**СПЕЦИФІКАЦІЯ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ (SRS)**

Тема роботи: Інформаційна система для автоматизації діяльності кінотеатру  
**Версія:** 1.0

# 1 ВСТУП

Цей документ є специфікацією вимог до програмного забезпечення (Software Requirements Specification, SRS) для проєкту «Інформаційна система для автоматизації діяльності кінотеатру». Він детально описує функціональні та нефункціональні вимоги до системи, її архітектуру, характеристики та обмеження. Документ слугує основною угодою між замовником та розробником, визначаючи обсяг робіт та критерії приймання готового продукту.

## 1.1 Огляд продукту

Програмний продукт, що розробляється, є клієнт-серверним вебзастосунком, призначеним для комплексного обслуговування діяльності сучасного кінотеатру. Система складається з двох основних компонентів: публічної клієнтської частини (Frontend), що надає глядачам доступ до репертуару, розкладу сеансів та можливості онлайн-бронювання і купівлі квитків, та захищеної адміністративної панелі (Backend), яка надає персоналу кінотеатру інструменти для управління контентом, сеансами, кінозалами та моніторингу бронювань.

Система покликана автоматизувати ключові бізнес-процеси взаємодії з клієнтом, оптимізувати роботу адміністративного персоналу та підвищити конкурентоспроможність кінотеатру в умовах цифрової трансформації ринку розважальних послуг.

## 1.2 Мета

Основною метою проєкту є створення сучасного, продуктивного, надійного та масштабованого вебсайту, який забезпечує повний цикл взаємодії клієнта з послугами кінотеатру – від отримання інформації про фільми до купівлі квитка. Окрім цього, важливою ціллю є надання адміністрації кінотеатру ефективних та інтуїтивно зрозумілих інструментів для управління операційною діяльністю, що дозволить знизити навантаження на персонал та отримати доступ до базової аналітики для прийняття управлінських рішень.

## 1.3 Межі

Діяльність у межах цього проєкту чітко окреслена для забезпечення досягнення поставлених цілей у визначені терміни.

У межах проєкту реалізуються:

* Розробка публічної клієнтської частини (Frontend) у вигляді односторінкового застосунку (SPA).
* Розробка захищеної адміністративної панелі для управління контентом.
* Реалізація серверної частини (Backend) у вигляді REST API для обробки бізнес-логіки та взаємодії з базою даних.
* Проєктування та створення реляційної бази даних для зберігання всієї інформації системи.
* Інтеграція із зовнішнім платіжним шлюзом Stripe для реалізації функції онлайн-оплати.

Поза межами проєкту залишаються:

* Розробка нативних мобільних застосунків для платформ iOS та Android.
* Інтеграція з системами бухгалтерського обліку.
* Впровадження розширених програм лояльності, систем накопичення бонусів чи системи продажу супутніх товарів (напої, закуски).

## 1.4 Посилання

Основним джерелом інформації для цього документа є пояснювальна записка до комплексного курсового проєкту «Інформаційна система для автоматизації діяльності кінотеатру». Усі візуальні матеріали, зокрема UML-діаграми (прецедентів, діяльності, розгортання) та моделі даних (ER-діаграма, реляційна схема), наведені у відповідних розділах зазначеної курсової роботи.

## 1.5 Означення та абревіатури

API – Application Programming Interface (Програмний інтерфейс застосунку)

CRUD – Create, Read, Update, Delete (Створити, Прочитати, Оновити, Видалити)

CSS – Cascading Style Sheets (Каскадні таблиці стилів)

ДСТУ – Державний стандарт України

HTML – HyperText Markup Language (Мова гіпертекстової розмітки)

HTTP – HyperText Transfer Protocol (Протокол передачі гіпертексту)

JWT – JSON Web Token (Веб-токен формату JSON)

JSON – JavaScript Object Notation (Об’єктна нотація JavaScript)

REST – Representational State Transfer (Передача репрезентативного стану)

SPA – Single Page Application (Односторінковий застосунок)

SQL – Structured Query Language (Мова структурованих запитів)

SRS – Software Requirements Specification (Специфікація вимог до програмного забезпечення)

СУБД – Система управління базами даних

# 2 ЗАГАЛЬНИЙ ОПИС

## 2.1 Перспективи продукту

Продукт розробляється як самостійне програмне рішення, орієнтоване на кінотеатри будь-якого масштабу. Завдяки гнучкій клієнт-серверній архітектурі та стандартизованому API, система має значний потенціал для адаптації до суміжних галузей, де потрібне онлайн-бронювання місць за схемою, зокрема для театрів, концертних майданчиків та конференц-залів. Проєкт не є модифікацією існуючої системи, а створюється з нуля з урахуванням сучасних технологічних стеків та підходів до розробки.

