

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК
СЕКЦІЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ПРОЕКТУВАННЯ

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему: «Web-додаток підтримки діяльності Сумського національного академічного театру драми та музичної комедії імені М. С. Щепкіна»

за спеціальністю 122 «Комп'ютерні науки»,
освітньо-професійна програма «Інформаційні технології проектування»

Виконавець роботи: студентка групи ІТ-71-9 Гаврилів Анна Олегівна

**Кваліфікаційна робота бакалавра
захищена на засіданні ЕК
з оцінкою**

_____ «__» _____ 2021 р.

Науковий керівник

(підпис)

к.т.н., доц. Нагорний В.В.

(науковий ступінь, вчене звання, прізвище та ініціали)

Голова комісії

(підпис)

Шифрін Д. М.

(науковий ступінь, вчене звання, прізвище та ініціали)

Засвідчую, що у цій дипломній роботі немає
запозичень з праць інших авторів
без відповідних посилань.

Студент _____
(підпис)

Суми-2021

Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра комп'ютерних наук
Секція інформаційних технологій проектування
Спеціальність 122 «Комп'ютерні науки»
Освітньо-професійна програма «Інформаційні технології проектування»

ЗАТВЕРДЖУЮ

Зав. секцією ІТП

_____ В. В. Шендрик
«__» _____ 2021 р.

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА СТУДЕНТУ

Гаврилів Анна Олегівна

1 Тема роботи Web-додаток підтримки діяльності Сумського національного академічного театру драми та музичної комедії імені М. С. Щепкіна

керівник роботи Нагорний Володимир В'ячеславович, к.т.н., доц,

затверджені наказом по університету від «14» квітня 2021 р. №0181-VI

2 Строк подання студентом роботи «7» червня 2021 р.

3 Вхідні дані до роботи відомості про роботу театру та репертуарний план

4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) аналіз предметної області, проектування Web-додатку, розробка Web-додатку.

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) сучасний стан проблеми, мета та задачі проекту, аналіз аналогів Web-додатків театрів, функціональні вимоги до Web-додатку, контекстна діаграма у нотації IDEF0, діаграма декомпозиції першого рівня у нотації IDEF0, діаграма декомпозиції другого рівня блоку «Купівля квитка», діаграма варіантів використання, діаграма потоків даних, логічна модель даних, діаграма послідовності та діаграма діяльності, засоби реалізації, демонстрація роботи, апробація, висновки.

6. Консультанти розділів роботи:

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7.Дата видачі завдання 01.10.2020

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Планування робіт	до 02.04.2021	
2	Розробка технічного завдання	до 9.04.2021	
3	Проведення аналізу предметної області	до 20.04.2021	
4	Проведення структурно-функціонального моделювання	до 30.04.2021	
5	Розробка додатку	до 20.05.2021	
6	Тестування додатку	до 25.05.2021	
7	Оформлення та здача пояснювальної записки та файлів розробленого проєкту	до 06.06.2021	

Студент _____
(підпис)

Гаврилів А.О.

Керівник роботи _____
(підпис)

к.т.н., доц. Нагорний В. В.

РЕФЕРАТ

Тема кваліфікаційної роботи бакалавра «Web-додаток підтримки діяльності Сумського національного академічного театру драми та музичної комедії імені М. С. Щепкіна».

Пояснювальна записка складається зі вступу, 3 розділів, висновків, списку використаних джерел із 20 найменувань, додатків. Загальний обсяг роботи – 85 сторінок, у тому числі 52 сторінки основного тексту, 2 сторінки списку використаних джерел, 31 сторінка додатків.

Кваліфікаційну роботу бакалавра присвячено розробці Web-додатку підтримки діяльності Сумського національного академічного театру драми та музичної комедії імені М. С. Щепкіна.

В першому розділі були сформульовані мета і задачі, проведено аналіз аналогів, в процесі якого були визначені переваги та недоліки аналогічних додатків театру, для уникнення недоліків в створеному додатку, та можливих варіантів покращення додатку. Були сформульовані функціональні вимоги, та обрані засоби реалізації.

Другий розділ присвячено виконанню структурно-функціонального моделювання, моделюванню варіантів використання Web-додатку, моделюванню аналізу Web-додатку театру та проектуванню моделі бази даних.

У третьому було описано процес розробки додатку, демонструється його архітектура та використання.

Результатом виконання роботи є розроблений Web-додаток підтримки діяльності Сумського національного академічного театру драми та музичної комедії імені М. С. Щепкіна.

Ключові слова: Web-додаток, ситуативна потреба, вистава, квиток, театр.

ЗМІСТ

ВСТУП.....	6
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	7
1.1 Огляд останніх досліджень і публікацій.....	7
1.2 Аналіз програмних продуктів – аналогів.....	8
1.3 Постановка задачі.....	13
1.4 Вибір засобів реалізації.....	14
2 ПРОЕКТУВАННЯ WEB-ДОДАТКУ	16
2.1 Структурно-функціональне моделювання.....	16
2.2 Моделювання варіантів використання Web-додатку.....	19
2.3 Модель аналізу Web-додатку театру.....	21
2.4 Проектування моделі бази даних	22
3 РОЗРОБКА WEB-ДОДАТКУ ПІДТРИМКИ ДІЯЛЬНОСТІ СУМСЬКОГО НАЦІОНАЛЬНОГО АКАДЕМІЧНОГО ТЕАТРУ ДРАМИ ТА МУЗИЧНОЇ КОМЕДІЇ ІМЕНІ М. С. ЩЕПКІНА.....	30
3.1 Архітектура Web-додатку.....	30
3.2 Реалізація Web-додатку	31
3.3 Використання Web-додатку.....	36
ВИСНОВКИ	51
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	53
ДОДАТОК А	55
ДОДАТОК Б.....	64
ДОДАТОК В ЛІСТИНГ КОДУ ГОЛОВНОЇ СТОРІНКИ ДОДАТКУ	72
ДОДАТОК Г ЛІСТИНГ КОДУ SPECTACLE CONTROLLER.....	78
ДОДАТОК Д ЛІСТИНГ КОДУ KABINET CONTROLLER	82
ДОДАТОК Е ОПРИЛЮДНЕННЯ РЕЗУЛЬТАТІВ РОБОТИ.....	85

ВСТУП

Постійний розвиток в галузях науки, зростаючі обсяги інформації та складність процесів, що відбуваються навколо нас сприяють впровадженню інформаційних технологій майже у всі сфери діяльності людини. Мистецтво не є виключенням. Шанувальники театрального мистецтва часто відвідують сайти театрів в пошуках інформації і можуть з упевненістю зазначити, що сайт створює перше враження про театр, є його візитною карткою. Досить часто у відвідувачів театру виникає проблема з вибором вистави та купівлею квитка. Скоріш за все перше питання вирішиться після перегляду афіші, або рекомендації знайомих. З другим питанням складніше, глядача очікує похід до каси театру де може не бути квитків, або ж вона буде зачинена. Сьогодні ці питання можна вирішити більш зручним способом, не виходячи з дому, завдяки Web-додатку, який буде мати функцію та купівлі квитків, а також підбору вистави за вподобаннями глядача. Наразі театр має додаток, який включає в себе афішу, новини та інформацію про вистави і театр. Існуючий Web-додаток театру має застарілий дизайн, що негативно впливає на його репутацію серед глядачів. Також відсутній необхідний функціонал для купівлі квитків. Адміністрування сайту є незручним. На основі цього було прийнято рішення про створення нового Web-додатку.

Метою даної роботи є розробка Web-додатку підтримки діяльності Сумського національного академічного театру драми та музичної комедії імені М. С. Щепкіна. Для реалізації поставленої мети необхідно виконати наступні задачі:

- провести аналіз аналогічних додатків;
- сформулювати вимоги до додатку;
- провести моделювання додатку;
- реалізувати Web-додаток;
- провести тестування додатку.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд останніх досліджень і публікацій

Розробка Web-додатку театру має на меті якісний вплив як на репутацію театру, так і на зручність для користувачів, а отже збільшення кількості його відвідувачів.

Зараз в світі де ринок пропозицій досить великий, для споживача більш пріоритетним є не факт вирішення задачі найкращим чином, а питання вирішення в конкретній ситуації. Відповідно акцент зміщується від потреби до ситуації, в якій вона буде вирішена.

Зараз користувач спочатку обирає між пропозицією та її відсутністю, потім найкращий варіант з запропонованих, але, який в той же час задовольняє його потребам. Споживач купує не пропозицію, а вирішення своєї проблеми або ситуації. [1]

Таким чином купуючи квиток на виставу для побачення з дівчиною, споживач вирішує ситуацію з дозвіллям на цей вечір. Тож якщо обрана ним вистава буде зовсім не розташованого до побачення характеру, споживач навряд чи залишиться задоволеним в даній ситуації, і вирішить, що даний варіант, тобто похід до театру, не є кращим вирішенням його ситуації з дозвіллям, а отже не вдасться до повторного походу.

Так вибір вистави залежить від того з ким ми вирішили піти до театру та від нашого настрою. Отже використання підбору вистави за ситуативними потребами [2] глядача дозволить вирішити дану проблему.

