Специфікація програмного забезпечення

Вебсистема для продажу нових автомобілів

Software Requirements Specification

1.0

01.06.2025

Лось Мирослав Олександрович

Фатьянов Єгор Сергійович

**ІСТОРІЯ ЗМІН**

|  |  |  |  |
| --- | --- | --- | --- |
| **Дата** | **Опис** | **Автор** | **Коментарі** |
| 19.03.2025 | Створено пункти 1.1 - 1.3 | Фатьянов Єгор Сергійович |  |
| 20.03.2025 | Створено пункти 1.4 - 1.5, 2.1 - 2.3 | Лось Мирослав Олександрович |  |
| 21.03.2025 | Створено пункти 2.4, 2.5, 3.1 - 3.2 | Лось Мирослав Олександрович |  |
| 22.03.2025 | Створено пункти 3.3 - 3.5 | Фатьянов Єгор Сергійович |  |

**ЗАТВЕРДЖЕННЯ ДОКУМЕНТУ**

Наступну специфікацію вимог до програмного забезпечення було прийнято та схвалено:

|  |  |  |  |
| --- | --- | --- | --- |
| **Підпис** | **Друковане ім’я** | **Назва** | **Дата** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**ЗМІСТ**

Історія змін 2

Затвердження документу 2

1 Вступ 5

1.1 Огляд продукту 5

1.2 Мета 5

1.3 Межі 6

1.4 Посилання 6

1.5 Означення та абревіатури 6

2 Загальний опис 7

2.1 Перспективи продукту 7

2.2Функції продукту 7

2.3 Характеристики користувачів 8

2.1 Загальні обмеження 8

2.5 Припущення й залежності 8

3 Конкретні вимоги 9

3.1 Вимоги до зовнішніх інтерфейсів 9

3.1.1 Інтерфейс користувача 9

3.1.2 Апаратний інтерфейс 9

3.1.3 Програмний інтерфейс 9

3.1.4 Комунікаційний протокол 10

3.1.5 Обмеження пам’яті 10

3.1.6 Операції 10

3.1.7 Функції продукту 11

3.1.8 Припущення й залежності 11

3.2 Властивості програмного продукту 11

3.3 Атрибути програмного продукту 12

3.3.1 Надійність 12

3.3.2 Доступність 12

3.3.3 Безпека 12

3.3.4 Супроводжуваність 13

3.3.5 Переносимість 13

3.3.6 Продуктивність 13

3.4 Вимоги бази даних 14

3.5 Інші вимоги 15

4 Додаткові матеріали 17

# 1 ВСТУП

* 1. Огляд продукту

AutoHub — це вебзастосунок, що імітує онлайн-роботу автосалону, користувачі можуть переглядати каталог нових автомобілів, фільтрувати їх за характеристиками, переглядати детальну інформацію, а також зберігати обрані авто. Адміністратор має можливість додавати, редагувати та видаляти записи в каталозі, а також переглядати історію замовлень.

Проєкт реалізовано як навчальний, з акцентом на практичну реалізацію ключових аспектів веброзробки, роботи з базами даних та клієнтськими інтерфейсами. Він дозволяє відпрацювати навички побудови повноцінної SPA-архітектури без використання серверного бекенду.

AutoHub створено за принципами JAMstack з використанням Supabase як повноцінної backend-платформи, що охоплює зберігання, авторизацію та обробку запитів. Застосунок також враховує адаптивність інтерфейсу та підтримку ролей користувачів.

* 1. Мета

Основна мета — створення MVP (мінімально життєздатного продукту), який демонструє типові функції сучасного автосалону. Проєкт покликаний продемонструвати застосування CRUD-операцій, базової взаємодії з базою даних, роботи з хмарним сховищем і розмежування прав доступу.

Також важливою ціллю є відпрацювання навичок побудови адаптивного інтерфейсу та інтеграції з Supabase як хмарною платформою для backend-функціоналу. В межах проєкту реалізовано повну клієнтську логіку — від авторизації до динамічного оновлення інтерфейсу.

Розробка спрямована на практичне засвоєння фронтенд-архітектури, роботи з localStorage, реалізації валідації форм, conditional rendering та обробки помилок. AutoHub ілюструє повний цикл створення навчального вебпродукту з акцентом на UX та стабільність.

AutoHub ілюструє повний цикл створення навчального вебпродукту з акцентом на UX та стабільність.

