

Технічні науки

УДК 004.942

Шаричев Олег Романович

студент Інституту комп'ютерних систем

Одеського національного політехнічного університету

Шарычев Олег Романович

студент Института компьютерных систем

Одесского национального политехнического университета

Sharychev Oleh

Student of the Institute of Computer Systems

Odessa National Polytechnic University

Науковий керівник:

Мартинюк Олександр Миколайович

кандидат технічних наук, доцент кафедри КІСМ

Одеський національний політехнічний університет

ДОСЛІДЖЕННЯ МОДЕЛІ ВЕБ-СЕРВІСУ ДЛЯ ЗАБЕЗПЕЧЕННЯ

ОРЕНДИ АВТОМОБІЛІВ

ИССЛЕДОВАНИЕ МОДЕЛИ ВЕБ-СЕРВИСА ДЛЯ ОБЕСПЕЧЕНИЯ

АРЕНДЫ АВТОМОБИЛЕЙ

RESEARCH OF THE WEB SERVICE MODEL FOR CAR RENTAL

Анотація. У дослідженні розглянуто аналоги до розроблюваного веб-сервісу – виділено їх переваги та недоліки й зроблено висновок, які функції потрібно додати до веб-сервісу. Побудовані UML діаграми прецедентів, компонентів та послідовностей, які показують основні запити та відповіді між клієнтом, сервером та базою даних, виконано аналіз цих діаграм та розрахунок верхньої оцінки обчислювальної складності їх реалізації. Виконано побудову мережі Петрі із графом

досяжних маркувань, виконано аналіз та моделювання мережі, за результатами якого зроблено висновок щодо правильності побудованої на попередньому етапі проектування структури. Особливість моделі полягає у введенні позицій, переходів та зв'язків, що забезпечують можливість користувачам оплачувати оренду онлайн та використовувати карту місцевості для перегляду місцезнаходження автомобілю – це виділяє веб-сервіс серед аналогів. Також введено позиції, переходи та зв'язки, які забезпечують валідацію форм на клієнтській стороні веб-сервісу. Побудована загальна структурна схема, яка демонструє основні потоки даних на сервері та клієнті, а також між сервером та клієнтом. Виконано побудову основних граф-схем алгоритмів, які показують новизну розроблюваного веб-сервісу.

Ключові слова: веб-сервіс, клієнт-серверна програмна система, клієнт, сервер, мережа Петрі.

Анотація. В исследовании рассмотрены аналоги разрабатываемого веб-сервиса – выделены их преимущества и недостатки и сделан вывод, какие функции нужно добавить в веб-сервис. Построены UML диаграммы прецедентов, компонентов и последовательностей, которые показывают основные запросы и ответы между клиентом, сервером и базой данных, выполнен анализ этих диаграмм и расчет верхней оценки вычислительной сложности их реализации. Выполнено построение сети Петри с графом достижимых маркировок, выполнены анализ и моделирование сети, по результатам которого сделан вывод о правильности построенной на предыдущем этапе проектирования структуры. Особенность модели заключается во введении позиций, переходов и связей, обеспечивающих возможность пользователям оплачивать аренду онлайн и использовать карту местности для просмотра местонахождения автомобиля – это выделяет

веб-сервис среди аналогов. Также введено позиции, переходы и связи, которые обеспечивают валидацию форм на клиентской стороне веб-сервиса. Построена общая структурная схема, на которой представлены основные потоки данных на сервере и клиенте, а также между сервером и клиентом. Выполнено построение основных граф-схем алгоритмов, которые показывают новизну разрабатываемого веб-сервиса.