1.2 Аналіз програмних продуктів – аналогів

Сьогодні задля полегшення планування дозвілля майже всі театри мають свій Web-додаток. Деякі з них пропонують користувачам лише перегляд афіші, інші крім цього мають інтернет-касу, для купівлі квитків.

Для аналізу додатків, було обрано два театри, які знаходяться в м. Суми та м. Києві.

Сумський театр для дітей та юнацтва

Web-додаток Сумського театру для дітей та юнацтва надає користувачу можливість перегляду афіші та інформації про виставу.

Web-додаток містить інформацію про історію та новини театру. Сторінка «квитки» містить лише планування залу з зазначенням вартості квитка на тому чи іншому місці.

Web-додаток театру не має обробника подій в разі відсутності вистав. Так на час карантину користувач додатку бачить напис «Завантаження» або порожнє місце в блоці відведеному під вистави.

Карусель головної сторінки Web-додатку немає ніякого функціонального навантаження, так як містить фото театру.

Передбачений розробниками блок для зображення місцезнаходження театру за допомогою Google карти, містить інформацію про помилку завантаження Google карти.

Серед переваг цього додатку можна виділити наявність календаря подій, що є зручним функціоналом, для перегляду афіші.

На рисунках 1.1 – 1.2 представлено Web-додаток, а саме: головна сторінка та сторінка афіші під час карантину.

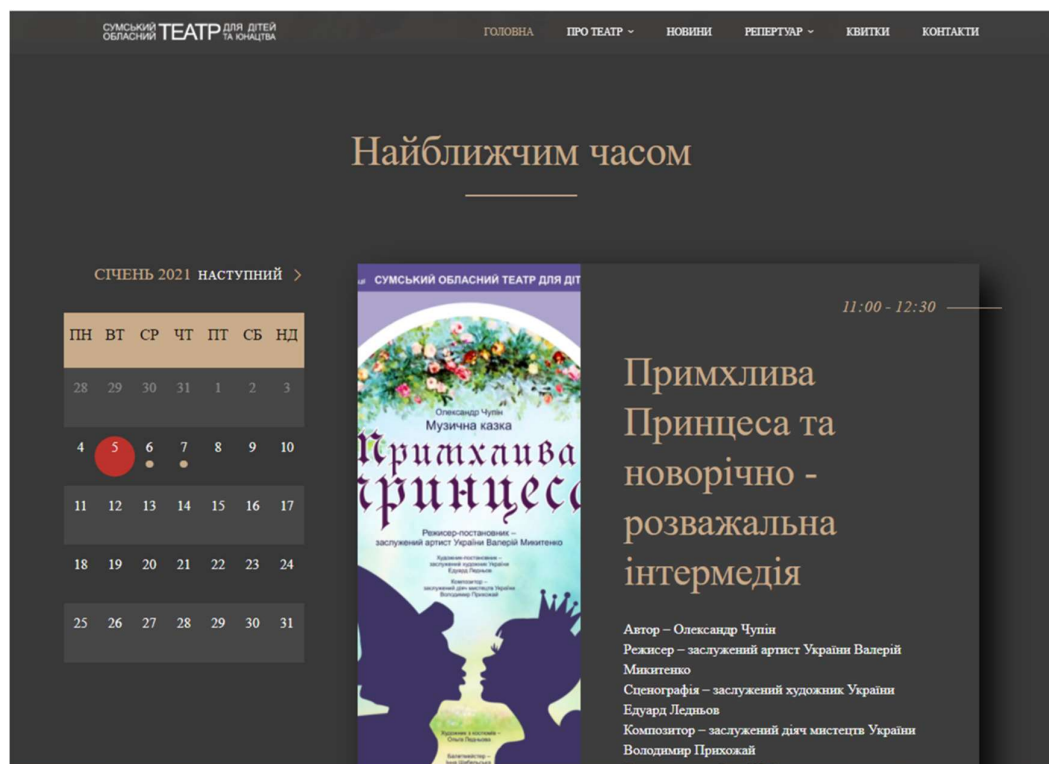


Рисунок 1.1 – Головна сторінка Web-додатку Сумського театру для дітей та юнацтва

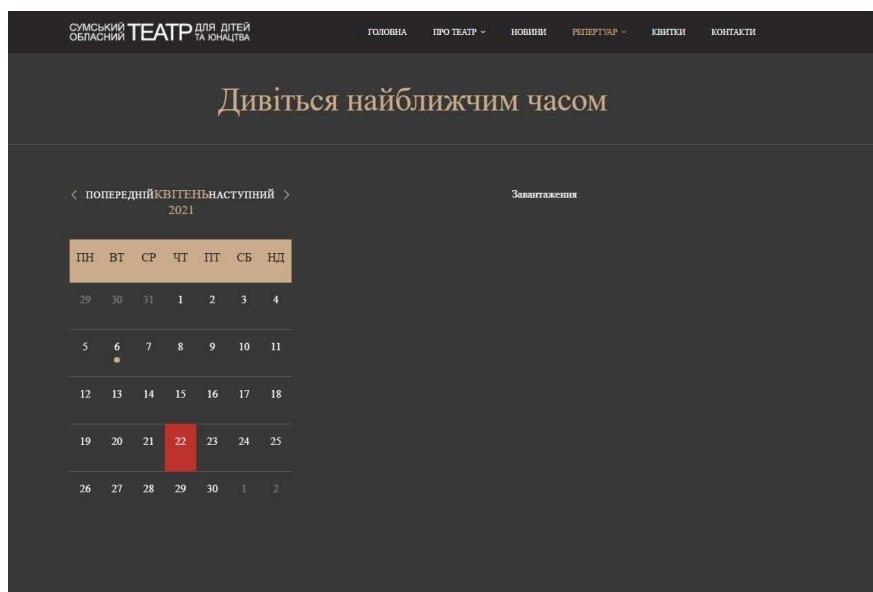


Рисунок 1.2 – Сторінка афіші Web-додатку Сумського театру для дітей та юнацтва під час карантину

Київський національний академічний «Молодий театр»

Web-додаток Київського національного академічного «Молодого театру» надає користувачу можливість перегляду афіші, інформації про виставу та можливість купівлі квитка. Web-додаток містить також інформацію про новини театру та його митців. Перевагою додатку є наявність кабінету користувача.

Додаток має візуально цікаво оформлену сторінку афіші, яка підкреслює його предметну область.

На рисунках 1.3 – 1.5 представлено Web-додаток, а саме: головна сторінка, сторінка афіші та сторінка придбання квитка.

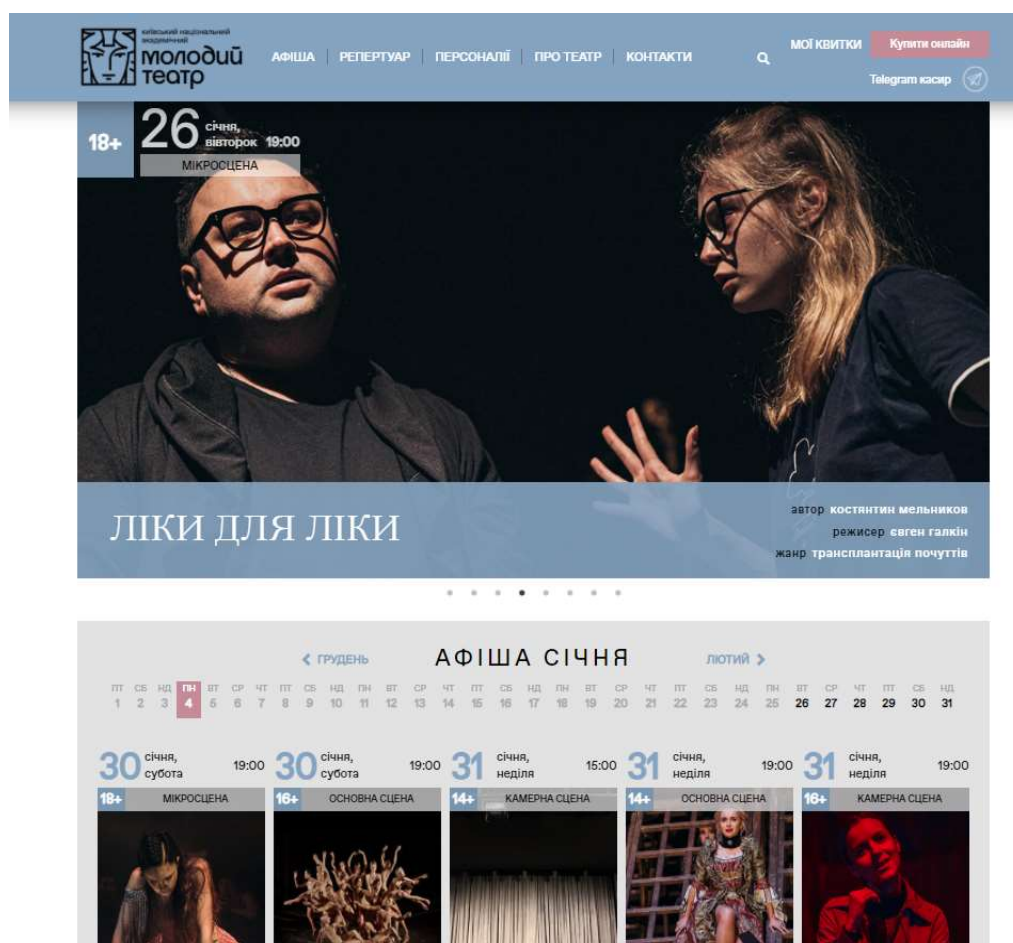


Рисунок 1.3 – Головна сторінка Web-додатку Київського національного академічного «Молодого театру»

