обчислювальні середовища, включаючи edge computing та fog computing, та на вивченні впливу контейнеризації на різні галузі індустрії.

Список використаних джерел

1. Технології автоматизації системних процесів: Навч. Посібник/Д.Ю. Меркотан, О.Я. Сова, О.О. Троцько, О.А. Симоненко, О.В. Гаман, О.Є. Степаненко, Г.Г. Мягких, В.П. Величко. Київ: ВІТІ, 2021. – 231 с.

2. Управління проєктами : навч. посібник. Львів : Львівський державний університет внутрішніх справ, 2021. 152 с

Робота виконана під науковим керівництвом PhD, доцента
ДЕСЯТКО А.М.

ЗАПРОВАДЖЕННЯ SPRING BOOT У ПЛАТФОРМІ ІТ-АКАДЕМІЇ

СКРИПНИК Д.О., 1 курс 2м група ФІТ ДТЕУ,
освітня програма «Інженерія програмного забезпечення»

У статті розглянуто впровадження технології Spring Boot у платформу ІТ Академії як стратегічний крок для модернізації освітнього процесу та підготовки майбутніх фахівців у галузі інформаційних технологій. Розглядаються переваги використання Spring Boot у контексті навчання, включаючи зручність розробки, швидкість створення прототипів, а також можливості інтеграції з існуючими технологіями.

The article considers the implementation of Spring Boot technology in the IT Academy platform as a strategic step for the modernization of the educational process and the training of future specialists in the field of information technologies. The benefits of using Spring Boot in an educational context are discussed, including ease of development, speed of prototyping, and integration with existing technologies.

Актуальність. У сучасному цифровому ландшафті, де технології змінюються на миттєвому рівні, освіта в галузі інформаційних технологій відіграє критичну роль у формуванні майбутнього кадрового потенціалу. Сучасні платформи навчання повинні бути готові швидко адаптуватися до цих змін, впроваджуючи інноваційні технології та методики, щоб гарантувати, що їх студенти будуть готові до викликів, які ставлять перед ними сучасні ринки праці.

Однією з таких перспективних інновацій є Spring Boot – фреймворк для розробки вебдодатків на мові програмування Java, який отримав широке визнання серед розробників програмного забезпечення. Впровадження Spring Boot у платформу ІТ Академії може відкрити нові горизонти для навчання та підготовки студентів. Ця ініціатива може допомогти у забезпеченні студентів необхідними навичками та знаннями, які відповідають сучасним вимогам ринку праці, а також зробити навчання більш захоплюючим та практично спрямованим.

Метою статті є дослідження впливу технології Spring Boot на платформі ІТ Академії на якість навчання студентів та їх підготовку до викликів сучасної ІТ-індустрії.

Об'єктом дослідження є об'єктом дослідження є процес впровадження технології Spring Boot у навчальну платформу ІТ Академії.

Предметом дослідження є практичні аспекти використання Spring Boot у навчальному процесі, включаючи переваги, можливості та потенційні виклики.

Аналіз попередніх досліджень. Дослідження в галузі використання технології Spring Boot у навчальних закладах показали, що її інтеграція може значно покращити якість навчання та підготовку студентів до реальних завдань розробки програмного забезпечення. Проте, виникають виклики, пов'язані з розробкою відповідних навчальних матеріалів та забезпеченням кваліфікованого викладацького складу. Дослідження цих аспектів є важливим для успішної інтеграції Spring Boot у навчальні програми.

Виклад основного матеріалу. Платформа IT-академії, аналогічно до Mate Academy, надає важливі можливості для навчання осіб, які бажають розвинути свої навички в IT-індустрії. Ці платформи функціонують як осередки навчання, пропонуючи різноманітні курси, семінари та ресурси з метою допомогти студентам розширити свій технічний багаж та підготуватися до успішної кар'єри в галузі інформаційних технологій. За даними на 2021 рік (для відображення ринку без впливу війни), цей сектор зріс на 36% лише за один рік, що суттєво збільшило кількість вакансій та попит на професіоналів у цій галузі.

