

КЛІЄНТ-СЕРВЕРНА ТЕХНОЛОГІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ВИБОРУ МАЛОТОННАЖНОЇ ВАНТАЖІВКИ

Юнашев Д. С.¹, Ільге І. Г.¹, Савчук Б. Є.²

Харківський національний автомобільно-дорожній університет, Харків

²Львівський національний університет природокористування, Львів

Інформаційні системи, пов'язані з транспортуванням вантажів, стали важливою частиною забезпечення транспортних компаній актуальною інформацією про товари та замовлення. Проте є також потреби малих і середніх підприємств та приватних клієнтів, які переважно користуються послугами з перевезення малотоннажних вантажів.

Малотоннажні автомобілі зазвичай є найпоширенішими та найкомпактнішими легкими вантажівками [1-3]. Більшість легких вантажівок і фургонів, вироблених у світі, належать до класу N1. Це означає, що загальна маса не повинна перевищувати 3,5 тонни (до 4,54 тонни в США), а вантажопідйомність може досягати 1,5-1,8 тонни.

Великі компанії зазвичай не хочуть виконувати такі дрібні замовлення, тому потенційні клієнти витрачають багато часу на пошук оголошень на десятках сайтів. Тому існує потреба у створенні онлайн-сервісу, який поєднав би пропозиції власників невеликих автомобілів із потребами клієнтів.

При розробці клієнт-серверної технології інформаційних систем важливу роль відіграють сервери баз даних, які зберігають і обробляють інформацію. Вимоги до серверів баз даних особливо високі: обладнання повинне відповідати стандартам безпеки та відмовостійкості та мати належну продуктивність і захист даних. Це залежить від того, наскільки швидко та ефективно програма отримує, обробляє, передає та захищає дані. Всього цього можна досягти за допомогою правильно підібраної СУБД.

На ринку СУБД представлено понад 300 інструментів. [4] Що стосується серверного обладнання існує ще більше можливостей. Тому вибір конфігурації апаратного і програмного забезпечення може бути складним. Враховуючи те, що

інформація, яка буде зберігатись в БД має, в основному, структуровану форму та табличний вигляд, було вирішено зупинити вибір на реляційній базі SQL.

При розробці інформаційних систем, особливо веб-сайтів, потрібен засіб відображення та тестування продукту. Розробники повинні розуміти, як виглядає макет і як працюють різні функції та сценарії. Однак придбання хостингового простору для цього коштує дорого. Крім того, сайт, який розробляється, має бути ізольованим від зовнішнього світу.

Тому рекомендується розробити та запустити веб-сайт на своєму комп'ютері з метою тестування. Для цього створені спеціальні програмні пакети (локальні сервери або веб-сервери). Вони містять те саме програмне забезпечення, яке встановлено на хостингу.

Локальні веб-сервери також відомі як платформи WAMP, які беруть першу літеру назви програми на сервері: А – Apache, М – MySQL, Р – PHP. Буква W означає, що платформа встановлена на Windows. На комп'ютері, де зберігаються файли сайту, встановлюється локальний сервер. Тоді до сторінок веб-сайту можна отримати доступ через браузер, подібно до Інтернету, за винятком того, що в URL-адресі вказується «localhost».

Різновидів платформ WAMP створено чимало [5]. Серед них був обраний ХАМРР, локальний сервер з повним набором модулів для серйозної роботи. Переваги ХАМРР: кросплатформенність; окрім PHP, ця збірка також містить Perl; безкоштовний і відкритий код; проста установка; зручний інтерфейс.

Невеликий недолік полягає в тому, що ХАМРР трохи повільний, але це можна компенсувати правильним налаштуванням веб-сервера. Ще один незначний мінус - інтерфейс користувача тільки англійською мовою.

Клієнтську частину програми можна розробити для різних операційних систем або веб-сайтів в Інтернеті. Для цього добре підходять такі мови програмування, як Python і C# [6-7]. Після порівняння мов, обране простіше рішення C# та надійний і перевірений віконний інтерфейс WinForms. Його ключовою особливістю є можливість напівавтоматично створювати програмний код інтерфейсу за допомогою

простих графічних робочих елементів, таких як вікна та кнопки. Після цього код потрібно лише трохи змінити в залежності від завдання розробника.

Щоб безпосередньо реалізувати систему, спочатку розробляють інфологічну, даталогічну та логічну моделі бази даних. Відповідні таблиці та зв'язки між таблицями створюються в СУБД MySQL. Підключити свою базу даних до C# можна за допомогою бібліотеки MySql.Data. Також розробляється графічний інтерфейс користувача, щоб полегшити використання системи.

Література:

1. Типи і види вантажних автомобілів. [Он-лайн]. Доступно: <https://specmash.org.ua/article/tipi-i-vidi-vantazhnih-avtomobiliv>
2. Класифікація вантажних автомобілів. [Он-лайн]. Доступно: <https://wiki.transinfo.by/klassifikaciya-gruzovih-avtomobilei/>
3. Малотоннажні вантажівки. [Он-лайн]. Доступно: <https://autoline.ua/-/malotonnazhni-vantazhivki--c1660>
4. Типи баз даних: особливості, відмінності та приклади. [Он-лайн]. Доступно: <https://dou.ua/lenta/articles/types-of-databases/>
5. Види популярних локальних серверів. [Он-лайн]. Доступно: <https://armedsoft.com/ua/blog/vydy-populyarnyh-lokalnyy-serveriv>
6. Порівняльний аналіз найпопулярніших мов програмування. [Он-лайн]. Доступно: <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2020/paper/download/10490/8923>
7. Що обрати: C# або Python. [Он-лайн]. Доступно: <https://foxminded.ua/ru/sergej-nemchinskij-chto-vybrat-v-2022-godu-c-ili-python/>