## 2.2 Функції продукту

Система надає два основні набори функцій, що відповідають ролям користувачів.

Основні функції для глядача (кінцевого користувача):

* Перегляд афіші актуальних фільмів, детальної інформації про них (опис, трейлер, акторський склад) та розкладу сеансів.
* Інтерактивний вибір місць на схемі залу для конкретного сеансу.
* Онлайн-бронювання та оплата квитків за допомогою банківської картки.
* Реєстрація та авторизація в особистому кабінеті для перегляду історії замовлень та управління персональними даними.

Основні функції для адміністратора:

* Безпечна автентифікація для доступу до адміністративної панелі.
* Управління даними про фільми, кінозали та сеанси (операції CRUD).
* Базовий моніторинг активних бронювань та заповненості залів.
* Перегляд ключових показників роботи кінотеатру на інформаційній панелі (Dashboard).

## 2.3 Характеристики користувачів

Цільова аудиторія програмного продукту поділяється на дві основні групи:

1. Кінцеві користувачі (глядачі): Широка аудиторія віком переважно від 16 до 45 років, яка активно користується Інтернетом та мобільними пристроями для планування дозвілля. Від цієї групи не вимагається спеціальних технічних навичок, окрім базового досвіду користування сучасними вебсайтами.
2. Адміністративний персонал: Співробітники кінотеатру (адміністратори, менеджери), які є впевненими користувачами ПК та стандартних офісних програм. Передбачається, що персонал пройде базове навчання для роботи з адміністративною панеллю системи.

## 2.4 Загальні обмеження

* Для роботи з системою як з боку клієнта, так і з боку адміністратора, потрібне стабільне інтернет-з'єднання.
* Клієнтська частина системи вимагає використання сучасного веббраузера з підтримкою JavaScript (ES6+) та CSS3.
* Функціональність обробки платежів повністю залежить від доступності та правил стороннього сервісу Stripe API.
* Серверна частина системи розгортається на операційній системі Windows з використанням середовища виконання Node.js та СУБД MySQL.
* Усі матеріали проєкту, включно з документацією та інтерфейсом користувача, виконуються українською мовою.

## 2.5 Припущення й залежності

Припущення: Передбачається, що користувачі-глядачі знайомі з загальним процесом онлайн-покупок. Адміністративний персонал має базові навички роботи з комп'ютером, достатні для освоєння інтерфейсу панелі управління.

Залежності: Надійність та доступність функції онлайн-оплати залежить від стабільності роботи платіжного шлюзу Stripe. Загальна продуктивність системи залежить від конфігурації та навантаження на серверне обладнання.

# 3 КОНКРЕТНІ ВИМОГИ

## 3.1 Вимоги до зовнішніх інтерфейсів

Інтерфейс користувача є веб-орієнтованим та реалізований у вигляді SPA. Дизайн орієнтований на мінімалізм, інтуїтивність та зручність використання. Поточна реалізація оптимізована для настільних комп'ютерів (desktop-first). Ключові екрани включають:

* Головна сторінка: Відображення сітки постерів актуальних фільмів.
* Сторінка фільму: Детальна інформація, розклад сеансів за датами.
* Сторінка вибору місць: Інтерактивна схема кінозалу зі статусами місць (вільно, зайнято, обрано).
* Особистий кабінет: Дані профілю, історія квитків з QR-кодами.
* Сторінки реєстрації та авторизації.
* Адміністративна панель: Табличне представлення даних, модальні вікна для редагування та створення сутностей, інформаційна панель з графіками.

Прямі вимоги до апаратних інтерфейсів відсутні, оскільки система є вебзастосунком.

* REST API: Серверна частина надає клієнту стандартизований RESTful API для обміну даними. Взаємодія відбувається через HTTP-запити (GET, POST, PUT, DELETE), а дані передаються у форматі JSON.
* Stripe API: Система інтегрується із зовнішнім платіжним сервісом Stripe. Для проведення оплати система створює платіжну сесію через Stripe API та перенаправляє користувача на захищену сторінку оплати Stripe Checkout.

Взаємодія між клієнтським браузером та вебсервером/сервером застосунків відбувається за протоколом HTTP/HTTPS. Взаємодія між сервером застосунків та сервером бази даних здійснюється через внутрішнє TCP/IP з'єднання.

## 3.2. Функціональні вимоги

**FR-1: Перегляд афіші та інформації про фільми.** Система повинна відображати список актуальних фільмів у прокаті. Користувач повинен мати можливість перейти на сторінку конкретного фільму для перегляду детальної інформації (опис, жанр, тривалість, акторський склад, трейлер).

