIN

INSERT INTO Logs (UserID, Action, Timestamp)

VALUES (NEW.UserID, 'User login attempt', NOW());

END;

1. **Trigger: NotifyOnNewRoute**
   * **Подія**: Вставка запису у таблицю Routes
   * **Опис**: Автоматично надсилає повідомлення водію після додавання нового маршруту.
   * **Дія**:

sql

Copy code

CREATE TRIGGER NotifyOnNewRoute

AFTER INSERT ON Routes

FOR EACH ROW

BEGIN

INSERT INTO Notifications (UserID, Message, Timestamp)

VALUES (NEW.DriverID, CONCAT('New route assigned: ', NEW.RouteDetails), NOW());

END;

1. **Trigger: ValidateFuelEntry**
   * **Подія**: Вставка запису у таблицю FuelRecords
   * **Опис**: Перевіряє валідність даних про заправку палива.
   * **Дія**:

sql

Copy code

CREATE TRIGGER ValidateFuelEntry

BEFORE INSERT ON FuelRecords

FOR EACH ROW

BEGIN

IF NEW.FuelAmount <= 0 THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Invalid fuel amount';

END IF;

END;

**5.2 Збережені процедури**

Збережені процедури будуть використовуватися для виконання складних запитів та операцій у базі даних. Нижче наведено приклади збережених процедур, які повинні бути реалізовані в системі NextPointLogix:

1. **Procedure: CalculateRouteFuelCost**
   * **Опис**: Розраховує загальну вартість палива для заданого маршруту.
   * **Дія**:

sql

Copy code

CREATE PROCEDURE CalculateRouteFuelCost(IN route\_id INT)

BEGIN

DECLARE total\_cost FLOAT;

SELECT SUM(FuelCost) INTO total\_cost

FROM FuelRecords

WHERE RouteID = route\_id;

SELECT total\_cost AS TotalFuelCost;

END;

1. **Procedure: GenerateInvoiceForDriver**
   * **Опис**: Генерує інвойс для водія за вказаний період.
   * **Дія**:

sql

Copy code

CREATE PROCEDURE GenerateInvoiceForDriver(IN driver\_id INT, IN start\_date DATE, IN end\_date DATE)

BEGIN

DECLARE total\_amount FLOAT;

SELECT SUM(Amount) INTO total\_amount

FROM Expenses

WHERE DriverID = driver\_id AND Date BETWEEN start\_date AND end\_date;

INSERT INTO Invoices (DriverID, Date, Amount, Status)

VALUES (driver\_id, NOW(), total\_amount, 'Pending');

END;

1. **Procedure: AssignVehicleToDriver**
   * **Опис**: Призначає транспортний засіб водію.
   * **Дія**:

sql

Copy code

CREATE PROCEDURE AssignVehicleToDriver(IN driver\_id INT, IN vehicle\_id INT)

BEGIN

UPDATE Drivers

SET AssignedVehicleID = vehicle\_id

WHERE DriverID = driver\_id;

UPDATE Vehicles

SET Status = 'In Use'

WHERE VehicleID = vehicle\_id;

END;

1. **Procedure: OptimizeRoutes**
   * **Опис**: Оптимізує маршрути на основі заданих параметрів.
   * **Дія**:

sql

Copy code

CREATE PROCEDURE OptimizeRoutes(IN optimization\_param VARCHAR(50))

BEGIN

-- Логіка оптимізації маршрутів на основі параметра

-- Наприклад, найменша кількість транспорту, найменша витрата палива і т.д.

