

РОЗРОБКА СТРУКТУРИ ЗАСТОСУНКУ ДЛЯ УПРАВЛІННЯ РЕСУРСАМИ КІНОТЕАТРУ ІЗ ВИКОРИСТАННЯМ JHIPSTER

Вінницький національний технічний університет

Анотація

Розроблено структуру монолітного веб-застосунку для управління ресурсами кінотеатру за допомогою технології JHipster 8.0. Система використовує архітектуру REST API із JWT-автентифікацією, кешування на основі Hibernate, підтримку багатомовності, а також інтеграцію Kafka для обробки повідомлень. Визначено основні сутності застосунку: фільм, зал, місце, сеанс, квиток, снік, склад, співробітник, розклад, продаж та клієнт. Реалізовано взаємозв'язки між сутностями, визначено DTO-шари та сервіси з використанням MapStruct. Виконано налаштування пагінації для підвищення продуктивності.

Ключові слова: Hipster, JWT, REST API, Kafka, кінотеатр.

Abstract

Developed a monolithic web application structure for cinema resource management using JHipster 8.0 technology. The system utilizes REST API architecture with JWT authentication, Hibernate-based caching, multilingual support, and Kafka integration for message handling. Defined main application entities: movie, hall, seat, screening, ticket, snack, inventory, employee, schedule, sale, and customer. Implemented entity relationships, DTO layers, and services using MapStruct. Configured pagination to improve performance.

Keywords: JHipster, JWT, REST API, Kafka, cinema.

Вступ

Сучасні кінотеатри потребують ефективних систем управління ресурсами, здатних інтегрувати різні аспекти діяльності: від продажу квитків і контролю запасів до управління персоналом та аналітики продажів. Використання сучасних фреймворків для розробки веб-застосунків дозволяє значно оптимізувати ці процеси, забезпечуючи високу продуктивність і надійність.

Результати дослідження

У процесі розробки структури застосунку для управління ресурсами кінотеатру використано сучасний підхід до створення програмного забезпечення на основі технології JHipster 8.0 [1]. Застосунок реалізований у формі монолітної архітектури з використанням REST API, що дозволяє ефективно обслуговувати веб-інтерфейс та мобільних клієнтів. Система використовує JWT (JSON Web Token) [2] автентифікацію, що забезпечує надійний механізм авторизації користувачів без необхідності зберігання сесій на сервері, що є суттєвою перевагою для масштабованості та продуктивності системи [3].

Структура застосунку визначена за допомогою файлу конфігурації JHipster, де чітко вказані параметри бази даних (H2 для розробки та PostgreSQL для продакшну), налаштовано кешування на рівні Hibernate [4] для оптимізації роботи з базою даних, забезпечено підтримку багатомовності (українська та англійська мови) та інтеграцію Apache Kafka [5] для ефективної обробки асинхронних подій. Для зручності розробки та підтримки додано підтримку Swagger [6] для генерації документації API.

Комплексна модель предметної області включає сутності, які детально відповідають бізнес-процесам кінотеатру. Сутність Movie містить інформацію про фільми (назва, опис, тривалість, дата виходу, рейтинг), що необхідно для формування афіші та каталогізації. Сутність Hall описує зали кінотеатру (назва, місткість), Seat – окремі місця у залі, включаючи номер, ряд та статус доступності. Screening представляє сеанси, що відбуваються у визначений час з певною ціною. Сутність Ticket відповідає за облік квитків з інформацією про дату купівлі, ціну та статус. Snack охоплює продукти, доступні для покупки, Inventory керує запасами товарів, Employee описує персонал кінотеатру,

Schedule відображає робочий графік співробітників, Sale забезпечує фінансовий облік продажів, а Customer містить інформацію про клієнтів для маркетингових та лояльнісних програм.

Предметна область розробленого застосунку повністю відповідає реальним бізнес-процесам кінотеатру та спрямована на ефективне вирішення задач управління ресурсами та взаємодії з клієнтами. Для точного моделювання процесів у застосунку визначено низку основних сутностей, кожна з яких має чітке призначення та відповідає окремій сфері діяльності кінотеатру.

Сутність Movie (Фільм) відображає реальні об'єкти, які кінотеатр демонструє своїм відвідувачам. Вона містить необхідні для клієнтів та адміністраторів атрибути: назву, опис, тривалість, дату виходу та рейтинг, що дає змогу формувати каталог фільмів та створювати ефективні рекламні кампанії.

Сутність Hall (Зал) відповідає реальним фізичним залам кінотеатру. Атрибути цієї сутності включають назву залу та місткість, що дозволяє керувати ресурсами та ефективно планувати використання приміщень.

Сутність Seat (Місце) описує кожне окреме місце в залі з деталізацією номеру, ряду та статусу (доступне, зарезервоване чи пошкоджене). Це дозволяє точно контролювати доступність місць та покращує досвід клієнтів під час бронювання.

Сутність Screening (Сеанс) моделює сеанси фільмів, які відбуваються в певний час у визначеному залі. Вона має атрибути дати та часу проведення сеансу, а також ціни, що є ключовими для планування та фінансового управління.