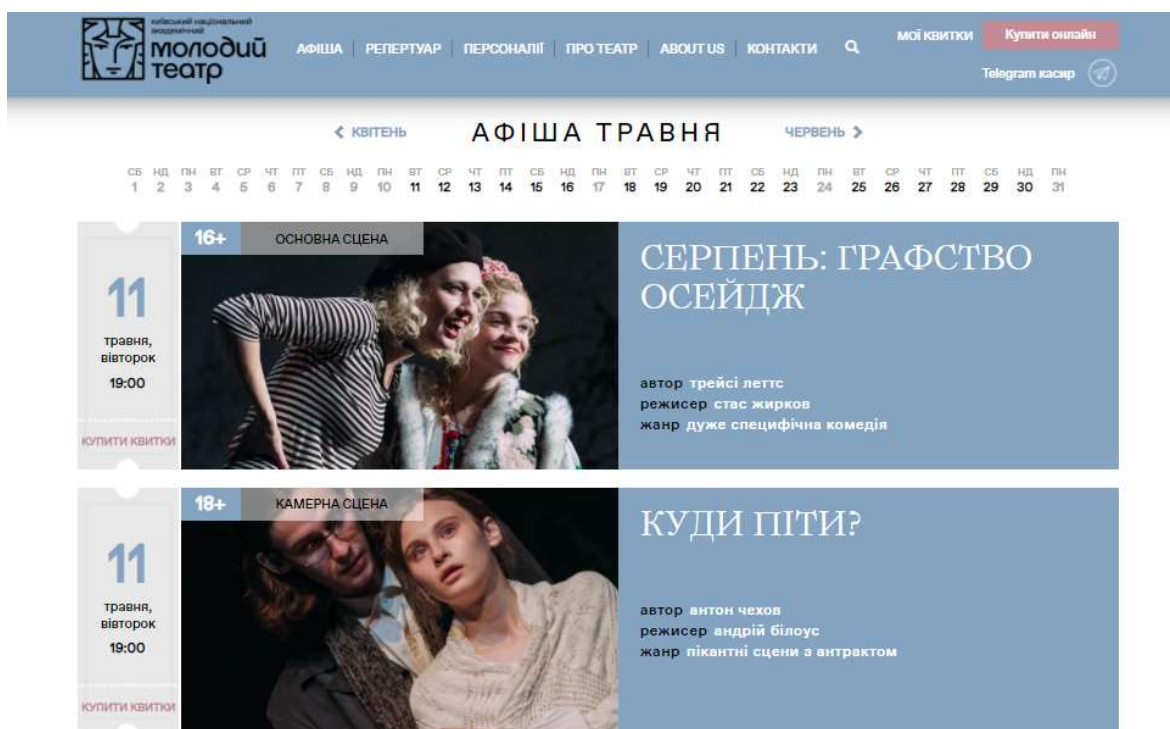


Рисунок 1.4 – Сторінка афіші Web-додатку Київського національного академічного «Молодого театру»

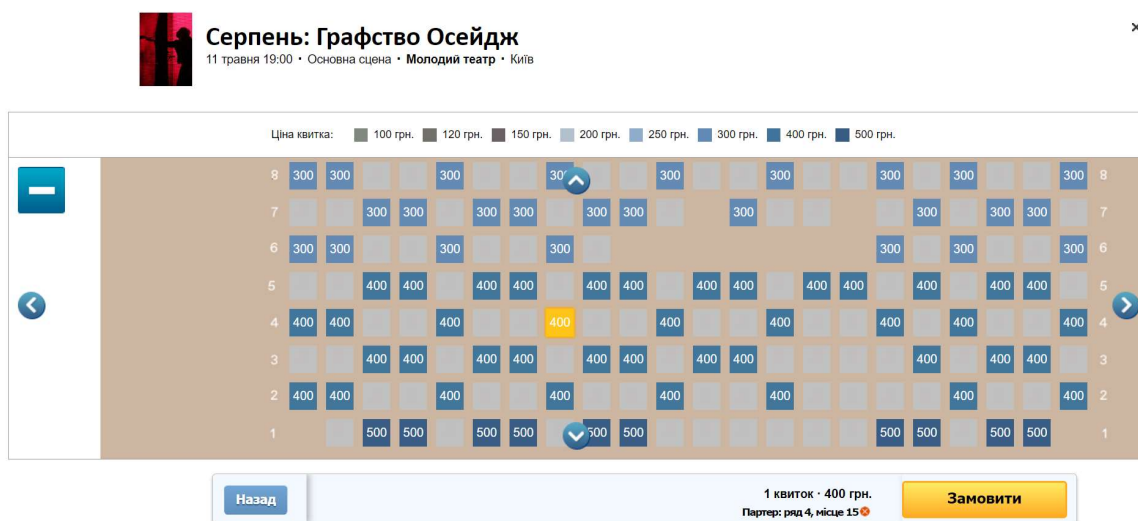


Рисунок 1.5 – Сторінка придбання квитка Web-додатку Київського національного академічного «Молодого театру»

Для визначення актуальності розробки результати проведеного аналізу характеристик створюваного Web-додатку, наявного та аналогів були сформовані у вигляді порівняльної таблиці 1.1.

Таблиця 1.1 – Порівняння аналогів

Функції	ТЮГ	Молодий театр	Наявний Web-додаток театру ім. М. С. Щепкіна»	Власна розробка
Купівля квитку	-	+	-	+
Наявність афіші	+	+	+	+
Кабінет користувача	-	+	-	+
Сучасний інтерфейс	+	+	-	+
Модуль підбору вистави	-	-	-	+
Дійсність посилань	-	+	-	+

На основі аналізу аналогів було прийнято рішення про доцільність створення власного Web-додатку театру ім. М. С. Щепкіна.

1.3 Постановка задачі

Сьогодні майже кожна людина має доступ до мережі Інтернет. Кожного дня людина занурюється в світ електронної інформації задля знаходження відповідей на свої питання починаючи від прогнозу погоди закінчуючи вибором фільму для вечірнього перегляду. Отже створення Web-додатку театру з можливістю підбору вистави та придбання квитків є доцільним.

Для реалізації поставленої мети необхідно провести аналіз аналогічних додатків, для визначення їх переваг та недоліків, що допоможе в формулюванні вимог до додатку, а також допоможе уникнути помилок та недоліків в створеному додатку.

Наступним етапом є проведення моделювання додатку, що забезпечує створення більш якісного продукту, завдяки аналізу бізнес-процесів.

Створюваний Web-додаток має відповідати наступним функціональним вимогам:

- можливість купівлі квитків;
- можливість підбору вистави за ситуативними потребами;
- можливість перегляду історії відвіданих вистав;
- адміністрування Web-додатку;
- перегляд афіші, інформації про виставу та митців театру;
- можливість перегляду інформації про театр та його новини.

Цільовою аудиторією створюваного Web-додатку є працівники закладу, глядачі театру, жителі та гості міста Суми.

1.4 Вибір засобів реалізації

В наш час Web-додатки є найпопулярнішими з додатків. Це обумовлено тим що сучасні мобільні телефони надають користувачеві в мережі Інтернет такий же набір можливостей, що й персональні комп'ютери. Це зумовлено підтримкою HTML5. Крім даного засобу для розробки Web-додатку також використовується CSS, PHP, та JavaScript. [3]

HTML та CSS призначені для верстки додатку, яка відбувається на основі HTML тегів та CSS властивостей.[4]

Для спрощення роботи над додатком було прийнято рішення про використання фреймворку Bootstrap. Даний фреймворк дозволяє уникнути самостійного написання більшості css-стилів та Web-сценаріїв, шляхом підключення фреймворку. Bootstrap також варто використовувати задля адаптивної верстки. Bootstrap вже має в собі медіа-запити, для того, щоб при відкритті додатку на будь-якому пристрої він мав гарний вигляд. Отже завдяки використанню фреймворку Bootstrap питання кросплатформності та адаптивності буде вирішено, що значно спростить задачу розробки Web-додатку.[5]

Звісно окрім переваг даний фреймворк має декілька недоліків. Оскільки використання Bootstrap стає все популярнішим, досить часто можна зустріти сайти з однаковими кнопками та іншими елементами додатку. Це питання можна легко вирішити шляхом кастомізації запропонованих стандартних елементів. Іншою проблемою є надлишковість інформації. Оскільки виконуючи розробку самостійно кожна властивість має своє призначення, в даному ж випадку, файл містить властивість для будь-якого варіанту використання. Для вирішення цієї проблеми можна обирати лише ті компоненти для завантаження, які ви вважаєте будуть корисні для вашої розробки.

Отже, вирішивши питання з візуальною частиною додатку, можна переходити до функціональної. Для цього прийнято використовувати мову PHP. Це

спеціальна мова програмування розроблена для створення Web-додатків.[6] Рушійною силою в даному питанні є PHP-скрипти, які виконуються на стороні серверу. [7]

Кожен Web-додаток має відповідати вимогам безпеки та надійності, і для вирішення цього питання доцільним є використання php-фреймворку Laravel.