END;

**6. Валідаційні правила**

Валідаційні правила забезпечують цілісність та правильність даних у системі. Нижче наведено основні валідаційні правила, які мають бути реалізовані в системі NextPointLogix:

**6.1 Користувачі**

1. **Перевірка унікальності імені користувача та електронної пошти**:
   * Ім'я користувача та електронна пошта мають бути унікальними в системі.

sql

Copy code

ALTER TABLE Users ADD CONSTRAINT UC\_Username UNIQUE (Username);

ALTER TABLE Users ADD CONSTRAINT UC\_Email UNIQUE (Email);

1. **Перевірка формату електронної пошти**:
   * Електронна пошта має відповідати стандартному формату.

sql

Copy code

CREATE TRIGGER ValidateEmailFormat

BEFORE INSERT OR UPDATE ON Users

FOR EACH ROW

BEGIN

IF NEW.Email NOT REGEXP '^[A-Za-z0-9.\_%+-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,}$' THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Invalid email format';

END IF;

END;

1. **Перевірка довжини паролю**:
   * Пароль повинен містити щонайменше 8 символів.

sql

Copy code

CREATE TRIGGER ValidatePasswordLength

BEFORE INSERT OR UPDATE ON Users

FOR EACH ROW

BEGIN

IF LENGTH(NEW.Password) < 8 THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Password must be at least 8 characters long';

END IF;

END;

**6.2 Маршрути**

1. **Перевірка валідності дат та часу маршруту**:
   * Дата початку повинна бути раніше дати завершення.

sql

Copy code

CREATE TRIGGER ValidateRouteDates

BEFORE INSERT OR UPDATE ON Routes

FOR EACH ROW

BEGIN

IF NEW.StartTime >= NEW.EndTime THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'End time must be after start time';

END IF;

END;

1. **Перевірка наявності водія та автобуса**:
   * Водій та автобус повинні бути доступні на вибраний час.

sql

Copy code

CREATE TRIGGER ValidateDriverAndBusAvailability

BEFORE INSERT ON Routes

FOR EACH ROW

BEGIN

IF (SELECT COUNT(\*) FROM Routes WHERE DriverID = NEW.DriverID AND NEW.StartTime BETWEEN StartTime AND EndTime) > 0 THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Driver is not available at this time';

END IF;

IF (SELECT COUNT(\*) FROM Routes WHERE BusID = NEW.BusID AND NEW.StartTime BETWEEN StartTime AND EndTime) > 0 THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Bus is not available at this time';

END IF;

END;

**6.3 Заправка палива**

1. **Перевірка позитивної кількості палива**:
   * Кількість заправленого палива повинна бути більше нуля.

sql

Copy code

CREATE TRIGGER ValidateFuelAmount

BEFORE INSERT OR UPDATE ON FuelRecords

FOR EACH ROW

BEGIN

IF NEW.FuelAmount <= 0 THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Fuel amount must be greater than zero';

END IF;

END;

1. **Перевірка валідності ціни за літр**:
   * Ціна за літр повинна бути більше нуля.

sql

Copy code

CREATE TRIGGER ValidateFuelPrice

BEFORE INSERT OR UPDATE ON FuelRecords

FOR EACH ROW

BEGIN

IF NEW.FuelPrice <= 0 THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Fuel price must be greater than zero';

END IF;

END;

**6.4 Автобуси**

1. **Перевірка унікальності номерного знака автобуса**:
   * Номерний знак автобуса повинен бути унікальним.

sql

Copy code

ALTER TABLE Buses ADD CONSTRAINT UC\_BusNumberPlate UNIQUE (NumberPlate);

**6.5 Інші дані**

1. **Перевірка валідності контактного номеру телефону**:
   * Номер телефону повинен відповідати стандартному формату.

sql

Copy code

CREATE TRIGGER ValidatePhoneNumberFormat

BEFORE INSERT OR UPDATE ON Contacts

FOR EACH ROW

BEGIN

IF NEW.PhoneNumber NOT REGEXP '^[0-9]{10}$' THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Invalid phone number format';

END IF;

END;

**8. Права доступу та безпека**

У системі NextPointLogix важливо забезпечити належний рівень доступу до даних та їх захист. Права доступу повинні бути налаштовані для кожного типу користувача відповідно до їх ролей і обов'язків. Нижче наведено основні права доступу та безпеку для різних типів користувачів:

**8.1 Ролі та права доступу**

1. **Адміністратор системи**
   * **Доступ до даних**: Усі дані системи.
   * **Права**: Читання, запис, оновлення, видалення.
   * **Дії**: Керування користувачами, налаштування системи, перегляд та редагування всіх даних, створення звітів.
2. **Логіст-оператор**
   * **Доступ до даних**: Дані про маршрути, автобуси, водіїв, пасажирів, календар подій.
   * **Права**: Читання, запис, оновлення, видалення.
   * **Дії**: Планування маршрутів, моніторинг транспорту, облік палива, редагування даних про маршрути.
3. **Фінансовий менеджер**
   * **Доступ до даних**: Дані про витрати, доходи, пробіг, заправки палива, звіти про фінансову діяльність.
   * **Права**: Читання, запис, оновлення.
   * **Дії**: Аналіз витрат та доходів, створення фінансових звітів, перевірка даних про пробіг та витрати на паливо.
4. **Водії**
   * **Доступ до даних**: Власні маршрути, календар подій, дані про заправки палива.
   * **Права**: Читання, запис.
   * **Дії**: Перегляд та підтвердження маршрутів, внесення даних про заправки палива, отримання повідомлень від оператора.
5. **Пасажири**
   * **Доступ до даних**: Дані про свої заявки на поїздки, статус маршруту.
   * **Права**: Читання.
   * **Дії**: Перегляд статусу заявки на поїздку, отримання повідомлень про статус маршруту.

**8.2 Безпека даних**

1. **Аутентифікація та авторизація**
   * Використання багатофакторної аутентифікації для доступу до системи.
   * Рольова модель доступу до даних для забезпечення належного рівня доступу.
2. **Шифрування**
   * Шифрування даних при зберіганні та передачі для захисту конфіденційної інформації.
   * Використання SSL/TLS для захищеного з'єднання.
3. **Журнали аудиту**
   * Запис дій користувачів у системі для моніторингу та виявлення підозрілої активності.
   * Регулярний перегляд журналів аудиту для виявлення та усунення потенційних загроз.
4. **Резервне копіювання**
   * Регулярне створення резервних копій даних для запобігання втраті інформації.
   * Зберігання резервних копій у захищених локаціях.
5. **Виявлення та реагування на інциденти**
   * Налаштування системи моніторингу для виявлення потенційних загроз.
   * План реагування на інциденти для швидкого усунення наслідків порушень безпеки.
6. **Політики безпеки**
   * Встановлення політик щодо паролів, доступу до даних та користування системою.
   * Регулярне оновлення політик безпеки відповідно до нових загроз та вимог.

**9. Приклади даних**

У цьому розділі наведені приклади даних, які будуть використовуватися в системі NextPointLogix. Ці приклади допоможуть краще зрозуміти структуру даних та забезпечать основу для тестування системи.

**9.1 Дані користувачів**

**Адміністратор системи:**

* **Ім'я користувача**: admin
* **Пароль**: \*\*\*\*\*\*\*\*
* **Електронна пошта**: admin@NextPointLogix.com
* **Роль**: Адміністратор

**Логіст-оператор:**

* **Ім'я користувача**: logist1
* **Пароль**: \*\*\*\*\*\*\*\*
* **Електронна пошта**: logist1@NextPointLogix.com
* **Роль**: Логіст-оператор

**Фінансовий менеджер:**

* **Ім'я користувача**: finance1
* **Пароль**: \*\*\*\*\*\*\*\*
* **Електронна пошта**: finance1@NextPointLogix.com
* **Роль**: Фінансовий менеджер

**Водій:**

* **Ім'я користувача**: driver1
* **Пароль**: \*\*\*\*\*\*\*\*
* **Електронна пошта**: driver1@NextPointLogix.com
* **Роль**: Водій