Застосунок створено з урахуванням вимог масштабованості, що дозволяє у майбутньому розширити його функціонал без суттєвих змін у структурі. Проєкт також демонструє підхід до побудови повністю автономного клієнтського застосунку з інтеграцією зовнішньої хмарної платформи.

1.3 Межі

Проєкт обмежений наступними факторами:

* не підтримуються платіжні системи або інтеграції з банками;
* обмежена мовна підтримка (українська мова);
* всі перевірки ролей відбуваються на клієнтському рівні;
* всі дані зберігаються лише у Supabase, без локального резервного копіювання.

1.4 Посилання

1. <https://supabase.com> — хмарна платформа (БД, авторизація, storage)

2. <https://tailwindcss.com> — CSS-фреймворк для адаптивного UI

3. <https://nodejs.org> — серверна платформа, що використовується на етапі прототипування

4. <https://auto.ria.com> — приклад аналогічного функціоналу

1.5 Означення та абревіатури

* MVP (Minimum Viable Product) — найменша реалізація продукту, придатна до використання;
* CRUD (Create, Read, Update, Delete) — базові операції з даними;
* UI (User Interface) — інтерфейс користувача;
* SDK (Software Development Kit) — набір інструментів для розробки;

Supabase — open-source альтернатива Firebase, платформа для баз

даних, автентифікації та зберігання.

**2 ЗАГАЛЬНИЙ ОПИС**

2.1 Перспективи продукту

AutoHub створено як MVP для демонстрації функціоналу онлайн-автосалону, однак проєкт має потенціал до розширення у повноцінну SaaS-платформу з багатокористувацькою підтримкою, інтеграцією платіжних систем, особистими кабінетами користувачів, чатами з продавцями, автоматичною перевіркою VIN-кодів та імпортом даних з офіційних автодилерських баз.

У перспективі система може підтримувати:

* мультибрендові філії;
* особистий кабінет продавця та покупця;
* повноцінну реєстрацію/авторизацію з JWT або OAuth2;
* модуль аналітики та CRM для відстеження інтересів клієнтів.

2.2 Функції продукту

AutoHub надає користувачу наступний набір функцій:

* переглядкаталогу: головна сторінка відображає список усіх

наявних автомобілів;

* фільтрація: авто можна фільтрувати за брендом, моделлю, роком випуску, трансмісією, типом пального, типом приводу, кольором, ціною;
* детальнийперегляд: окрема сторінка для кожного автомобіля із зображеннями та технічними характеристиками;
* обране: можливість зберегти вибрані авто у локальному сховищі браузера;
* адміністрування: лише для адміністратора — додавання, редагування та видалення автомобілів;
* замовлення: адмін може бачити повідомлення-запити від користувачів із зазначенням обраного авто та дати створення.

2.3 Характеристики користувачів

У системі передбачено три типи користувачів:

* гість (незареєстрований): може переглядати каталог, фільтрувати авто, відкривати деталі;
* звичайнийкористувач: має доступ до всіх функцій як гість, а також може додавати авто до "Обраного", надсилати запити;
* адміністратор: окрім повного доступу до сайту, може керувати автомобілями (CRUD), переглядати таблицю замовлень, бачити приховані кнопки (наприклад, «Редагувати», «Видалити»).

Ідентифікаціяадміністратора реалізується через перевірку user.email у Supabase: якщо він співпадає з адресою, внесеною до списку адміністраторів, користувач отримує адмін-інтерфейс.

2.4 Загальні обмеження

* підтримується лише українська мова;
* платформа доступна тільки при наявності інтернет-з’єднання;
* перевірка прав доступу відбувається виключно на клієнті;
* обмеження на розмір зображень — до 5 МБ;
* пошук або сортування в таблиці замовлень наразі не реалізовано;
* дані не зберігаються локально поза Supabase.

2.5 Припущенняйзалежності

Продукт вимагає:

* сучасного браузера з підтримкою JavaScript (Chrome, Firefox, Edge);
* працездатності серверної частини Supabase (DB, Storage, Auth);
* наявності API ключів Supabase у .env файлі для локального запуску;
* коректної ініціалізації Supabase SDK на стороні клієнта;
* підключення Tailwind CSS (вручну або через CDN).

**3 КОНКРЕТНІ ВИМОГИ**

3.1Вимоги до зовнішніх інтерфейсів

3.1.1 Інтерфейс користувача

Інтерфейс AutoHub побудований з використанням Tailwind CSS [6] і відповідає принципам адаптивного дизайну. Основними елементами є картки автомобілів, фільтри, кнопки управління (для адміністратора), а також повідомлення про статус дій користувача.