Laravel дозволяє спростити вирішення таких питань, як аутентифікація, кешування, маршрутизація та ін. Фреймворк включає в себе хостинг та платформу для розгортання додатків.[8]

При розробці додатку велику увагу необхідно приділяти тестуванню, адже від цього залежить якість вашої системи. При виконанні юніт та інтеграційного тестуванні завдяки Laravel можна обійтися без стороннього забезпечення, оскільки даний фреймворк підтримує виконання тестування компонентів.[9]

2 ПРОЕКТУВАННЯ WEB-ДОДАТКУ

2.1 Структурно-функціональне моделювання

Для систематизації інформації щодо роботи додатку та функцій, які він має містити першочергово необхідно було провести моделювання роботи додатку.

Виконувати опис бізнес-процесів покликана методологія IDEF0. Головним компонентом даної моделі є діаграми. В основі стандарту IDEF0 лежить концепція функціональних блоків та зв'язків між ними. На першому етапі за методологією IDEF0 виконується моделювання контекстної діаграми, що представляє собою єдиний блок опису системи та її зв'язків з оточуючим середовищем.[10]

Контекстна діаграма в нотації IDEF0 Web-додатку підтримки діяльності Сумського національного академічного театру драми та музичної комедії імені М. С. Щепкіна. представлена на рисунку 2.1.

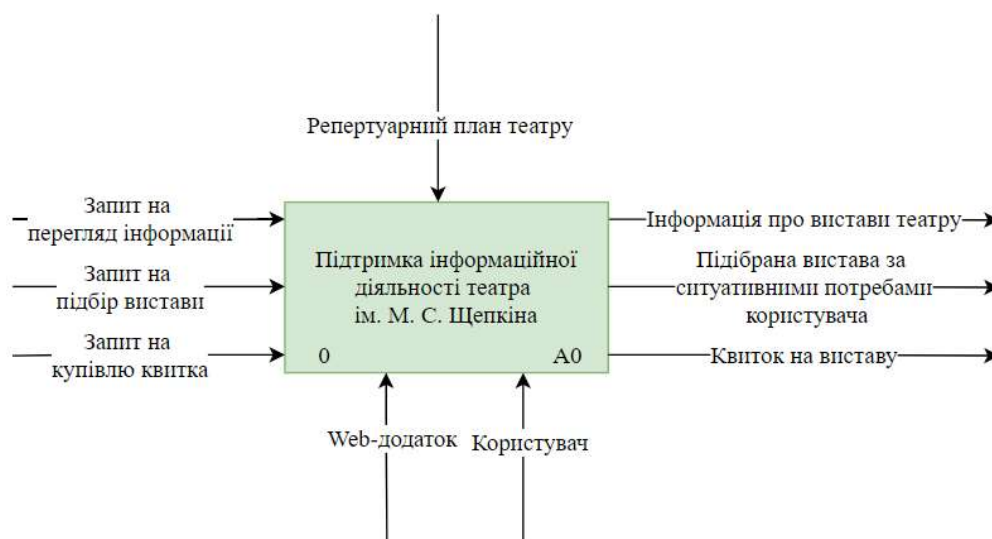


Рисунок 2.1 – Контекстна діаграма в нотації IDEF0

Кожен рівень IDEF0 діаграми має блоки та дуги. Блоки відображають функції які повинна мати розроблювана система, дуги в свою чергу відображають взаємозв'язки між блоками, їх взаємодію. [11]

На вході до системи знаходяться запити користувача на перегляд інформації, підбір вистави та купівлю квитка. Керування виконується репертуарним планом театру. Підтримка роботи додатку театру виконується завдяки наявності Web-додатку та користувача. Результатами роботи є інформація про вистави театру, обрана вистава та квиток.

Нотація IDEF0 підтримує декомпозицію діаграм до потрібного рівня деталізації. Найнижчий рівень діаграми утворений при декомпозиції ту ж саму ділянку, що й батьківський процес, тільки опис виконується більш детальноше.[10]

Для більш кращого розуміння бізнес-процесів, діаграму в нотації IDEF0 було декомпововано на підсистеми. Декомпозиція процесу підтримки діяльності театру імені М. С. Щепкіна. зображена на рисунку 2.2.

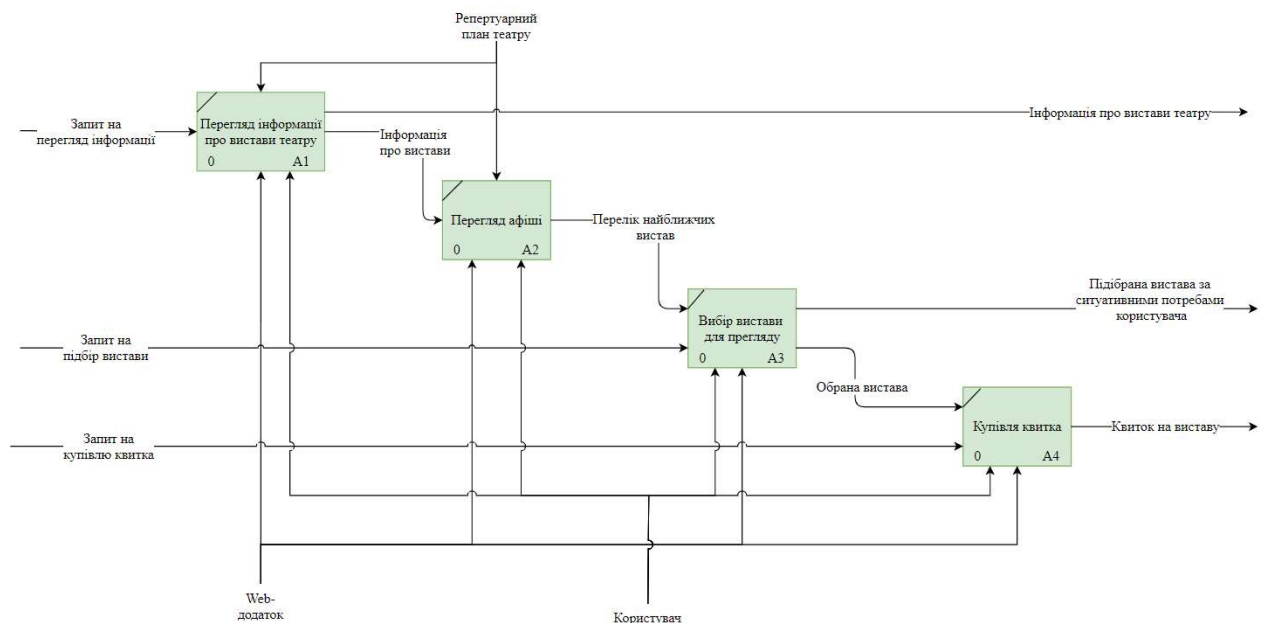


Рисунок 2.2 – Декомпозиція IDEF0-діаграми Web-додатку

Діаграма IDEF0 декомпована на наступні блоки:

- перегляд інформації про вистави театру;
- перегляд афіші;
- вибір вистави для перегляду;
- купівля квитка.

Для виконання кожного функціонального блоку необхідна наявність Web-додатку та користувача. Для функціонування блоку «перегляд інформації про вистави театру» необхідно виконання запиту на перегляд інформації. На виході з цього блоку ми отримуємо інформацію про вистави театру.

Для виконання блоку «вибір вистави для перегляду», необхідна наявність запиту на підбір вистави. Результатом роботи цього блоку є підібрана вистава за ситуативними потребами глядача.

Для виконання роботи блоку «купівля квитка» необхідна наявність запиту на купівлю квитка. На виході з цього блоку отримуємо квиток на виставу.

Роботу блоку «Купівля квитка» декомповано ще на один рівень (рис. 2.3).

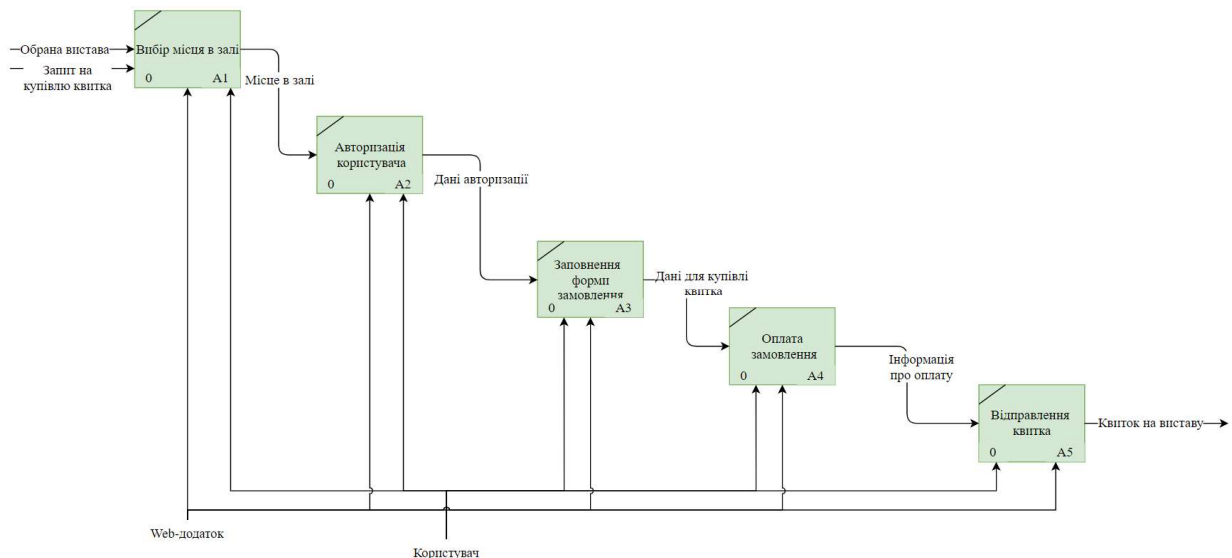


Рисунок 2.3 – Другий рівень декомпозиції IDEF0-діаграми Web-додатку

2.2 Моделювання варіантів використання Web-додатку

Одним із результатів проведення моделювання додатку є діаграма варіантів використання (Use Case), що демонструє взаємодію між акторами системи та можливими сценаріями поведінки.