**Пасажир:**

* **Ім'я користувача**: passenger1
* **Пароль**: \*\*\*\*\*\*\*\*
* **Електронна пошта**: passenger1@NextPointLogix.com
* **Роль**: Пасажир

**9.2 Дані маршрутів**

**Маршрут 1:**

* **Ідентифікатор маршруту**: 001
* **Назва маршруту**: Маршрут A
* **Дата**: 2024-08-10
* **Час початку**: 08:00
* **Час завершення**: 12:00
* **Кількість зупинок**: 5
* **Протяжність маршруту (км)**: 50

**Маршрут 2:**

* **Ідентифікатор маршруту**: 002
* **Назва маршруту**: Маршрут B
* **Дата**: 2024-08-10
* **Час початку**: 14:00
* **Час завершення**: 18:00
* **Кількість зупинок**: 7
* **Протяжність маршруту (км)**: 70

**9.3 Дані транспортних засобів**

**Автобус 1:**

* **Ідентифікатор автобуса**: BUS001
* **Номерний знак**: AB1234CD
* **Модель**: Mercedes-Benz Sprinter
* **Рік випуску**: 2020
* **Водій**: driver1
* **Пробіг (км)**: 30000

**Автобус 2:**

* **Ідентифікатор автобуса**: BUS002
* **Номерний знак**: EF5678GH
* **Модель**: Ford Transit
* **Рік випуску**: 2019
* **Водій**: driver2
* **Пробіг (км)**: 45000

**9.4 Дані про заправки**

**Заправка 1:**

* **Ідентифікатор заправки**: FUEL001
* **Дата**: 2024-08-10
* **Час**: 09:00
* **Автобус**: BUS001
* **Тип палива**: Дизель
* **Кількість літрів**: 50
* **Ціна за літр**: 1.20
* **Загальна вартість**: 60.00

**Заправка 2:**

* **Ідентифікатор заправки**: FUEL002
* **Дата**: 2024-08-10
* **Час**: 15:00
* **Автобус**: BUS002
* **Тип палива**: Дизель
* **Кількість літрів**: 70
* **Ціна за літр**: 1.20
* **Загальна вартість**: 84.00

**9.5 Дані пасажирів**

**Пасажир 1:**

* **Ідентифікатор пасажира**: P001
* **Ім'я**: Іван Іванов
* **Контактний номер**: +380501234567
* **Точка посадки**: вул. Центральна, 10
* **Точка висадки**: вул. Приміська, 20
* **Дата**: 2024-08-10
* **Час посадки**: 08:30
* **Час висадки**: 09:00

**Пасажир 2:**

* **Ідентифікатор пасажира**: P002
* **Ім'я**: Петро Петров
* **Контактний номер**: +380501234568
* **Точка посадки**: вул. Лісова, 5
* **Точка висадки**: вул. Набережна, 15
* **Дата**: 2024-08-10
* **Час посадки**: 14:30
* **Час висадки**: 15:00

Top of Form

Bottom of Form

**10. Опис бізнес-процесів**

Цей розділ описує ключові бізнес-процеси, які підтримує система NextPointLogix. Він надає огляд основних операційних процесів, визначає їхні етапи та пояснює взаємодію між користувачами та системою.

**10.1 Процес планування маршрутів**

**Мета:** Оптимізувати маршрути перевезень пасажирів, забезпечуючи ефективне використання транспортних засобів та мінімізацію витрат на паливо.

**Учасники:** Логіст-оператор, Водій, Система штучного інтелекту (AI).

**Етапи:**

1. **Отримання заявок:** Логіст-оператор отримує заявки від пасажирів на перевезення.
2. **Оптимізація маршрутів:** AI аналізує заявки та пропонує оптимальні маршрути на основі заданих параметрів (найменша кількість транспортних засобів, найменша витрата палива, найкоротша тривалість маршруту).
3. **Корекція маршрутів:** Логіст-оператор переглядає запропоновані маршрути, вносить корективи за потреби (кількість автомобілів, кількість точок на маршруті, максимальний час у дорозі).
4. **Затвердження маршрутів:** Логіст-оператор затверджує маршрути та інформує водіїв.
5. **Виконання маршрутів:** Водії виконують маршрути згідно з планом.