* каталог має сіткове представлення з головним фото, назвою, брендом, роком, ціною.
* фільтраційна панель (зліва або зверху) дозволяє обирати параметри: бренд, модель, рік, тип пального, трансмісію, тип приводу, колір, діапазон цін.
* кожне авто відкривається на окремій сторінці, де відображаються: технічні характеристики, список зображень, кнопка "Додати до обраного".
* користувач бачить лише функції, доступні відповідно до ролі.
* інтерфейс сумісний із мобільними пристроями, має адаптивну верстку та підтримує темну тему.

3.1.2 Апаратний інтерфейс

AutoHub не залежить від специфічного апаратного забезпечення. Для доступу до функцій достатньо пристрою з веббраузером (ПК, ноутбук, планшет, смартфон). Продукт не вимагає встановлення локального ПЗ або додаткових апаратних драйверів.

3.1.3 Програмний інтерфейс

Вся логіка клієнтської взаємодії реалізована на JavaScript із використанням SDK Supabase. Основні API:

* supabase.from('cars').select() — отримання списку авто;
* supabase.from('cars').insert({...}) — додавання нового авто (admin only);
* supabase.from('cars').update({...}).eq('id', ...) — редагування;
* supabase.from('cars').delete().eq('id', ...) — видалення;
* supabase.storage.from('images').upload(...) — збереження фото;
* supabase.from('messages').insert({...}) — збереження повідомлення користувача.

Для аутентифікації/ідентифікації використовується .auth.getUser() та порівняння email з «адмінським».

3.1.4 Комунікаційний протокол

Взаємодія з Supabase та усіма зовнішніми сервісами здійснюється через захищений протокол HTTPS.

API-запити — формату JSON over HTTP/REST.

Всі запити проходять через CORS-безпеку Supabase, ключі авторизації передаються через .env або вбудовуються в supabaseClient.js.

3.1.5 Обмеження пам’яті

1. Обмеження розміру файлів при завантаженні фото: до 5 МБ на зображення.
2. Картинки зберігаються у Supabase Storage, посилання кешуються.
3. Локально зберігається лише список обраних авто в localStorage, обсяг — кілька КБ.
4. Кількість збережених карток в обраному необмежена технічно, але обмежується пам’яттю браузера.

3.1.6 Операції

Система реалізує наступні основні операції:

* create — додавання нового авто (тільки адміністратор);
* read — перегляд авто, фільтрація, деталізація;
* update — редагування параметрів авто;
* delete — видалення авто;
* favorite — додавання до обраного (тільки на стороні клієнта);
* message — створення заявки на авто (анонімно або з email);
* view\_orders — таблиця замовлень (лише адміністратор).

3.1.7 Функції продукту

Функції збігаються з тими, що описані в розділі 2.2, але подані у форматі технічного опису:

* каталогавто**:** GET /cars — всі дані з БД;
* фільтрація**:** фільтри реалізовані клієнтською логікою, параметри додаються до запиту;
* детальнийперегляд**:** GET /cars/:id;
* додаваннядо обраного**:** додавання ID у localStorage;
* замовлення**:** POST /messages — створення запису в таблиці;
* управлінняавто**:** POST/PUT/DELETE через SDK Supabase.

3.1.8 Припущення й залежності

Повторює загальні залежності (розділ 2.5), але уточнює:

* клієнтська частина повинна мати коректно ініціалізований SDK Supabase;
* усі ключі (anon key, project URL) зберігаються в .env;
* дані користувача визначаються через Supabase Auth (тільки email);
* жоден запит не працює без підключення до інтернету.

3.2 Властивості програмного продукту

Програмний продукт AutoHub характеризується сучасним дизайном, побудованим за принципами мінімалізму та інтуїтивності, що забезпечує комфортну взаємодію як для пересічних користувачів, так і для адміністраторів.

Його архітектура базується на хмарному середовищі Supabase, що дозволяє досягти високого рівня масштабованості, централізованого управління даними та гнучкого розгортання незалежно від фізичного середовища. Усі ці характеристики разом створюють надійне підґрунтя для подальшого розвитку системи та її розширення.