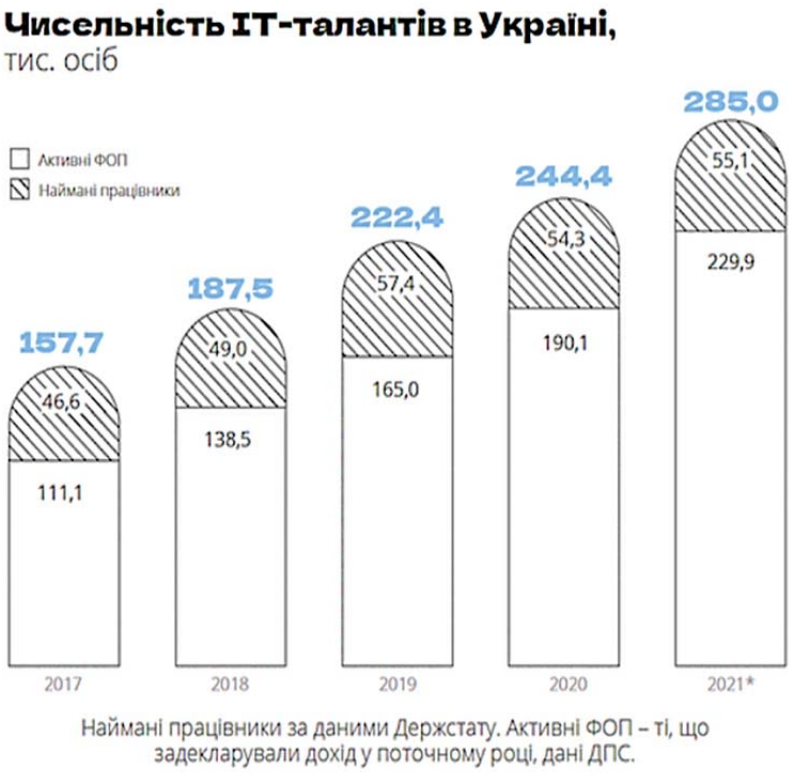

Рис. 1. Графік зросту кількості IT спеціалістів

Джерело: [1]

Оскільки попит на кваліфікованих IT-фахівців продовжує зростати, для таких платформ вкрай важливо постійно оновлюватись із новітніми технологіями та інструментами, щоб відповідати мінливим потребам галузі. Одним із способів підвищення продуктивності та ефективності цих платформ є впровадження Spring Boot.

Підвищення продуктивності платформи IT академії має важливе значення для забезпечення безперебійного навчання для студентів і викладачів. Spring Boot, потужна структура для створення додатків на основі Java, пропонує ряд переваг, які можуть значно підвищити продуктивність платформи. Використовуючи функції Spring Boot, розробники можуть створювати автономні програми виробничого рівня, які легко розгортати та підтримувати. Фреймворк має впевнений погляд на конфігурацію програми, спрощуючи процес розробки та дозволяючи швидше повторювати та розгортати нові функції. Цей спрощений підхід не тільки підвищує продуктивність, але й скорочує час розробки та витрати, пов'язані з обслуговуванням платформи.

Spring Boot – фреймворк для спрощення і прискорення розробки веб- і мікросервісних додатків на базі Spring Framework. Надає інструменти для швидкого старту проєкту, мінімізації рутини під час конфігурування та інтеграції з основним фреймворком Spring. Основне завдання Spring Boot – надання простого та ефективного способу розроблення застосунків, усунення ручного налаштування через стандартні конфігурації та автоматичне налаштування, що покращує зручність і швидкість розроблення.

Фреймворк був створений компанією Pivotal Software (нині належить VMware) у 2014 році, і відтоді став одним із найпопулярніших інструментів у світі Java-розробки [2].

Використання Spring Boot для створення вебдодатку на платформі IT академії надає безліч переваг, які сприяють ефективному навчанню та підготовці студентів до сучасних вимог ринку праці. Ось деякі з них:

- *Швидка розробка та простота*: Spring Boot надає простий та ефективний спосіб створення вебдодатків, що дозволяє студентам швидко розгортати та тестувати свої проєкти без зайвих складнощів.
- *Мінімальна конфігурація*: Spring Boot використовує конвенцію над конфігурацією, що дозволяє уникнути додаткової настройки та зосередитися на розробці функціоналу вебдодатку.
- *Широкі можливості інтеграції*: Spring Boot інтегрується з різноманітними технологіями та бібліотеками, що дозволяє студентам ознайомитися з різноманітними аспектами розробки вебдодатків та використовувати сучасні підходи та інструменти.
- *Підтримка мікросервісної архітектури*: Spring Boot надає потужні засоби для створення мікросервісних архітектур, що дозволяє студентам вивчати та використовувати сучасні підходи до побудови вебдодатків.
- *Розвинена екосистема та спільнота*: Spring Boot має широку та активну спільноту розробників, що надає доступ до безлічі документації, уроків та підтримки