Актори представляють собою групу осіб або систему, що взаємодіє з сценаріями роботи додатку, але не є його частиною.

Сценарії в свою чергу є так званими сервісами, які додаток надає користувачам. [12]

Варто зазначити, що дана діаграма містить декілька варіантів відносин між сценаріями поведінки розширення, або включення.

Діаграми ВВ діяльності додатку для користувача та банківської системи представлена рисунку 2.4.

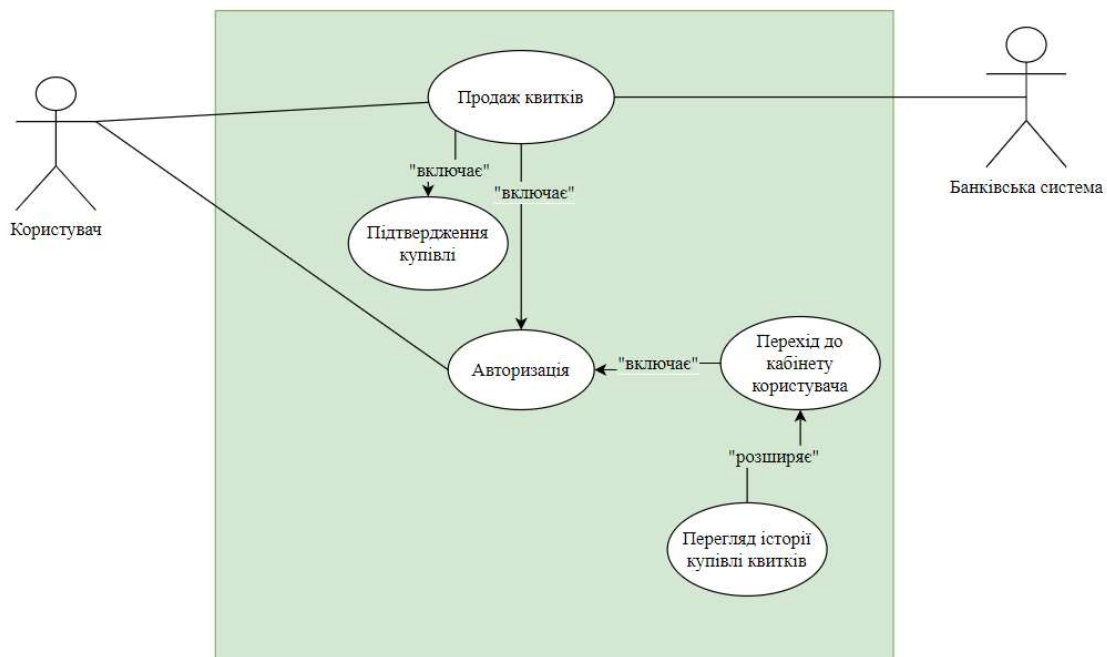


Рисунок 2.4 – Діаграми ВВ діяльності додатку для користувача та банківської системи

Як актори в даній діаграмі ВВ виділені користувач та банківська система.

Варіант використання авторизація дозволяє користувачу авторизуватися в системі, та перейти до свого кабінету.

Варіант використання продаж квитків вимагає від користувача авторизації, а також включає процес підтвердження купівлі квитка. Варіант використання продаж квитків дозволяє користувачеві придбати квитки на обрану ним виставу.

Для моделювання роботи адміністратора з додатком була створення окрема діаграма ВВ зображена на рисунку 2.5.

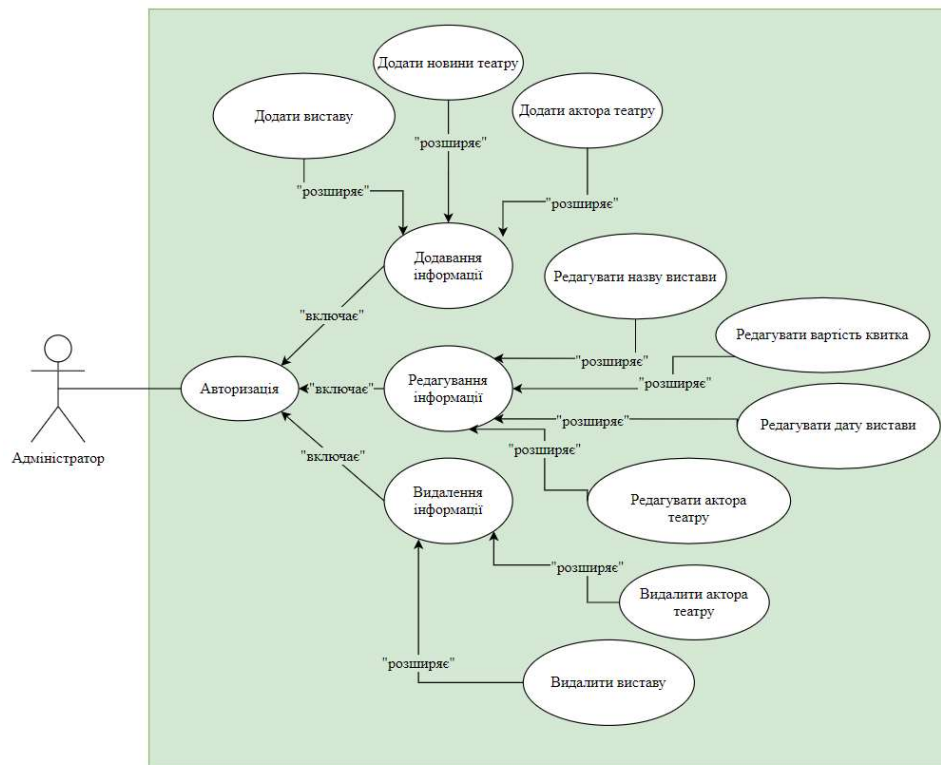


Рисунок 2.5 – Діаграми ВВ діяльності для адміністратора

Як актора в даній діаграмі ВВ виділено адміністратора додатку.

Варіанти використання додавання інформації, редагування інформації та видалення інформації вимагають від автора авторизації в системі.

2.3 Модель аналізу Web-додатку театру

При розробці бізнес-додатків найважливішими з UML діаграм є діаграми послідовності, що відображають послідовність повідомлень та взаємодій, між акторами та об'єктами системи. На рисунку 2.6 представлена діаграма послідовності Web-додатку.



Рисунок 2.6 – Діаграма послідовності

При моделюванні процесу виконання операцій використовують діаграму діяльності. Вона виконує візуалізацію поведінки системи та її окремих елементів. На рисунку 2.7 представлена діаграма діяльності Web-додатку.