Користувацький інтерфейс реалізовано з урахуванням адаптивної верстки, що гарантує коректне відображення на різних типах пристроїв. Система має вбудовану перевірку ролей, яка визначає рівень доступу до функціоналу без потреби в окремому бекенді. Передбачена можливість швидкої інтеграції нових модулів без зміни базової архітектури. Увесь функціонал побудовано з використанням відкритих технологій, що дозволяє підтримувати та вдосконалювати проєкт у майбутньому.

3.3 Атрибути програмного продукту

Система AutoHub забезпечує базову надійність завдяки стабільній роботі Supabase як хмарного середовища. При стандартному використанні (перегляд, фільтрація, додавання до обраного) не виникає критичних помилок.

Архітектура без участі сервера дозволяє уникнути типових вразливостей, пов’язаних з backend-інфраструктурою. Вся взаємодія виконується через захищені канали з мінімальною затримкою та без необхідності повторних запитів.

3.3.1 Надійність

Всі запити до бази перевіряються на предмет помилок (наприклад, відсутності доступу, помилки завантаження фото), та в разі неуспішного виконання відображаються відповідні повідомлення користувачу. Адміністративні дії супроводжуються підтвердженням.

Для всіх критичних дій реалізовано обробку винятків та fallback-варіанти взаємодії з інтерфейсом. Це дозволяє уникнути втрати даних або непередбачених сценаріїв у роботі з контентом.

3.3.2 Доступність

AutoHub доступний 24/7 з будь-якого пристрою, що має інтернет та браузер. Supabase гарантує високу доступність інфраструктури за рахунок використання хмарної платформи PostgreSQL, CDN-сховищ і автоматичних бекапів.

Клієнтська частина не потребує встановлення і може бути розгорнута на будь-якому безкоштовному хостингу. Це забезпечує гнучкість розміщення та дає змогу підтримувати доступ до системи навіть у разі відмови одного з вузлів.

3.3.3 Безпека

Безпека реалізується на базовому рівні:

* усі запити до БД здійснюються через авторизований SDK Supabase, який використовує anon ключ;
* роль адміністратора визначається за email — перевірка доступна лише клієнту (у навчальній версії);
* критичні дії (видалення авто) потребують підтвердження;
* дані користувача, включно з email, не зберігаються в сторонніх сховищах або у cookies;
* фотографії із Supabase Storage мають публічні URL, але їхні шляхи не передбачають маніпуляцій.

3.3.4 Супроводжуваність

Проєкт має чітку структуру:

* вся логіка збережена у модульному JavaScript (catalog.js, supabaseClient.js);
* конфігурації зосереджені в одному місці (.env);
* кожна функція має зрозумілий опис, назву, параметри;
* код легко читається та може бути розширений, наприклад, через створення окремих модулів для фільтрів або деталей авто;
* застосунок підтримує зміну дизайну, теми, кольорів — завдяки Tailwind CSS.

3.3.5 Продуктивність

Основний акцент у розробці був зроблений на забезпечення високої продуктивності:

* завантаження головної сторінки з усіма авто відбувається менш ніж за 2 секунди (при нормальному інтернеті);
* запити до бази оптимізовані за допомогою .select() із обмеженням полів;
* фільтрація реалізується клієнтською логікою для миттєвої реакції без перезавантаження сторінки;
* зображення кешуються браузером після першого завантаження;
* використання Supabase як BaaS дозволяє уникати затримок у хостингу або БД.

3.4 Вимоги до бази даних

База даних вебсистеми AutoHub реалізована у Supabase на основі PostgreSQL. Вона містить основні сутності, необхідні для зберігання даних про автомобілі, взаємодію користувачів із каталогом і ведення історії замовлень. Архітектура БД забезпечує логічну зв’язність через зовнішні ключі та підтримує масштабування у майбутньому.

Основні таблиці:

* cars — інформація про автомобілі, доступні в автосалоні;
* messages — повідомлення або запити користувачів, пов’язані з конкретними авто;
* profiles — профілі користувачів, включно з email, ім’ям, роллю тощо;
* favorites — збережені авто, які користувач позначив як обрані.

Кожна таблиця містить атрибути, які визначають унікальність записів (первинні ключі) та зв’язки між сутностями (зовнішні ключі). Наприклад, таблиця messages містить зовнішній ключ car\_id, який посилається на cars.id, тим самим реалізуючи зв’язок типу «багато до одного».