Spring Boot представляє сучасний та ефективний спосіб розвитку платформи IT академії, який відповідає найкращим практикам і стандартам галузі. Використовуючи можливості Spring Boot, розробники можуть створювати надійні, масштабовані та безпечні програми, які відповідають вимогам динамічного IT-ландшафту. Фреймворк забезпечує вбудовану підтримку для різноманітних виробничих служб, дозволяючи розробникам зосередитися на створенні основних функцій, а не на управлінні інфраструктурою. Це зміщення фокусу дозволяє пришвидшити цикли розробки, швидше вивести на ринок нові функції та загальне покращення продуктивності платформи та взаємодії з користувачем.

Перш ніж інтегрувати Spring Boot у платформу IT академії, важливо провести ретельну оцінку поточної архітектури платформи. Розуміння існуючої структури, використовуваних технологій і залежностей має вирішальне значення для визначення того, як Spring Boot може легко вписатися в екосистему. Аналізуючи поточну архітектуру платформи, розробники можуть визначити потенційні області, де Spring Boot можна використовувати для підвищення продуктивності, масштабованості та зручності обслуговування. Ця фаза оцінювання служить основою для успішного процесу інтеграції та закладає основу для раціоналізованої стратегії впровадження.

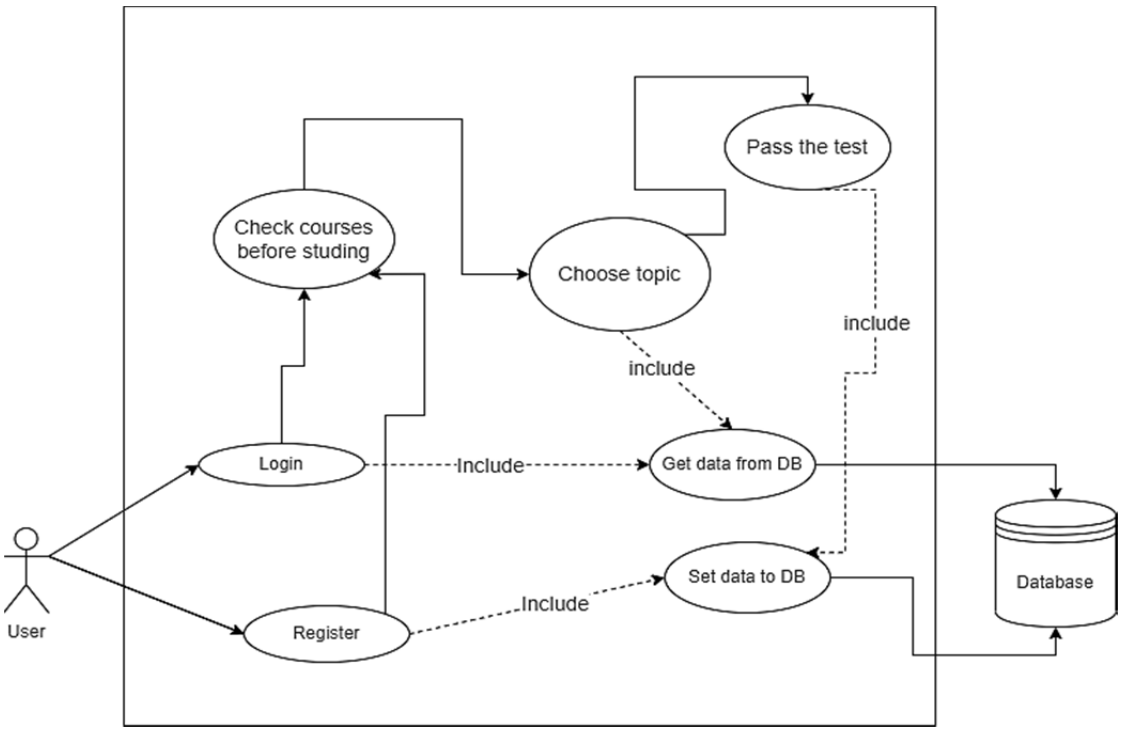

Рис. 2. Діаграма прецедентів, сценарій використання

Джерело: побудовано автором в середовищі draw.io

Після завершення оцінки поточної архітектури платформи наступним кроком є – визначення конкретних областей у платформі IT академії, де впровадження Spring Boot може принести максимальну користь. Spring Boot відомий тим, що спрощує розробку вебдодатків і мікросервісів, пропонуючи спрощений підхід до створення надійних і масштабованих програмних рішень з основих його можливостей можна виділити такі модулі:

- Spring Core: основа фреймворка, яка також забезпечує IoC та AOP;
- Spring Data: модуль який допомагає працювати з БД;
- Spring MVC: модуль для побудови вебдодатків;
- Spring Security: модуль для реалізації захисту додатку за допомогою аутентифікації та авторизації;
- Spring Test: модуль, що допомагає в написанні юніт тестів для додатку;
- Spring Integration: модуль, що спрощує інтеграцію додатку з іншими;
- Spring Cloud: модуль за допомогою якого можна працювати з хмарними сервісами.