**10.2 Процес заправки палива**

**Мета:** Забезпечити точний облік витрат на паливо та моніторинг його використання.

**Учасники:** Водій, Логіст-оператор, Фінансовий менеджер.

**Етапи:**

1. **Внесення даних заправки:** Водій вводить дані про заправку (дата, час, тип палива, кількість літрів, ціна, загальна вартість).
2. **Верифікація даних:** Логіст-оператор перевіряє та підтверджує дані про заправку.
3. **Аналіз витрат:** Фінансовий менеджер аналізує дані про витрати на паливо, робить висновки та складає звіти.

**10.3 Процес обліку пробігу**

**Мета:** Забезпечити точний облік пробігу транспортних засобів для аналізу їх використання та планування технічного обслуговування.

**Учасники:** Водій, Логіст-оператор, Фінансовий менеджер.

**Етапи:**

1. **Введення показів спідометра:** Водій вводить покази спідометра на початку та в кінці дня.
2. **Верифікація пробігу:** Логіст-оператор перевіряє розрахований пробіг на основі даних спідометра та карти.
3. **Аналіз пробігу:** Фінансовий менеджер аналізує різницю між розрахованим і фактичним пробігом, приймає або відхиляє дані про пробіг, складає звіти про використання транспортних засобів.

**10.4 Процес управління транспортними засобами та водіями**

**Мета:** Забезпечити оптимальне використання транспортних засобів та ефективний розподіл водіїв.

**Учасники:** Адміністратор системи, Логіст-оператор.

**Етапи:**

1. **Реєстрація транспортних засобів та водіїв:** Адміністратор системи додає нові транспортні засоби та водіїв у систему.
2. **Планування змін:** Логіст-оператор планує зміни водіїв та транспортних засобів на маршрутах.
3. **Моніторинг доступності:** Логіст-оператор перевіряє доступність водіїв та транспортних засобів на конкретну дату.

**10.5 Процес фінансового контролю**

**Мета:** Забезпечити точний фінансовий облік витрат та доходів, пов'язаних з транспортними перевезеннями.

**Учасники:** Фінансовий менеджер.

**Етапи:**

1. **Облік витрат:** Фінансовий менеджер веде облік витрат на паливо, технічне обслуговування, ремонт тощо.
2. **Формування звітів:** Фінансовий менеджер формує звіти про фінансові показники діяльності транспортної компанії.
3. **Аналіз даних:** Фінансовий менеджер аналізує фінансові дані для виявлення можливих шляхів оптимізації витрат.

**10.6 Процес зворотного зв’язку та підтримки**

**Мета:** Забезпечити високий рівень задоволеності користувачів та оперативне вирішення проблем.

**Учасники:** Водій, Пасажир, Логіст-оператор, Адміністратор системи.

**Етапи:**

1. **Збір відгуків:** Водії та пасажири залишають відгуки про послуги.
2. **Аналіз відгуків:** Логіст-оператор аналізує отримані відгуки для покращення сервісу.
3. **Вирішення проблем:** Адміністратор системи вирішує технічні проблеми, що виникають у процесі роботи.