Зв’язки між таблицями:

* messages.car\_id → cars.id — один автомобіль може мати багато повідомлень;
* favorites.user\_id → profiles.id, favorites.car\_id → cars.id — обране авто належить певному користувачу;
* profiles.id → auth.users.id — зв’язок між профілем користувача та авторизаційним записом Supabase Auth.

Особливості реалізації:

* використання типу UUID для унікальної ідентифікації записів;
* індексування первинних та зовнішніх ключів для покращення продуктивності запитів;
* каскадне видалення (ON DELETE CASCADE) для збереження цілісності даних при видаленні залежних записів;
* автоматичне заповнення полів created\_at при створенні запису.

Ключові структурні компоненти бази даних:

* cars — зберігає назву, бренд, модель, технічні характеристики (тип пального, трансмісія, привід, розгін, потужність тощо), масив зображень, колір, рік випуску та ціну;
* messages — фіксує звернення користувачів щодо конкретних авто (текст повідомлення, дата створення);
* favorites — реалізує many-to-many зв’язок між користувачами та авто, які вони зберегли у список вподобаних;
* profiles — слугує розширенням системи авторизації Supabase, зберігаючи роль користувача (admin/user) та контактні дані.

Резервне копіювання та розширення:

Таким чином, проєктна база даних AutoHub має чітку логіку побудови, забезпечує масштабованість та відповідає сучасним вимогам до вебзастосунків, що працюють із контентом на основі CRUD-операцій.

3.5 Інші вимоги

AutoHub повністю базується на Supabase як backend-рішенні. Supabase виконує роль бази даних (PostgreSQL), сховища зображень (Storage), а також системи автентифікації (обмежено). Це дозволяє уникнути створення власного сервера та зосередитися на фронтенд-логіці.

Основні сервіси Supabase, які використовуються:

* supabase database — таблиці cars, messages;
* supabase storage — зображення авто;
* supabase auth — перевірка user.email для ролі адміністратора.

Всі API-запити здійснюються через офіційний @supabase/supabase-js SDK . Ключ доступу (анонімний) зберігається в .env або безпосередньо в supabaseClient.js у навчальній версії.

Фото автомобілів завантажуються через клієнтський інтерфейс у Supabase Storage. До кожного авто може бути прив’язано кілька зображень (масив images).

Вимоги до зображень:

* максимальний розмір одного файлу: 5 МБ;
* рекомендовані формати: .jpg, .png, .webp;
* максимальна кількість зображень на авто: не обмежена функціонально, але рекомендовано до 10;
* зображення мають унікальні URL-шляхи, які зберігаються в полі images.

Зображення є публічними, тому важливо не використовувати персональні фото користувачів.

Ролі користувачів не зберігаються в базі. Замість цього, при ініціалізації сайту перевіряється email поточного користувача за допомогою supabase.auth.getUser(). Якщо email співпадає з email адміністратора (наприклад, "admin@autohub.com"), то на сторінці відображаються елементи керування: кнопки "Редагувати", "Видалити", "Додати авто", "Переглянути замовлення".

Всі інші користувачі працюють в режимі "гостьового" доступу.

Supabase забезпечує базове резервне копіювання даних. В рамках MVP-версії окремі копії не створюються. При необхідності може бути реалізовано:

* експорт даних з таблиці у форматі CSV;
* ручне збереження дампів бази через інтерфейс Supabase.

Архітектура AutoHub дозволяє масштабуватися без значних змін:

* фронтенд легко адаптується до багатомовності;
* нові сутності (бренди, категорії, типи комплектацій) можуть бути додані в окремі таблиці;

**4 ДОДАТКОВІ МАТЕРІАЛИ**

У додатках до пояснювальної записки наведено візуальні матеріали, що ілюструють ключові аспекти проєктування системи, а саме: діаграма прецедентів, яка демонструє основні сценарії взаємодії користувачів із системою; архітектурна діаграма системи, що відображає компоненти та взаємозв’язки між ними, а також ER-діаграма структури бази даних, яка показує логіку зберігання та зв’язки між сутностями (рис. 4.1–4.3).

Изображение выглядит как текст, диаграмма, рисунок, зарисовка

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 4.1 – UML Діаграма прицедентів

Изображение выглядит как текст, снимок экрана, диаграмма, число

Контент, сгенерированный ИИ, может содержать ошибки.Рисунок 4.2 – Схема архітектури системи

Изображение выглядит как текст, снимок экрана, диаграмма, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 4.3 – Схема бази даних