Усі ці модулі можуть бути використані незалежно один від одного або в поєднанні з іншими, що дає можливість розробляти різноманітні додатки на основі Java з гнучкістю та можливістю розширення. Визначивши ключові області, які відповідають сильним сторонам Spring Boot, наприклад швидку розробку додатків, просте налаштування та підвищення продуктивності, розробники можуть стратегічно спланувати процес інтеграції, щоб отримати вигоду від можливостей інструменту.

Архітектура додатку на платформі IT Академії з використанням Spring Boot визначається відповідно до принципів мікросервісної архітектури. Кожен компонент системи функціонує як окремий мікросервіс, що дозволяє розробникам гнучко структурувати та розширювати функціональність. Взаємодія між мікросервісами здійснюється за допомогою механізмів синхронної та асинхронної комунікації. Кожен мікросервіс має свою власну базу даних та використовує інструменти моніторингу та керування для забезпечення оптимальної продуктивності та надійності системи. Архітектура додатку, побудована на основі Spring Boot, сприяє швидкому розвитку, легкій масштабованості та забезпечує високий рівень гнучкості та надійності.

Для моделювання структури програмного коду необхідно створити діаграму класів, що дозволить візуалізувати класи, змінні, методи, ще до початку розробки і на її основі побудувати платформу з усіма дотриманнями принципів об'єктно орієтованого програмування. Використання діаграм класів має кілька переваг:

- *Візуалізація структури:* Діаграми класів допомагають візуалізувати структуру системи, зробити її більш зрозумілою та легко сприйнятливою.
- *Виявлення проблем*: Діаграми класів дозволяють виявити можливі проблеми або недоліки в структурі системи на ранніх етапах розробки.
- *Підвищення продуктивності:* Використання діаграм класів дозволяє покращити комунікацію між розробниками та забезпечити однозначне розуміння структури системи.
- *Модульність та розширюваність*: Діаграми класів допомагають визначити модулі та їх взаємозв'язки, що сприяє покращенню модульності та розширюваності системи.
- *Документація коду*: Діаграми класів можуть служити як частина документації програмного коду, що полегшує розуміння системи для інших розробників або для майбутніх версій самої системи.
- *Ідентифікація класів і взаємозв'язків:* Під час створення діаграм класів, розробники аналізують систему, визначають класи та їх атрибути, а також зв'язки між класами. Цей процес допомагає виявити потреби в розробці нових класів або у внесенні змін до існуючих [3].