**FR-2: Перегляд розкладу сеансів.** На сторінці фільму має відображатися розклад доступних сеансів із зазначенням дати, часу та формату показу.

**FR-3: Інтерактивний вибір місць.** Після вибору сеансу система повинна відображати інтерактивну схему кінозалу зі статусами місць (вільно, зайнято, обрано).

**FR-4: Бронювання квитків.** Користувач повинен мати можливість обрати одне або декілька вільних місць та перейти до оформлення замовлення.

**FR-5: Реєстрація та авторизація.** Система має надавати можливість створення особистого кабінету (реєстрація за email та паролем) та входу в систему (авторизація) для збереження історії замовлень.

**FR-6: Автентифікація.** Доступ до адміністративної панелі повинен бути захищений системою логіну та паролю.

**FR-7: Управління фільмами (CRUD).** Адміністратор повинен мати можливість додавати нові фільми, редагувати інформацію про існуючі та видаляти неактуальні позиції.

**FR-8: Управління кінозалами (CRUD).** Система повинна дозволяти створювати нові кінозали, вказуючи їхню конфігурацію (кількість рядів та місць), та редагувати існуючі.

**FR-9: Управління сеансами (CRUD).** Адміністратор повинен мати можливість створювати нові сеанси, прив'язуючи фільм до конкретного залу, дати та часу.

**FR-10: Моніторинг бронювань.** Адміністратор повинен мати доступ до переліку активних бронювань на кожен сеанс для контролю заповненості залів.

**FR-11: Інформаційна панель (Dashboard).** В адміністративній частині має бути реалізовано панель для відстеження ключових показників (кількість користувачів, проданих квитків, загальний дохід) за допомогою візуалізованих графіків.

## 3.3. Атрибути програмного продукту (Нефункціональні вимоги)

**NFR-1:** Система повинна коректно обробляти помилкові ситуації (наприклад, спроба подвійного бронювання одного місця, помилки при оплаті) та надавати користувачеві відповідні інформативні повідомлення. Цілісність даних у базі даних повинна бути забезпечена на рівні СУБД.

**NFR-3:** Система повинна бути доступною для користувачів 24/7, за винятком планових технічних робіт.

**NFR-4:** Паролі адміністраторів та користувачів у базі даних повинні зберігатися у хешованому вигляді (з використанням алгоритму bcrypt). Доступ до адміністративної панелі та особистого кабінету користувача має бути обмежений на основі ролей та захищений за допомогою JWT-токенів.

**NFR-5:** Архітектура системи (розділення на клієнтську та серверну частини) повинна дозволяти подальше розширення функціонала без необхідності повної перебудови продукту. Код має бути структурованим та документованим для спрощення подальшої підтримки.

**NFR-6:** Клієнтська частина є кросбраузерною та має працювати в останніх версіях популярних браузерів (Google Chrome, Mozilla Firefox). Серверна частина, розроблена на Node.js, може бути розгорнута на будь-якій операційній системі, що підтримує дане середовище виконання.

**NFR-7:** Час завантаження основних сторінок (головна, сторінка фільму) не повинен перевищувати 2 секунди. Час відгуку АРІ на ключові запити (отримання списку сеансів, стану місць у залі) має бути в межах 500 мс.

## 3.4 Вимоги до програмного та апаратного забезпечення

Для середовища розробки:

* Операційна система: Windows 10/11
* Середовище виконання: Node.js (версія 18 або вище).
* Система керування пакунками: npm.
* Редактор коду: Visual Studio Code.
* Система керування базами даних: MySQL Server (версія 8.0 або вище).
* Веббраузер: Google Chrome, Mozilla Firefox (останні версії).

Для серверної частини (середовище розгортання):

* Операційна система: Windows.
* Середовище виконання Node.js, вебсервер, СУБД MySQL.

Для клієнтської частини:

* Будь-який сучасний веббраузер з підтримкою JavaScript (ES6+) та CSS3.

## 3.5 Вимоги до бази даних

Система повинна використовувати реляційну СУБД MySQL версії 8.0 або вище. Схема бази даних має бути спроєктована відповідно до третьої нормальної форми (3НФ) для забезпечення цілісності даних та уникнення їх надлишковості. База даних повинна зберігати інформацію про сутності: користувачі, фільми, жанри, кінозали, сеанси, квитки, платежі, коментарі та оцінки. Зв'язки між таблицями реалізуються за допомогою механізму первинних та зовнішніх ключів.

## 3.6 Інші вимоги

Продукт розроблено з використанням технологій з відкритим кодом. Усі матеріали проєкту виконуються українською мовою.