Ключевые слова: *веб-сервис, клиент-серверная программная система, клиент, сервер, сеть Петри.*

Summary. *The study examines the analogues of the developed web service - highlights their advantages and disadvantages and concludes which functions should be added to the web service: UML diagrams of precedents, components and sequences, which show the main requests and responses between client, server and database analysis of these diagrams and the calculation of the upper estimate of the computational complexity of their implementation. The construction of a Petri net with a graph of achievable markings was performed, the analysis and modeling of the network was performed, based on the results of which a conclusion was made about the correctness of the structure constructed at the previous design stage. The feature of the model is the introduction of positions, conversions and connections that allow users to pay rent online and use the map to view the location of the car - this distinguishes the web service among analogues. Positions, transitions, and links have also been introduced to validate forms on the client side of the web service, which increases efficiency. A general block diagram is constructed, which demonstrates the main data flows on the server and the client, as well as between the server and the client. The construction of the basic graph-schemes of algorithms which show novelty of the developed web service is executed.*

Key words: *web-service, client-server software system, client, server, Petri net.*

Постановка проблеми. На сьогоднішній день проблема стрімкого зростання кількості автомобілів у великих містах стає все більш актуальною. Тому все більше людей починають віддавати перевагу оренді автомобіля, ніж придбанню власного автомобілю. Це пов’язано із браком місць для паркування власного автомобілю у великих містах. Також, одною з основних причин того, що люди віддають перевагу орендованим автомобілям, є відсутність витрат на технічне обслуговування транспортного засобу. Від орендатора автомобілю вимагається тільки своєчасна оплата оренди та оплата пального для автомобілю.

Аналіз останніх досліджень та публікацій. Найбільш подібним аналогом до розроблюваного веб-сервісу є сервіс оренди автомобілів «DobaCar».

Із переваг даного сервісу можна виділити наступні: наявність функціоналу для створення для кожного оголошення календар оренди, який буде відображати доступні та недоступні дні для оренди; наявність функціоналу для перевірки особи за допомогою банківської системи; наявність функціоналу відгуків для кожного користувача; наявність функціоналу, який дозволяє орендаторам залишати заявки до кожного оголошення й орендодавцю підтверджувати найбільш привабливі заявки [1, с. 250].

Також можна виділити наступні недоліки сервісу «DobaCar»: відсутність функціоналу для відправки приватних повідомлень; відсутність функціоналу для додавання коментарів до оголошень про оренду; відсутність функціоналу для онлайн оплати оренди; відсутність функціоналу для відображення місцезнаходження автомобіля на карті [1, с. 251].

З аналогів розроблюваного сервісу також можна виділити сервіс «Booking.com». Даний сервіс у першу чергу призначений для замовлення готельних номерів, але у ньому є функціонал для оренди автомобілів.

Можна виділити наступні переваги цього сервісу: наявність функціоналу для відображення місцезнаходження автомобіля на карті; наявність функціоналу для пошуку автомобілів за класом; наявність функціоналу для оплати оренди онлайн; наявність функціоналу для бронювання автомобілю заздалегідь [2, с. 314].

Недоліки сервісу «Booking.com»: відсутність функціоналу для забезпечення можливості приватним особам виставляти власні автомобілі для оренди; відсутність функціоналу для відправки та отримання приватних повідомлень [2, с. 314].

Згідно з оглядом існуючих аналогів розроблюваного веб-сервісу, можна зробити висновок, що до його функціоналу необхідно додати наступний функціонал, який буде забезпечувати: можливість онлайн оплати оренди автомобіля; можливість користувачам сервісу відправляти та отримувати приватні повідомлення; можливість побачити місцезнаходження автомобілю на карті; можливість користувачам додавати коментарі до кожного оголошення; можливість орендаторам залишати заявки на оренду у кожному оголошенні, а орендодавцям підтверджувати їх.

Відокремлення невирішених раніше частин загальної проблеми.

У мережі Інтернет на даний момент існує велика кількість різних сервісів, які забезпечують оренду автомобілів. Але всі ці сервіси не надають можливості приватним особам виставляти власні автомобілі для здавання їх у оренду. Тому проблема створення сервісу оренди автомобілів, який би дозволяв виставляти приватним особам власні автомобілі для здавання їх у оренду, є відкритою.