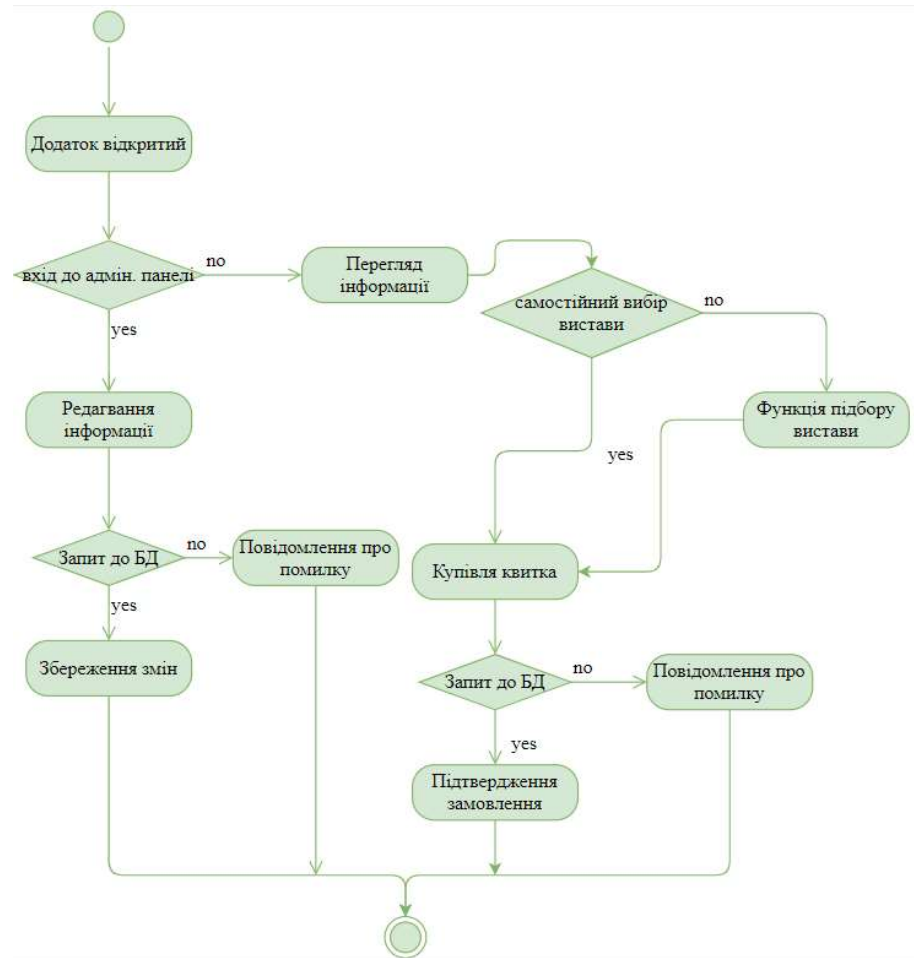


Рисунок 2.7 – Діаграма діяльності

2.4 Проектування моделі бази даних

Логічне моделювання бази даних це процес виконання схеми бази даних враховуючи обрану модель для представлення даних. Логічна модель для розробника додатку, є інформаційною базою для фізичного проектування.[13]

Модель бази даних також має велике значення в процесі експлуатації. Правильно спроектована модель дає уявлення користувачу про вплив внесених ним змін в базу даних, на вже створені в ній записи.[14]

База даних і взаємозв'язки між сутностями є основним елементом Web-додатку.

На рисунку 2.7 представлена логічна модель даних Web-додатку театру.

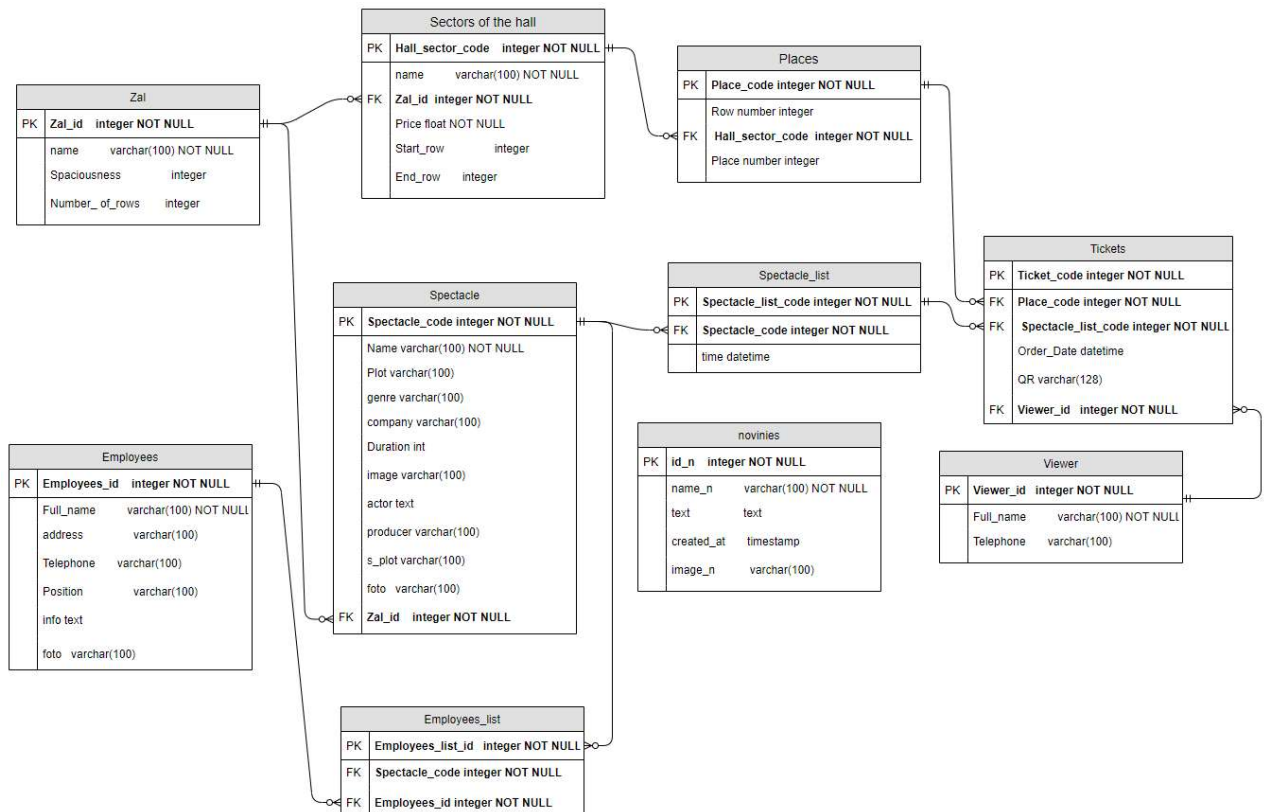


Рисунок 2.7 – Логічна модель даних

В процесі логічного моделювання було виділено такі сутності:

- зал (Zal);
- сектор залу (Sectors_of_the_hall);
- місце (Places);
- глядач (Viewer);
- вистава (Spectacle);

- робітник (Employees);
- новини (Novinies);
- квиток (Tickets).

Зв'язок сутності «Зал» з сутністю «Сектор залу» означає, що зал може мати 0 або більше секторів, але сектор повинен мати один зал.

Зв'язок сутності «Зал» з сутністю «Вистава» означає, що зал може мати 0 або більше вистав, але вистава повинна мати один зал.

Зв'язок сутності «Місце» з сутністю «Сектор залу» означає, що сектор може мати 0 або більше місць, але місце повинно мати сектор.

Зв'язок сутності «Місце» з сутністю «Білет» означає, що місце може мати 0 або більше білетів, але білет повинно мати одне місце.

Зв'язок сутності «Глядач» з сутністю «Білет» означає, що глядач може мати 0 або більше білетів, але білет повинен мати одного глядача.

Зв'язок сутності «Робітник» з сутністю «Вистава» виконується через допоміжну таблицю Employees_list.

Зв'язок сутності «Робітник» з сутністю «Employees_list» означає, що робітник може мати 0 або більше Employees_list, але Employees_list повинно мати одного робітника.

Зв'язок сутності «Вистава» з сутністю «Employees_list» означає, що вистава може мати 0 або більше Employees_list, але Employees_list повинно мати одну виставу.

Зв'язок сутності «Вистава» з сутністю «Білет» виконується через допоміжну таблицю Spectacle_list.

Зв'язок сутності «Вистава» з сутністю «Spectacle_list» означає, що вистава може мати 0 або більше Spectacle_list, але Spectacle_list повинен мати одну виставу.

Зв'язок сутності «Квиток» з сутністю «Spectacle_list» означає, що Spectacle_list може мати 0 або більше квитків, але квиток повинен мати один Spectacle_list.

Виконавши аналіз сутностей, які будуть використані в додатку, можна переходити до створення структури бази даних. Структура бази даних має містити в собі назви таблиць, атрибутів, типів, ключі та обмеження. [13] Структура БД представлена в таблиці 2.1.

Таблиця 2.1 – Структура БД

Таблиця	Поле	Зміст	Тип	Ключі	Обмеження
Zal	Zal_id	Ідентифікатор залу	integer	PK	NOT NULL
	Name	Назва залу	varchar(100)		NOT NULL
	Spaciousness	Кількість місць	integer		
	Number_ of_rows	Кількість рядів	integer		
Sectors of the hall	Hall_sector_code	Ідентифікатор сектору	integer	PK	NOT NULL
	Name	Назва сектору	varchar(100)	FK	NOT NULL
	Zal_id	Ідентифікатор залу	integer		NOT NULL
	Price	Ціна квитка	float		NOT NULL
	Start_row	Початковий ряд сектору	integer		

Продовження таблиці 2.1

Таблиця	Поле	Зміст	Тип	Ключі	Обмеження
Sectors of the hall	End_row	Кінцевий ряд сектору	integer		
Places	Place_code	Ідентифікатор місця	integer	PK	NOT NULL
	Row_number	Номер ряду	integer		
	Hall_sector_code	Ідентифікатор сектору	integer	FK	NOT NULL
	Place	Номер місця	integer		
Viewer	Viewer_id	Ідентифікатор глядача	integer	PK	NOT NULL
	Full_name	Ім'я глядача	varchar(100)		NOT NULL
	Telephone	Номер телефону	varchar(100)		
Spectacle	Spectacle_code	Ідентифікатор вистави	integer	PK	NOT NULL
	Name	Назва вистави	varchar(100)		NOT NULL
	Plot	Логлайн	varchar(100)		
	Actor	Актори	text		
	Producer	Режисер	varchar(100)		