**ДИЗАЙН REST API для NextPointLogix:**

**Загальні принципи:**

1. **Автентифікація: Використання токенів доступу (JWT) для автентифікації.**
2. **Формат відповіді: JSON.**
3. **Версіонування: Наприклад, /api/v1/....**

**Кінцеві точки API:**

**1. Користувачі**

* **Створення користувача**
  + **Метод: POST**
  + **Кінцева точка: /api/v1/users**
  + **Опис: Створення нового користувача.**
  + **Параметри:**
    - **username: string**
    - **password: string**
    - **role: string (Driver, Operator, Financial Manager, Admin)**
    - **contactInfo: string**
* **Отримання інформації про користувача**
  + **Метод: GET**
  + **Кінцева точка: /api/v1/users/{userId}**
  + **Опис: Отримати інформацію про користувача за його ID.**
* **Оновлення користувача**
  + **Метод: PUT**
  + **Кінцева точка: /api/v1/users/{userId}**
  + **Опис: Оновити інформацію про користувача.**
  + **Параметри: як у створенні користувача.**
* **Видалення користувача**
  + **Метод: DELETE**
  + **Кінцева точка: /api/v1/users/{userId}**
  + **Опис: Видалити користувача.**

**2. Транспортні засоби**

* **Додавання транспортного засобу**
  + **Метод: POST**
  + **Кінцева точка: /api/v1/vehicles**
  + **Опис: Додавання нового транспортного засобу.**
  + **Параметри:**
    - **vehicleType: string**
    - **licensePlate: string**
    - **fuelType: string**
    - **capacity: integer**
    - **status: string (Available, In Use, Under Maintenance)**
* **Отримання інформації про транспортний засіб**
  + **Метод: GET**
  + **Кінцева точка: /api/v1/vehicles/{vehicleId}**
  + **Опис: Отримати інформацію про транспортний засіб за його ID.**
* **Оновлення транспортного засобу**
  + **Метод: PUT**
  + **Кінцева точка: /api/v1/vehicles/{vehicleId}**
  + **Опис: Оновити інформацію про транспортний засіб.**
  + **Параметри: як у додаванні транспортного засобу.**
* **Видалення транспортного засобу**
  + **Метод: DELETE**
  + **Кінцева точка: /api/v1/vehicles/{vehicleId}**
  + **Опис: Видалити транспортний засіб.**

**3. Маршрути**

* **Додавання маршруту**
  + **Метод: POST**
  + **Кінцева точка: /api/v1/routes**
  + **Опис: Додавання нового маршруту.**
  + **Параметри:**
    - **routeName: string**
    - **startLocation: string**
    - **endLocation: string**
    - **distance: decimal**
    - **estimatedTime: string**
* **Отримання інформації про маршрут**
  + **Метод: GET**
  + **Кінцева точка: /api/v1/routes/{routeId}**
  + **Опис: Отримати інформацію про маршрут за його ID.**
* **Оновлення маршруту**
  + **Метод: PUT**
  + **Кінцева точка: /api/v1/routes/{routeId}**
  + **Опис: Оновити інформацію про маршрут.**
  + **Параметри: як у додаванні маршруту.**
* **Видалення маршруту**
  + **Метод: DELETE**
  + **Кінцева точка: /api/v1/routes/{routeId}**
  + **Опис: Видалити маршрут.**

**4. Розклад**

* **Додавання розкладу**
  + **Метод: POST**
  + **Кінцева точка: /api/v1/schedules**
  + **Опис: Додавання нового розкладу.**
  + **Параметри:**
    - **date: date**
    - **departureTime: time**
    - **arrivalTime: time**
    - **driverId: integer**
    - **busId: integer**
    - **routeId: integer**
* **Отримання інформації про розклад**
  + **Метод: GET**
  + **Кінцева точка: /api/v1/schedules/{scheduleId}**
  + **Опис: Отримати інформацію про розклад за його ID.**
* **Оновлення розкладу**
  + **Метод: PUT**
  + **Кінцева точка: /api/v1/schedules/{scheduleId}**
  + **Опис: Оновити інформацію про розклад.**
  + **Параметри: як у додаванні розкладу.**
* **Видалення розкладу**
  + **Метод: DELETE**
  + **Кінцева точка: /api/v1/schedules/{scheduleId}**
  + **Опис: Видалити розклад.**