Мета та задачі статті. Мета полягає у дослідженні архітектури веб-сервісу для забезпечення оренди автомобілів. Основні задачі: дослідження аналогів, проектування архітектури веб-сервісу, створення моделі, реалізації та проведення експерименту.

Розробка структури та моделі. Діаграма компонентів веб-сервісу представлена на рисунку 1.

Веб-сервіс складається з трьох основних компонентів: «SQL Server» - сервер з базою даних, яка зберігає у собі всю необхідну інформацію для функціонування сервісу, «HTTP Server» - сервер, який забезпечує обробку запитів клієнта, та формує відповіді, має зв'язок з базою даних, та за потреби звертається до неї за потрібною інформацією, «Client» - клієнтська частина веб-сервісу, забезпечує комунікацію з клієнтом, містить графічний інтерфейс користувача, виконує валідацію форм, які заповнює клієнт [3, с. 110].

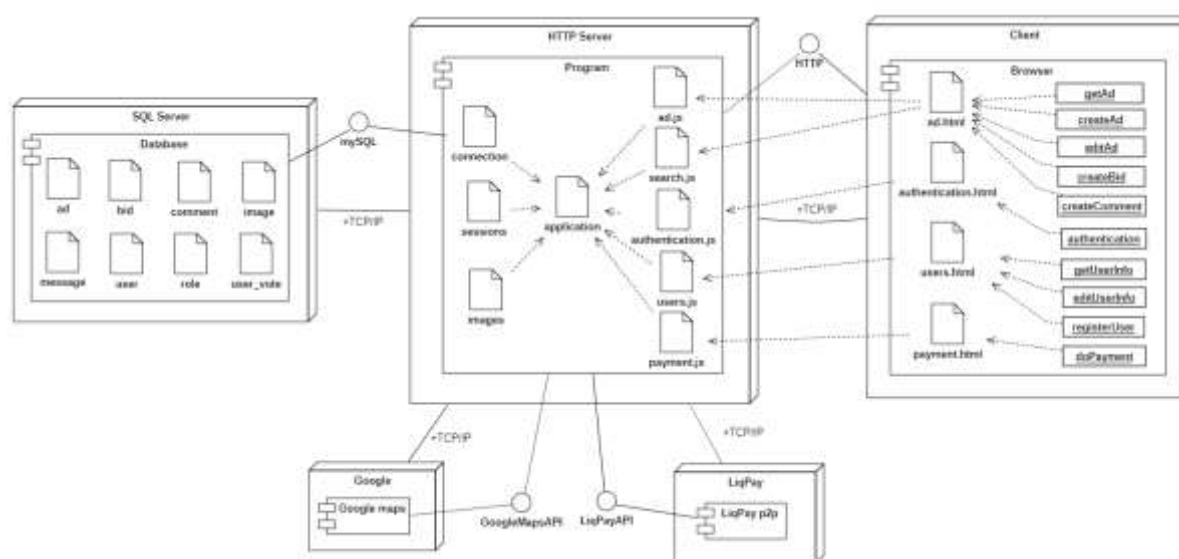


Рис. 1. Діаграма компонентів

Джерело: розробка автора

Також на рисунку представлені компоненти «Google» та «LiqPay» - перший забезпечує роботу з картами, на яких буде представлено місцезнаходження автомобілів, за допомогою «GoogleMapsAPI» [4, с. 470],

другий компонент забезпечує оплату оренди онлайн за допомогою «LiqPayAPI» [4, с. 550]. Компонент «SQL Server» містить такі програмні компоненти: ad, bid, comment, image, message, user, role, user_vote, які є таблицями у реляційній базі даних. Компонент «HTTP Server» містить такі програмні компоненти: connection, sessions, images, application, ad, search, authentication, users, payment, які є виконуваними файлами програми. Компонент «Client» містить такі програмні компоненти: ad, authentication, users, payment, які є файлами, що містять графічний інтерфейс користувача та обробники запитів користувача.