Продовження таблиці 2.1

Таблиця	Поле	Зміст	Тип	Ключі	Обмеження
Spectacle	Genre	Жанр	varchar(100)		
	Company	Компанія для вистави	varchar(100)		
	Duration	Тривалість	integer		
	Image	Зображення	varchar(100)		
	S_plot	Короткий опис	text		
	Foto	Фото	varchar(100)		
	Zal_id	Номер залу	integer	FK	NOT NULL
Employees	Employees_id	Ідентифікатор робітника	integer	PK	NOT NULL
	Full_name	Ім'я робітника	varchar(100)		NOT NULL
	Address	Адреса	varchar(100)		
	Position	Посада	varchar(100)		
	Telephone	Телефон	varchar(100)		
	Info	Інформація	text		
	Foto	Фото	varchar(100)		
Employees_list	Employees_list_id	Ідентифікатор таблиці	integer	PK	NOT NULL

Продовження таблиці 2.1

Таблиця	Поле	Зміст	Тип	Ключі	Обмеження
Employees_list	Spectacle_code	Ідентифікатор вистави	integer	FK	NOT NULL
	Employees_id	Ідентифікатор робітника	integer	FK	NOT NULL
Tickets	Ticket_code	Ідентифікатор квитка	integer	PK	NOT NULL
	Spectacle_list_code	Ідентифікатор вистави	integer	FK	NOT NULL
	Place_code	Ідентифікатор місця	integer		NOT NULL
	Order_Date	Дата купівлі	datetime		
	QR	QR код	varchar(128)		
	Viewer_id	Ідентифікатор глядача	integer	FK	NOT NULL
Spectacle_list	Spectacle_list_code	Ідентифікатор таблиці	integer	PK	NOT NULL
	Spectacle_code	Ідентифікатор вистави	integer	FK	NOT NULL
	Time	Дата вистави	datetime		

Продовження таблиці 2.1

Таблиця	Поле	Зміст	Тип	Ключі	Обмеження
Novinies	ID_n	Ідентифікатор новини	integer	PK	NOT NULL
	Name	Назва новини	varchar(100)		NOT NULL
	Text	Текст новини	text		
	Created_at	Дата створення	timestamp		
	Image	Зображення	varchar(100)		

3 РОЗРОБКА WEB-ДОДАТКУ ПІДТРИМКИ ДІЯЛЬНОСТІ СУМСЬКОГО НАЦІОНАЛЬНОГО АКАДЕМІЧНОГО ТЕАТРУ ДРАМИ ТА МУЗИЧНОЇ КОМЕДІЇ ІМЕНІ М. С. ЩЕПКИНА

3.1 Архітектура Web-додатку

Для представлення інформації про організацію сторінок додатку, навігацію між ними, та отримання доступу до них використовують побудову архітектури додатку [16].

Будь-який Web-додаток містить в собі як клієнтську так і серверну частину. Для реалізації серверної частини додатку було обрано використання фреймворку laravel [8]. Даний фреймворк базується на шаблоні проектування MVC [17] (model-view-controller). Даний шаблон розподіляє внутрішнє подання інформації від способів подання інформації та її прийняття користувачем. Це реалізовано завдяки використанню трьох пов'язаних між собою елементів.

Представлення є безпосередньо інтерфейсом додатку, з яким взаємодіє користувач.

Модель є динамічною структурою даних додатку, що виконує керування даними, логікою та правилами додатку.

Контролер в свою чергу виконує взаємодію між системою та користувачем. Він отримує запит користувача, оброблює його та відповідно надсилає результат до представлення [18]. Взаємодія між серверною та клієнтською частинами додатку зображена на рисунку 3.1.

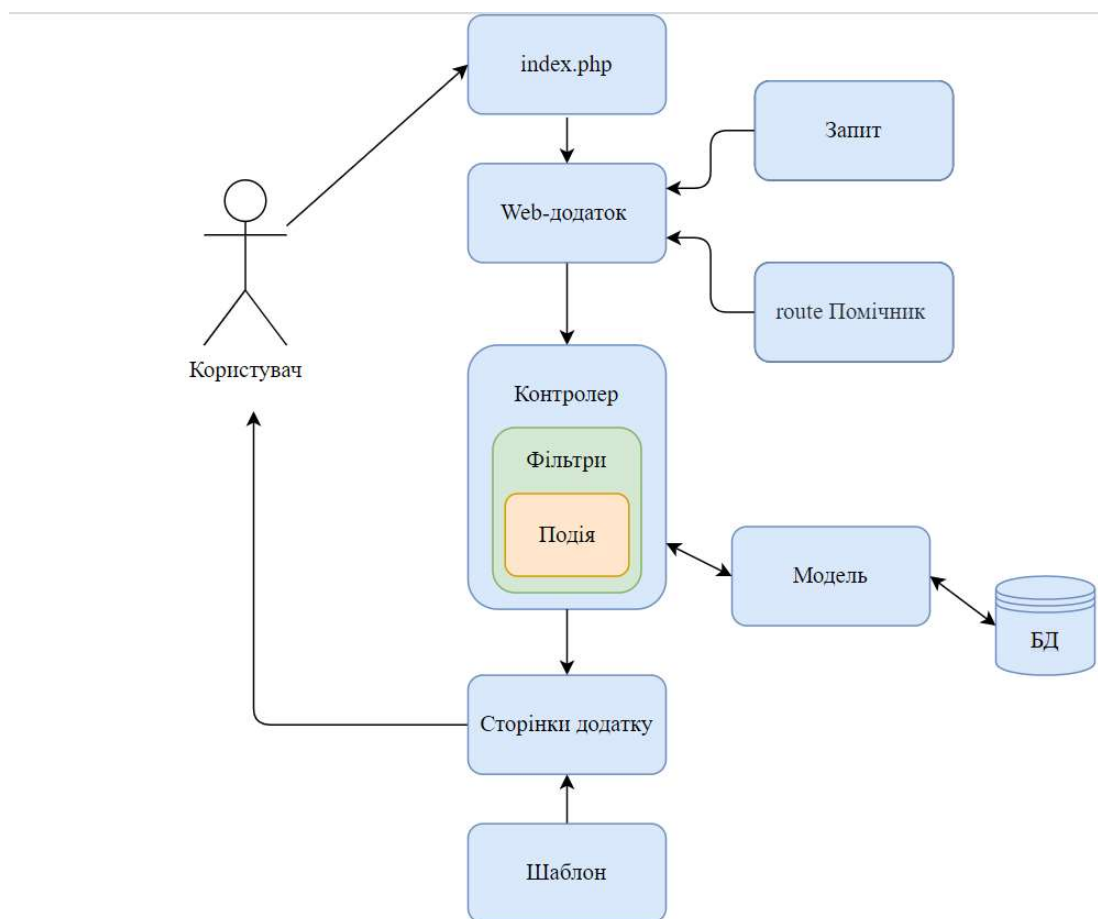


Рисунок 3.1 – Архітектура Web-додатку

3.2 Реалізація Web-додатку

Розробка Web-додатку підтримки діяльності театру складається з декількох етапів. Створення Web-сторінок додатку, їх оформлення, та створення функціональних модулів системи.

Першим кроком було створено макет сторінок, з використанням верстки за допомогою блоків. Результат роботи зображено на рисунку 3.2.

Шаблон головної сторінки додатку розміщено у додатку В.