**5. Заправки**

* **Додавання заправки**
  + **Метод: POST**
  + **Кінцева точка: /api/v1/fuelings**
  + **Опис: Додавання нової заправки.**
  + **Параметри:**
    - **fuelingDate: date**
    - **liters: decimal**
    - **cost: decimal**
    - **busId: integer**
    - **driverId: integer**
* **Отримання інформації про заправку**
  + **Метод: GET**
  + **Кінцева точка: /api/v1/fuelings/{fuelingId}**
  + **Опис: Отримати інформацію про заправку за її ID.**
* **Оновлення заправки**
  + **Метод: PUT**
  + **Кінцева точка: /api/v1/fuelings/{fuelingId}**
  + **Опис: Оновити інформацію про заправку.**
  + **Параметри: як у додаванні заправки.**
* **Видалення заправки**
  + **Метод: DELETE**
  + **Кінцева точка: /api/v1/fuelings/{fuelingId}**
  + **Опис: Видалити заправку.**

**Примітки:**

* **Усі кінцеві точки повинні перевіряти права доступу користувача.**
* **Дані, які стосуються безпеки, повинні бути зашифровані.**
* **Обробка помилок повинна бути зрозумілою і надавати детальну інформацію про помилки.**

**Data Models**

This section describes the essential data models and their relationships for the NextPointLogix project. These models define the structure of the database and how different entities interact with each other. The data models are designed to support all functional and non-functional requirements outlined in the technical specification.

**1. User Model**

**Attributes:**

* user\_id: Unique identifier for each user (Primary Key)
* username: Login name for authentication
* password: Encrypted password for security
* email: User's email address for communication
* role: Defines the role of the user (e.g., driver, operator, manager, administrator)

**2. Driver Model**

**Attributes:**

* driver\_id: Unique identifier for each driver (Primary Key)
* user\_id: Foreign key linking to the User model
* name: Full name of the driver
* license\_number: Driver's license number
* phone\_number: Contact number

**3. Vehicle Model**

**Attributes:**

* vehicle\_id: Unique identifier for each vehicle (Primary Key)
* license\_plate: Vehicle's license plate number
* model: Vehicle model name
* capacity: Number of passengers the vehicle can transport
* fuel\_type: Type of fuel used by the vehicle

**4. Route Model**

**Attributes:**

* route\_id: Unique identifier for each route (Primary Key)
* start\_location: Starting point of the route
* end\_location: Destination of the route
* distance: Distance of the route in kilometers
* estimated\_time: Estimated travel time for the route

**5. Trip Model**

**Attributes:**

* trip\_id: Unique identifier for each trip (Primary Key)
* route\_id: Foreign key linking to the Route model
* driver\_id: Foreign key linking to the Driver model
* vehicle\_id: Foreign key linking to the Vehicle model
* date: Date of the trip
* status: Current status of the trip (e.g., scheduled, ongoing, completed)

**6. Fuel Log Model**

**Attributes:**

* fuel\_log\_id: Unique identifier for each fuel entry (Primary Key)
* trip\_id: Foreign key linking to the Trip model
* fuel\_amount: Amount of fuel added
* price: Price per unit of fuel
* date: Date of fueling
* station\_name: Name of the fuel station

**7. Feedback Model**

**Attributes:**

* feedback\_id: Unique identifier for each feedback entry (Primary Key)
* user\_id: Foreign key linking to the User model
* trip\_id: Foreign key linking to the Trip model
* comments: User's feedback or comments
* rating: User's rating for the trip

**8. Booking Request Model**

**Attributes:**

* booking\_id: Unique identifier for each booking request (Primary Key)
* user\_id: Foreign key linking to the User model
* route\_id: Foreign key linking to the Route model
* date: Date of the requested trip
* status: Current status of the booking (e.g., pending, confirmed, canceled)

**Relationships**

* **User to Driver**: One-to-One
* **User to Feedback**: One-to-Many
* **Driver to Trip**: One-to-Many
* **Vehicle to Trip**: One-to-Many
* **Route to Trip**: One-to-Many
* **Trip to Fuel Log**: One-to-Many
* **User to Booking Request**: One-to-Many
* **Route to Booking Request**: One-to-Many

**Additional Considerations**

* Implement necessary indexing for attributes frequently queried to optimize performance.
* Ensure proper validation and constraints on the attributes, such as unique constraints for username, email, and license\_plate.