Мережу Петрі, представлену на рисунку 2, можна поділити на 5 основних складових: генератор запитів, який умовно є клієнтом веб-сервісу, який виконує будь які дії, тобто відправляє запити; клієнтська частина веб-сервісу – приймає запити від клієнта та проводить їх первинну обробку, наприклад, проводить верифікацію форм.

Також мережа містить частину, яка відповідає за відображення (рендер) об'єктів на екрані користувача; серверна частина веб-сервісу – проводить обробку запитів, які надійшли від клієнтської частини, робить запити до бази даних, оброблює відповіді від бази даних й формує відповіді до клієнтської частини веб-сервісу; база даних – отримує запити від серверної частини веб-сервісу, оброблює їх та формує відповіді до серверної частини; вершини для збирання статистики – збирають дані, отримані під час роботи моделі, такі як: мінімальний, максимальний, середній, сумарний час обробки запитів та кількість оброблених запитів.

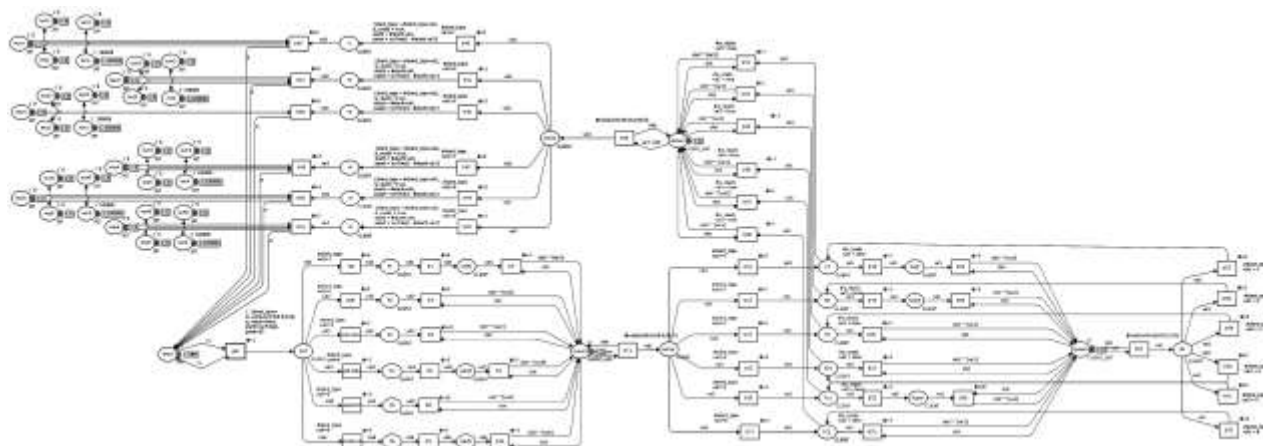


Рис. 2. Мережа Петрі

Джерело: розробка автора

Формальне подання мережі Петрі виглядає наступним чином: [5, с. 20]

$$C = (P, T, E),$$

де P – непушта кінцева множина позицій мережі,

T – непушта кінцева множина переходів,

$E = (P \times T) \cup (T \times P)$ – відношення інцендентності позицій та переходів (множина дуг мережі).

Для отриманої мережі:

$$P = \{cl, sort, h_{1...12}, valid_{1...2}, queue_{1...3}, server, hash_{1,2}, liqpay, db, r_{1...6}\}.$$

$$T = \{gen, reg, auth, ads view, edit data, payment, create ad, tr_{1...51}\}.$$

Особливість полягає у введенні позицій $P = \{liqpay, h_{12}\}$ й переходів $T = \{tr_{22}, tr_{26}, tr_{23}\}$ та зв'язків $E_1 = \{(liqpay \times tr_{26}), (h_{12} \times tr_{23})\}$ й $E_2 = \{(tr_{22} \times liqpay)\}$. Ці зв'язки введені задля забезпечення можливості користувачам оплачувати оренду онлайн та використовувати карту місцевості для перегляду місцезнаходження автомобілю.