Сутність Ticket (Квиток) служить для обліку проданих квитків та включає інформацію про дату купівлі, ціну та статус квитка (придбаний, анульований, використаний). Це забезпечує чіткий облік продажів та контроль процесу відвідування сеансів.

Сутність Snack (Снеки) відповідає за управління продажем товарів харчування, доступних у кінотеатрі. Вона містить назву товару, ціну та кількість, що допомагає ефективно керувати продажами та запасами.

Сутність Inventory (Склад) слугує для контролю товарних запасів кінотеатру, включаючи назву товару, його кількість та рівень, при якому необхідно робити замовлення на поповнення.

Сутність Employee (Співробітник) дозволяє управляти інформацією про персонал кінотеатру, включаючи персональні дані, посаду та дату прийняття на роботу.

Сутність Schedule (Розклад) використовується для організації робочого графіка співробітників, включаючи дату та час початку й закінчення робочих змін.

Сутність Sale (Продаж) охоплює фінансовий аспект роботи кінотеатру, містячи дату та час продажу, а також загальну суму транзакції.

Сутність Customer (Клієнт) містить інформацію про відвідувачів кінотеатру, зокрема контактні дані та систему лояльності, що дозволяє ефективно проводити маркетингові кампанії та покращувати взаємодію з клієнтами.

Розроблену структуру застосунку зображено на рисунку 1.

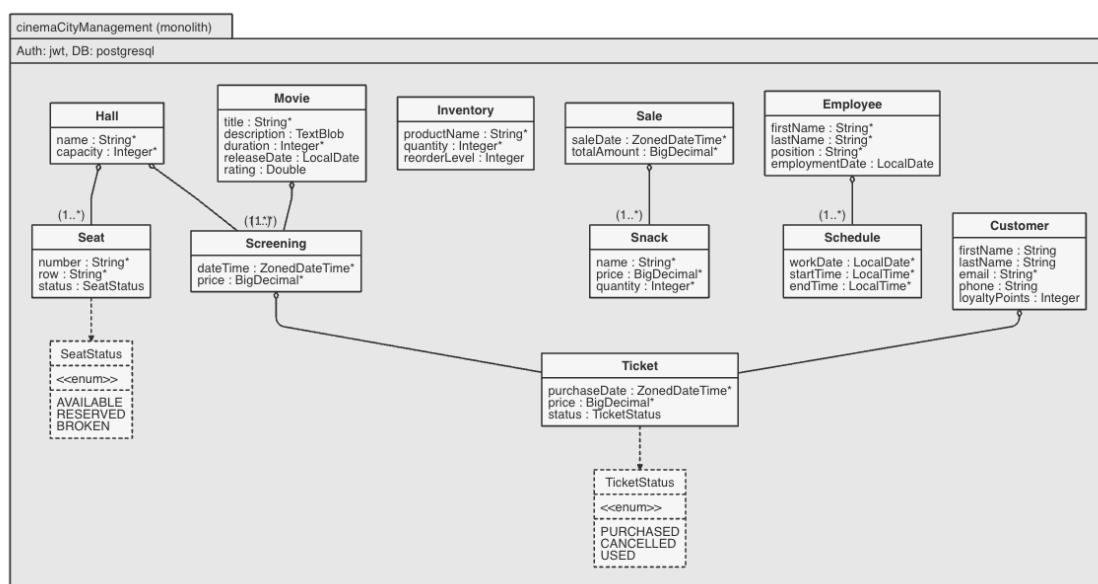


Рис. 1. Результат формування структури застосунку

Зв'язки між сутностями відображають реальні взаємодії в кінотеатрі та побудовані з використанням підходу OneToMany, який найкраще відповідає реальним бізнес-процесам та забезпечує ефективну організацію даних і високу продуктивність системи [7].

Зв'язки між сутностями побудовано з огляду на реальні бізнес-процеси. Використовується тип зв'язку OneToMany, оскільки один зал може мати багато місць та сеансів, один сеанс містить багато квитків, один клієнт може придбати кілька квитків, одна транзакція продажу може включати багато снєків, а один співробітник може мати багато графіків роботи. Такий підхід оптимальний для ефективної організації даних, зменшення дублювання та забезпечення високої продуктивності

Для покращення роботи застосунку створено шари DTO (Data Transfer Object) з використанням MapStruct [8], що дозволяє ефективно керувати передачами даних, а також зменшує навантаження на сервер та базу даних. Всі сервіси реалізовані у вигляді окремих класів, що робить код прозорим, легким у підтримці та розширенні. Для керування великими масивами інформації використовується пагінація, що значно покращує швидкодію та стабільність роботи системи.

Для подальшої розробки слід підготувати UML-діаграму класів, що зображено на рисунку 2.