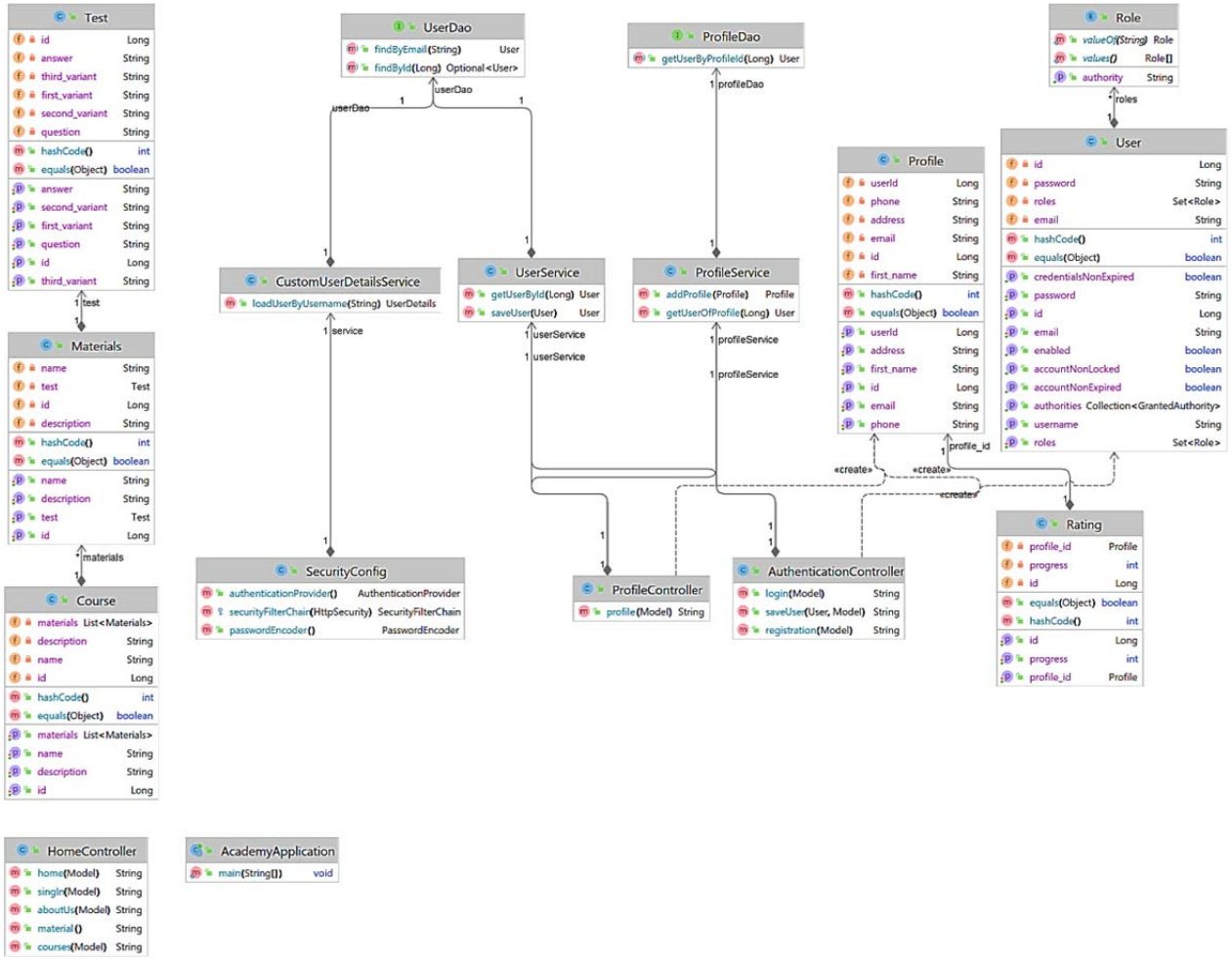

Рис. 3. Приклад діаграми класів для вебдодатку IT академії

Джерело: побудовано автором в середовищі Intellij IDEA

Кодування та інтеграція функцій Spring Boot у платформу IT академії передбачає використання надійних можливостей інфраструктури для розробки ефективних і масштабованих вебдодатків. Spring Boot спрощує процес розробки, надаючи такі готові функції, як вбудовані сервери, автоматичне налаштування та готові до виробництва показники. Розробники можуть використовувати анотації та утиліти Spring Boot для впровадження ключових функцій, зокрема:

- Створення RESTful API
- Постійність даних із Spring Data
- Функції безпеки з Spring Security
- Реалізація запланованих завдань за допомогою Spring Task Execution and Scheduling

Інтеграція цих функцій згуртовано покращує продуктивність і функціональність платформи, пропонуючи безперебійний доступ для учнів у IT академії.

Тестування та усунення несправностей впровадження Spring Boot на платформі IT академії є важливими для забезпечення надійності та ефективності розробленої програми. Spring Boot забезпечує повну підтримку тестування за допомогою таких модулів, як spring-boot-test, полегшуючи інтеграцію модульних тестів, інтеграційних тестів і наскрізних тестів. Дотримуючись найкращих практик у тестуванні, розробники можуть виявляти та вирішувати потенційні проблеми на ранніх етапах циклу розробки, сприяючи створенню стабільної та стійкої платформи для освітніх цілей. Постійне тестування та усунення несправностей не тільки перевіряють функціональність реалізованих функцій, але й сприяють загальній якості та продуктивності платформи IT академії на основі Spring Boot.

Фреймворк підтримує велику кількість бібліотек для тестування, ось найпопулярніші із них:

- *JUnit 5*: Відомий стандарт для модульного тестування Java-додатків.
- *Spring Test та Spring Boot Test*: Засоби для підтримки модульних та інтеграційних тестів для додатків Spring Boot.
- *AssertJ:* Бібліотека для зручних поточних тверджень.
- *Hamcrest:* Бібліотека для створення об’єктів-співставників (matchers), також відомих як обмеження або предикати.
- *Mockito:* Фреймворк для створення імітованих об’єктів у Java.
- *JSONassert:* Бібліотека для тверджень JSON.
- *JsonPath:* Бібліотека, яка дозволяє використовувати синтаксис XPath для навігації та вибору об’єктів у JSON.