Також введено позиції $P = \{valid_1, valid_2, valid_3\}$ й переходів $T = \{tr_2, tr_7, tr_{10}\}$ та зв'язків $E_1 = \{(valid_1 \times tr_2), (valid_2 \times tr_7), (valid_3 \times tr_{10})\}$. Ці зв'язки забезпечують валідацію форм на клієнтській стороні веб-сервісу.

Реалізація та апробація. Розроблена загальна структура веб-сервісу приведена на рисунку 3. Розроблюваний веб-сервіс має дволанкову клієнт-серверну архітектуру. Це означає, що клієнтська частина функціоналу сервісу виконується безпосередньо на пристрої користувача. Серверна частина, в свою чергу, відповідає за обробку та зберігання даних сервісу [6, с. 55]. Передача даних між серверною та клієнтською частиною відбувається за допомогою протоколу «HTTP».

Серверна частина розроблюваного сервісу поділяється на наступні модулі: «Node.js» – середовище виконання мови програмування «JavaScript»; «Express Framework» – веб-фреймворк, який надає набір функцій для роботи з веб-сервером; модуль автентифікації; модуль керування даними користувачів; модуль оплати; модуль керування даними оголошень; модуль пошуку; база даних – використовується для зберігання даних користувачів та даних оголошень; сховище сесій – використовується для зберігання даних про сесії авторизованих користувачів.

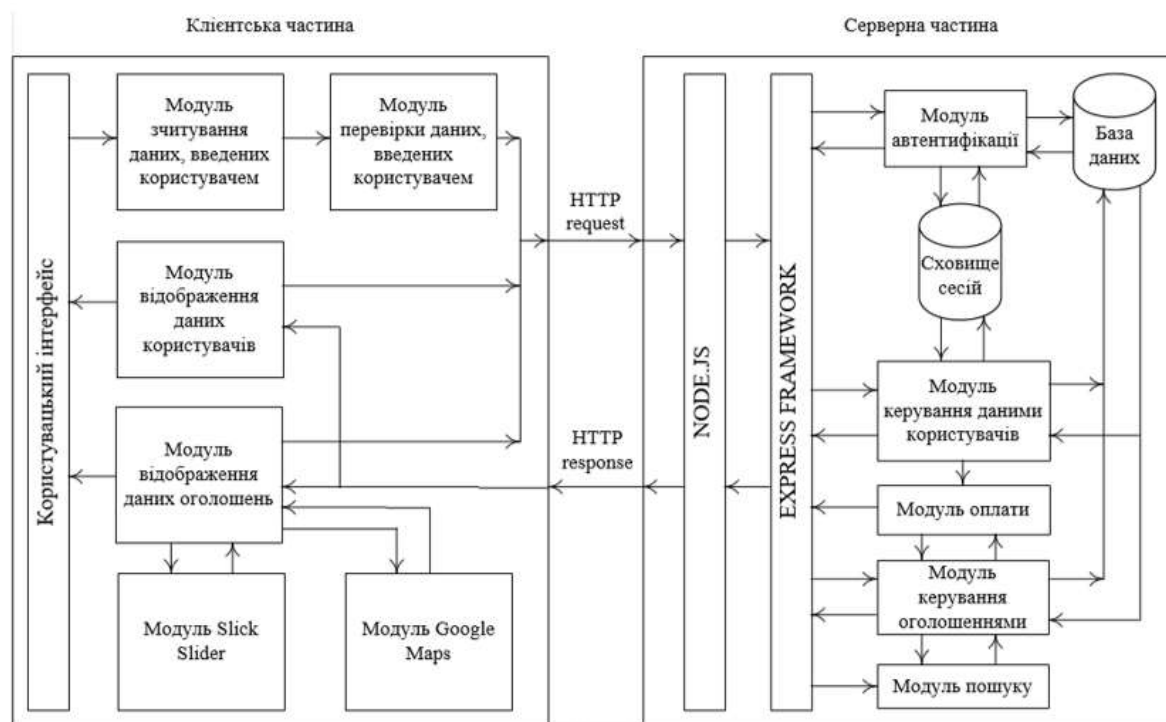


Рис. 3. Загальна структура

Джерело: розробка автора

Клієнтська частина розроблюваного веб-сервісу складається з наступних модулів: користувацький інтерфейс; модуль зчитування даних, введених користувачем; модуль перевірки даних, введених користувачем; модуль відображення даних користувачів; модуль відображення даних оголошень; модуль Slick Slider; модуль Google Maps.