This data model structure ensures robust handling of all operations required by the NextPointLogix project, facilitating efficient management of trips, user roles, and resources.

(додаю для майбутнього включення в опис технічного завдання логіку формування списку пасажирів на маршрут)

### Логіка формування списку пасажирів на маршрут

#### \*\*1. Заборона поєднання поїздок "На роботу" і "Додому":

* **Правило:** У маршруті можуть бути або заявки "На роботу", або "Додому".
  + Тільки якщо встановлено чекбокс "Дозволити змішані заявки", обидва напрямки можуть бути включені в один маршрут.
* **Дії:**
  + Перевірити значення direction для кожної заявки в списку.
  + Якщо заявки мають різні значення direction, показати попередження і не дозволити сформувати маршрут без дозволу.

#### **2. Обмеження тривалості маршруту:**

* **Правило:** Встановити максимальну тривалість маршруту (за замовчуванням, наприклад, 4 години). Якщо тривалість перевищена:
  + Попередити користувача.
  + Додати чекбокс "Дозволити збільшити" для розширення тривалості.
  + Додати поле для введення користувачем нового максимального часу.
* **Дії:**
  + Розрахувати тривалість маршруту за допомогою Google Directions API.
  + Якщо перевищено встановлене обмеження, показати попередження.
  + Дозволити користувачеві змінити максимальну тривалість.

#### **3. Інтервал часу для прибуття на роботу:**

* **Правило:** Усі пасажири повинні прибути на роботу відповідно до найранішої заявки.
  + Якщо найраніший час прибуття — 08:00, автобус повинен прибути на цей час.
  + Пасажири, які бажають прибути пізніше, автоматично матимуть час прибуття раніше запланованого.
* **Дії:**
  + Визначити найраніший час прибуття серед заявок.
  + Порівняти цей час із кожною заявкою.
  + Якщо різниця перевищує заданий інтервал (наприклад, 30 хвилин), показати попередження користувачу.

#### **4. Формування точок для Google API:**

* **Правило:** Початкова і кінцева точки маршруту залежать від напрямку:
  + "На роботу": Початкова точка — точка посадки першого пасажира. Кінцева точка — адреса роботи останнього пасажира.
  + "Додому": Початкова точка — адреса роботи першого пасажира. Кінцева точка — точка висадки останнього пасажира.
* **Дії:**
  + Сформувати масив координат, де перша і остання точки відповідають вказаним правилам.
  + Включити проміжні точки як waypoints.

#### **5. Відправка даних на Google Directions API:**

* **Дії:**
  + Відправити зібрані точки (початкова, проміжні та кінцева) на Google Directions API.
  + Отримати маршрут, включаючи:
    - Загальну тривалість.
    - Відстань.
    - Інструкції для водія.
  + Якщо маршрут перевищує обмеження, показати попередження.

#### **6. Візуалізація попереджень:**

* Попередження відображаються у вигляді:
  + Спливаючих повідомлень.
  + Червоних рамок навколо проблемних полів.
  + Текстових підказок із детальними поясненнями.

#### **Резюме:**

1. Виконується перевірка напрямків у заявках.
2. Перевіряється загальна тривалість маршруту.
3. Гарантується, що час прибуття на роботу відповідає найранішій заявці.
4. Формуються коректні точки для запиту до Google Directions API.
5. Користувач отримує візуальне попередження у разі порушення будь-якого з правил.