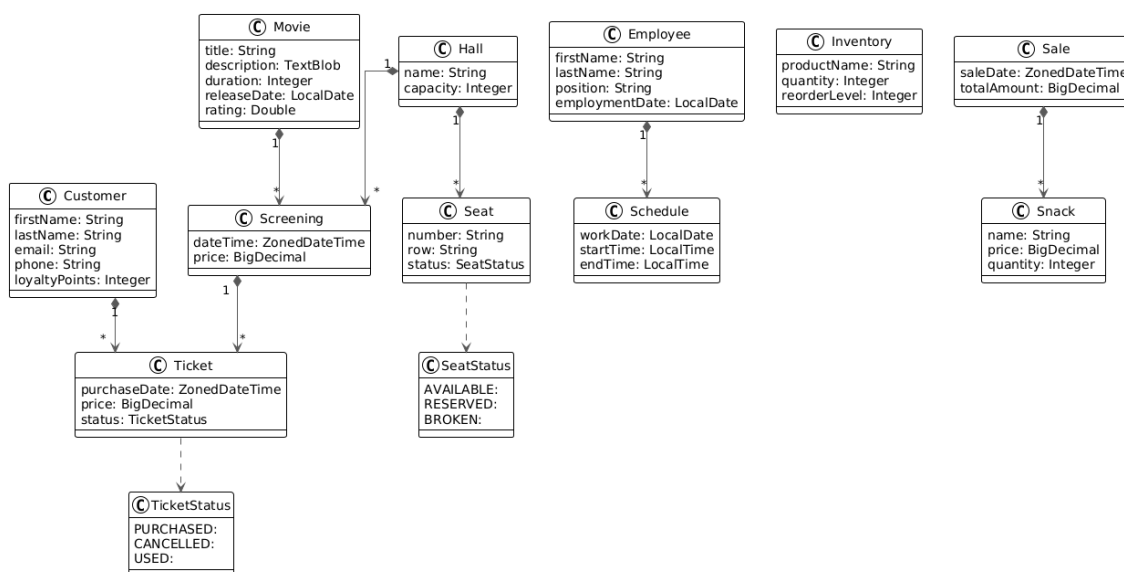


Рис. 2. Розроблена UML-діаграма класів для застосунку

Таким чином, розроблена структура застосунку на базі JHipster 8.0 дозволяє створити ефективне, масштабоване та зручне у використанні програмне забезпечення для управління ресурсами кінотеатру. В результаті запиту користувач отримує відповідь у форматі JSON, що містить згенерований особистий токен та токен для оновлення.

Висновки

У результаті дослідження розроблено ефективну структуру веб-застосунку для управління ресурсами кінотеатру з використанням сучасної технології JHipster 8.0. Застосована монолітна архітектура та REST API забезпечують стабільність, масштабованість і продуктивність. Використання JWT-технології гарантує надійний захист персональних даних користувачів та безпечний доступ до ресурсів системи. Інтеграція Kafka, багатомовність та оптимізація з використанням Hibernate сприяють покращенню досвіду користувачів та підвищують загальну ефективність роботи кінотеатру. Обрана структура сутностей та чітко визначені взаємозв'язки дозволяють повністю моделювати реальні бізнес-процеси та ефективно керувати усіма аспектами діяльності кінотеатру, що позитивно впливає на фінансові результати, якість обслуговування клієнтів та конкурентоспроможність на ринку.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. JHipster Documentation. [Електронний ресурс] – Режим доступу: <https://www.jhipster.tech/documentation-archive/v8.0.0/> (дата звернення: 20.03.2025).

2. JSON Web Token Introduction. Auth0. [Електронний ресурс] – Режим доступу: <https://auth0.com/docs/secure/tokens/json-web-tokens> (дата звернення: 20.03.2025).
3. Spring Boot Reference Guide. [Електронний ресурс] – Режим доступу: <https://docs.spring.io/spring-boot/docs/current/reference/html/> (дата звернення: 20.03.2025).
4. Hibernate ORM Documentation. [Електронний ресурс] – Режим доступу: <https://hibernate.org/orm/documentation/> (дата звернення: 20.03.2025).
5. Apache Kafka Documentation. [Електронний ресурс] – Режим доступу: <https://kafka.apache.org/documentation/> (дата звернення: 20.03.2025).
6. Swagger Documentation. [Електронний ресурс] – Режим доступу: <https://swagger.io/docs/> (дата звернення: 20.03.2025).
7. PostgreSQL Documentation. [Електронний ресурс] – Режим доступу: <https://www.postgresql.org/docs/> (дата звернення: 20.03.2025).
8. MapStruct Official Documentation. [Електронний ресурс] – Режим доступу: <https://mapstruct.org/documentation/stable/reference/html/> (дата звернення: 20.03.2025).

Слободиський Владислав Віталійович – студент групи ІСТ-21б, кафедра автоматизації та інтелектуальних інформаційних технологій, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м.Вінниця, e-mail: sloboda55@gmail.com

Богач Ілона Віталіївна – к.т.н., доцент кафедри автоматизації та інтелектуальних інформаційних технологій, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м.Вінниця, e-mail: ilona.bogach@gmail.com

Slobodyskyi Vladyslav Vitaliiovich – student of ICT-21b group, Department of Automation and Intelligent Information Technologies, Faculty of Intelligent Information Technology and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: sloboda55@gmail.com

Bogach Ilona Vitaliivna – Associate Professor of Automation and Intelligent Information Technologies Department, Faculty of Intelligent Information Technology and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: ilona.bogach@gmail.com