Передача даних між серверною та клієнтською частиною сервісу виконується за допомогою протоколу передачі «HTTP». На серверній частині модуль «Node.js» відповідає за прийом та передачу даних, в свою чергу, на клієнтській частині за передачу та прийом даних від серверної частини відповідають модулі перевірки даних, введених користувачем, відображення даних користувачів та відображення даних оголошень.

Результати моделювання мережі Петрі наведені у таблиці 1. Моделювання проводилось у середовищі «CPNTools» із наступним апаратним забезпеченням: процесор – «Intel Core i5-8250U», обсяг оперативної пам'яті – 8 Гб, відеоадаптер – «Nvidia Geforce MX130».

Таблиця 1

Результати моделювання

К-сть кроків	Середній час 1, мс	Середній час 2, мс	Середній час 3, мс	Середній час 4, мс	Середній час 5, мс	Середній час 6, мс
5000	63	54	53	58	57	59
10000	62	55	53	57	56	59
15000	60	55	52	56	56	57
20000	61	56	53	56	54	56
25000	60	56	53	55	54	57

Джерело: розробка автора

Із результатів моделювання мережі Петрі можна зробити висновок, що середній час обслуговування користувача знаходиться в межах 53-63 мс, що є цілком допустимим. Найбільш складним для обробки запитом є перший тип запитів – це реєстрація користувача, тому що виконується

валідація форм та розрахунків хеш-суми SHA256 паролю користувача, яка потім заноситься до бази даних. Найбільш простим для виконання є запит третього типу – це відображення доступних на даний час оголошень, тому що він потребує тільки запиту до бази даних та візуалізацію отриманих даних.

Висновки і пропозиції. У дослідженні представлено загальну структуру, діаграми послідовності розроблюваного сервісу та їх аналіз, визначено оцінку обчислювальної складності діаграм. Наведено побудовану мережу Петрі із графом досяжних маркувань та проведено його аналіз. Також проведено моделювання мережі Петрі, з результатів якого можна зробити висновок, що розроблена структура відповідає вимогам з часу обслуговування користувачів.

Розроблюваний веб-сервіс надає користувачам можливість пройти реєстрацію, автентифікацію, вибирати ті автомобілі, які їм до вподоби, зі списку доступних й оплачувати орендну плату онлайн. Також користувачі можуть розміщувати власні автомобілі для здавання їх у оренду й комунікувати один з одним за допомогою коментарів або приватних повідомлень.

Розроблюваний веб-сервіс може застосовуватися як платформа для того, щоб здавати або брати автомобілі в оренду, як приватними особами, так і підприємствами та фірмами.

Література

1. Васкевич Девід Стратегії клієнт/сервер; Київ: Діалектика. Москва, 1996. 384 с.
2. Троелсен, Е. Мова програмування C# 5.0 й платформа .NET 4.5 / Э. Троелсен; Пер. з англ. Ю.Н. Артеменко. М.: Вільямс, 2013. 1312 с.
3. Резиг Джон JavaScript для професіоналів / Джон Резиг, Расс Фергюсон, Джон Пакстон. М.: Вільямс, 2015. 240 с.

4. Кузнецов М., Симдянов І., Голишев С. РНР 5. Практика створення Web-сайтів; БХВ-Петербург. М., 2017. 960 с.
5. Пітерсон Дж. Теорія мереж Петрі і моделювання систем. М.: Мир, 1984. 295 с.
6. Рассел Джессі Веб-сервер / Джессі Рассел. М.: Книга за вимогою, 2012. 100 с.