Зазвичай ці загальновживані бібліотеки є корисними при написанні тестів. Однак, якщо вони не відповідають вашим потребам, ви можете додати додаткові залежності для тестування за вашим власним вибором [4].

Висновки. Підсумовуючи результати нашого дослідження, можна зазначити, що впровадження Spring Boot на платформі IT академії виявилося не лише доцільним, а й ефективним кроком у покращенні продуктивності та функціональності системи. Ретельне планування та підготовка перед впровадженням дозволили здійснити інтеграцію нового інструменту безперешкодно та з максимальною ефективністю.

Особлива увага до деталей під час процесу інтеграції та підтримки функціоналу Spring Boot позитивно позначилася на загальній продуктивності платформи. Використання цього інструменту стало ключовим етапом у підвищенні ефективності роботи, а також створило міцну основу для подальшого розвитку та вдосконалення.

Отже, можемо визначити, що впровадження Spring Boot на платформі IT академії виявилося успішним та перспективним кроком у розвитку системи, що сприятиме підвищенню якості навчання та покращенню практичного досвіду учнів.

Список використаних джерел

1. Констянит Васюк. Чому український IT-сектор став таким популярним // Режим доступу: http://surl.li/heuvu (останнє звернення 19.04.2024р.)
2. Як Spring Boot допомагає прискорити розробку додатків // Режим доступу: http://surl.li/sykxw (останнє звернення 19.04.2024р.)
3. Переваги використання діаграми класів // Режим доступу: http://surl.li/orbet (останнє звернення 19.04.2024р.)
4. Тестування програми на Spring Boot // Режим доступу: http://surl.li/sylyh (останнє звернення 19.04.2024р.)

Робота виконана під науковим керівництвом канд. екон. наук, доцента
ТИЩЕНКА Д.О.

ПРОГРАМНА КОМПОНЕНТА ІНФОРМАЦІЙНОЇ СИСТЕМИ ОБЛІКУ ЗАМОВЛЕНЬ НА СТАНЦІЇ АВТОМОБІЛЬНОГО ДЕТЕЙЛІНГУ

СЛІПЧУК В.О., 1 курс 2мз група ФІТ ДТЕУ,
освітня програма «Інженерія програмного забезпечення»

У статті розглянуто роль програмної компоненти в автоматизації обліку замовлень на станціях автомобільного детейлінгу. Розглядаються основні принципи її побудови та переваги використання у підприємницькій діяльності. Аналізуються важливі аспекти для підвищення ефективності роботи автомобільних детейлінгових центрів.

The article considers the role of software components in automating order management at automotive detailing stations. It discusses the fundamental principles of its construction and the advantages of its utilization in business operations. Important aspects for enhancing the efficiency of automotive detailing centers are analyzed.

Актуальність. Зростання популярності автомобільного детейлінгу та збільшення обсягів послуг у цій галузі ставлять перед станціями детейлінгу складні завдання з обліку замовлень. Пошук оптимальних шляхів оптимізації процесів прийому, обробки та виконання замовлень потребує впровадження ефективних програмних компонент систем обліку. Розробка програмних рішень для автоматизації обліку замовлень на станціях автомобільного детейлінгу має велике практичне значення для підвищення продуктивності та конкурентоспроможності підприємств цієї галузі.

«Car detailing» – це комплекс послуг, спрямованих на очищення автомобіля та його частин. Детейлінг набагато точніше та складніше звичайної мийки: кузов і салон розбираються на складові частини вручну. Іншими словами, детейлінг інколи порівнюють зі спа-процедурами для авто через очищення важкодоступних місць і дрібних деталей, в яких накопичуються бруд і пил.

У чому різниця між автомобільним детейлінгом та мийкою? Помити автомобіль може кожен, а ось послуга детейлінгу виконується досвідченими фахівцями з використанням правильних методів та спеціальних препаратів/очисних засобів, які не можуть пошкодити навіть делікатну оббивку і не залишають подряпин та пошкоджень на дорогому покритті. Після таких послуг ваш автомобіль виглядатиме і їздитиме, як новий – це і є головна перевага детейлінгу.