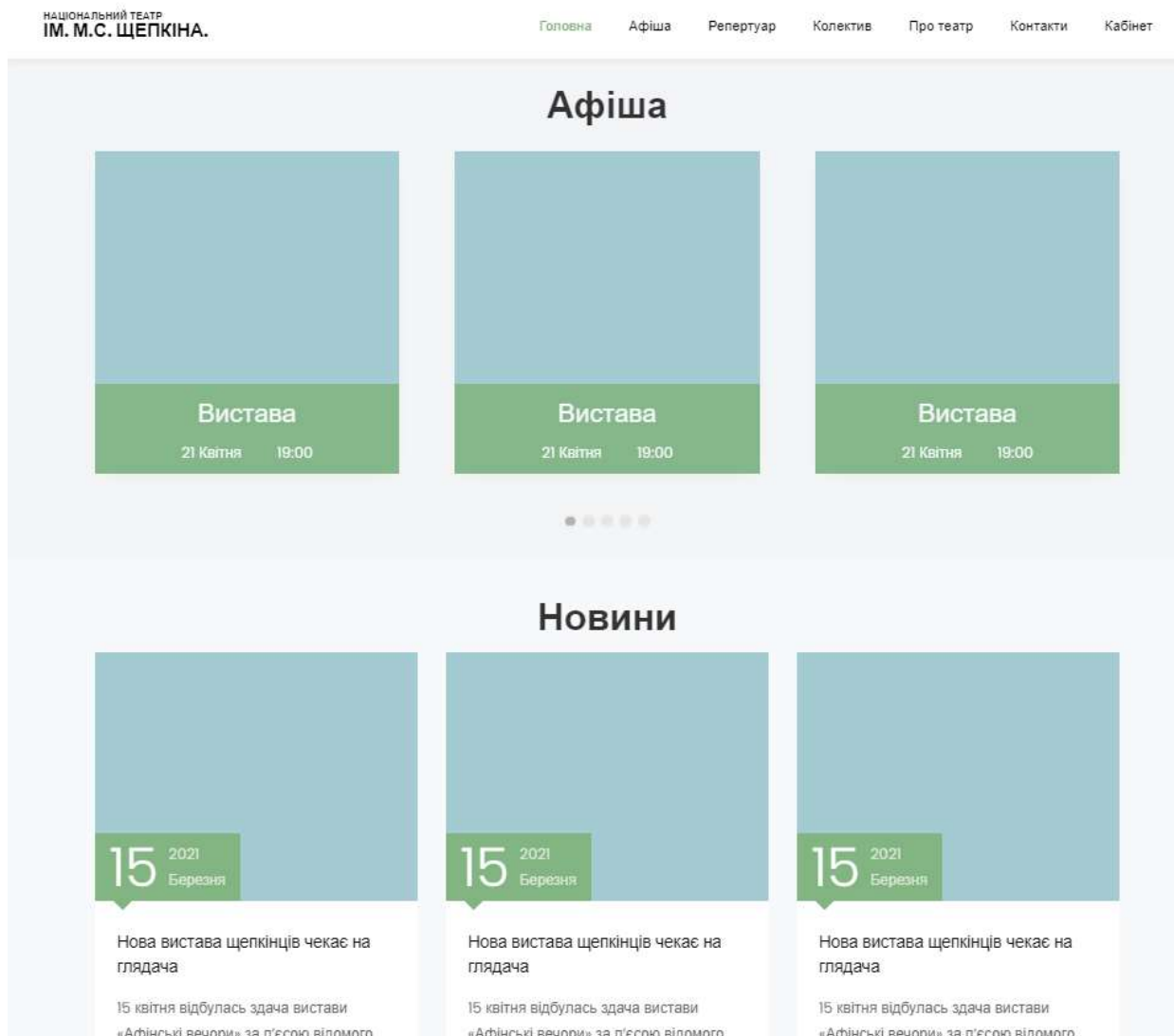


Рисунок 3.2 – Макет головної сторінки Web-додатку

Наступним кроком було створено БД. В laravel створення таблиці в БД передбачає створення файлу міграції, наповнення цього необхідними параметрами, та відповідно виконання міграції, за допомогою термінального вікна.

Результат створення таблиць в БД, та їх наповнення зображено на рисунках 3.3 та 3.4.

id	name	plot	genre	company	duration	image	zal_id	created_at	updated_at	producer	s_plot	foto
1	Лісова пісня	Втрата гармонії між людиною і природою обертається...	віршована п'єса	п'яти з сім'єю	120	/files/case-1.jpg	1	NULL	NULL	Роман Козак	Сценограф – Алія Байтенова. Балетмейстер – Ольга...	/files/case-1_foto.jpg
2	Гедда Габлер	Це тонка психологічна драма, жорсткий, мікроскопич...	Драма на 2 дії	Замислитися	120	/files/case-2.jpg	2	NULL	NULL	Лінас Зайкаускас	Асистент режисера – Владислав Калініченко Музичне...	/files/case-2_foto.jpg
3	ART	<r>Вистава вдало поєднує класичні режисерські прий...	Інтелектуальна комедія	П'яти з сім'єю	60	/files/case-3.jpg	1	NULL	2021-05-25 15:34:52	Владислав Писарев	<r>Режисер-постановник – Владислав Писарев</r>	/files/case-3_foto.jpg
6	Чудернацькі пригоди Бурсака	Вистава «Чудернацькі пригоди бурсака» створена за...	Містична варіація	Переплунути театральну постановку улюбленого твору	120	/files/case-4.jpg	1	NULL	NULL	Ю Смирнов, В.Калініченко	Художник-постановник - Л.Аполлонова. Художники 3...	/files/case-4_foto.jpg

Рисунок 3.3 – Таблиця «Вистави»

id_n	name_n	text	created_at	image_n	updated_at
1	Нова вистава щепкінців чекає на глядача...	15 квітня відбулась здача вистави «Афінські вечори...	2021-05-18 12:21:05	/files/image_1.jpg	NULL
2	З Днем народження, театре!	3 квітня Театр відзначає свій 88-й день народження	2021-05-18 15:57:09	/files/image_2.jpg	NULL
3	Щепкінці урочисто вшанували пам'ять великої україн...	2 березня на Малій сцені Національного академічног...	2021-05-12 16:09:26	/files/image_3.jpg	NULL

Рисунок 3.4 – Таблиця «Новини»

Після даного кроку було створено SpectacleController для відображення контенту на сторінках Web-додатку (рис. 3.5). Відображення інформації на сторінках додатку відбувається завдяки скрипту, наведеному у додатку Г.

```

class SpectacleController extends Controller
{
public function afisha()
{
    $spectacle= new Spectacle;
    $spectacle_list= new Spectacle_list;
    $now = Carbon::now();

    $test = $spectacle::select('*')
->join('spectacle_lists', 'spectacle_lists.id_vistavi', '=', 'spectacles.id')
->whereMonth('spectacle_lists.time', '=', $now->month)->where('spectacle_lists.time', '>', $now)
->orderBy('spectacle_lists.time', 'asc')->get();

    foreach ($test as $el){
        $dt = Carbon::parse($el->time)->locale('uk')->isoFormat('D MMMM');

        $dt2 = Carbon::parse($el->time)->locale('uk')->isoFormat('HH:mm');
        $el->time = $dt;
        $el->test_time = $dt2;
    }

    return view('afisha', ['data'=>$test]);
}
}

```

Рисунок 3.5 – Функція для відображення інформації на сторінці «Афіша»

Для виконання розмежування прав доступу до сторінок Web-додатку було створено дві ролі з різними правами доступу до системи: користувач та адміністратор. Всі необхідні налаштування для створення ролей та обмежень до них містить в собі Laravel-permission пакет [19].

Для того щоб користувачі лише певної ролі мали доступ до сторінки, це необхідно вказати в шляхах які знаходяться в файлі web.php. Зарєєстрований користувач повинен мати доступ лише до кабінету. Адміністратор в свою чергу до адмін-панелі (рис. 3.6 – 3.9).

```
Route::get('/home', [App\Http\Controllers\HomeController::class, 'index'])->name('home');
Route::middleware(['role:admin'])->prefix('admin_panel')->group(function () {
Route::get('/', [App\Http\Controllers\Admin\HomeController::class, 'index'])->name('homeAdmin');
Route::resource('spectacle', 'App\Http\Controllers\Admin\SpectacleController1');
Route::resource('noviny', 'App\Http\Controllers\Admin\NovinyController1');
Route::resource('afisha', 'App\Http\Controllers\Admin\AfishaController1');
});
Route::middleware(['role:user'])->prefix('kabinet')->group(function () {
Route::get('/{id}', 'App\Http\Controllers\KabinetController@index')->name('kabinet');
Route::get('/ticket/{id}', 'App\Http\Controllers\KabinetController@ticket')->name('ticket');
Route::get('/bilet/{id}', 'App\Http\Controllers\KabinetController@bilet')->name('bilet');
});
```

Рисунок 3.6 – Шляхи до сторінок кабінету користувача та адмін-панелі додатку

id	name	email	en	password
2	ann	anna1453s77@gmail.com	NL	\$2y\$10\$Ptv6RDLcL4K4ibFakh0pcO1VXcmigBKHQHdLDbPbvU0...
3	admin	anna145377@gmail.com	NL	\$2y\$10\$ZcZtgKabEISu3Cqe5Koi4O/nkQ30fVqUiDPg7j0BjRm...
4	Саша	o.o.zakharchenko@gmail.com	NL	\$2y\$10\$.PkRJeWMXyNsqiAo78ohBuAK8c0.dGTUZkIVHznVRJ3...

Рисунок 3.7 – Таблиця «Користувачі»

id	name	guard_name	created_at	updated_at
1	user	web	2021-05-22 17:00:32	2021-05-22 17:00:32
2	admin	web	2021-05-22 17:02:11	2021-05-22 17:02:11

Рисунок 3.8 – Таблиця «Ролі»

role_id	model_type	model_id	▼ 1
1	App\Models\User	4	
2	App\Models\User	3	
1	App\Models\User	2	

Рисунок 3.9 – Таблиця відповідності користувачів до ролей

Авторизований в системі користувач повинен мати доступ до свого кабінету. Кабінет повинен містити в собі інформацію про білети користувача, також користувач повинен мати можливість завантажити свій квиток у форматі PDF на інформаційний носій (рис. 3.10). Для генерації документа зручно використовувати пакет `barryvdh/dompdf` [20]. Відображення інформації у кабінеті користувача відбувається завдяки скрипту, наведеному у додатку Д.

```

public function билет($ticket_id)
{
    $spectacle= new Spectacle;
    $spectacle_list= new Spectacle_list;
    $now = Carbon::now();

    $test = $spectacle::select('spectacles.name','spectacles.image','spectacles.id as spectacle_id',
    'spectacle_lists.time','tickets.created_at','places.row','places.place','sectors.price',
    'sectors.name as sector_name','zals.name as zal_name','tickets.viewer_id', 'tickets.id as ticket_id')
    ->join('spectacle_lists', 'spectacle_lists.id_vistavi', '=', 'spectacles.id')
    ->join('tickets', 'spectacle_lists.id', '=', 'tickets.spectacle_list_id')
    ->join('places', 'places.id', '=', 'tickets.place_id')
    ->join('sectors', 'places.sector_id', '=', 'sectors.id')
    ->join('zals', 'zals.id', '=', 'sectors.zal_id')
    ->join('users', 'users.id', '=', 'tickets.viewer_id')
    ->where('tickets.id', '=', $ticket_id)
    ->orderBy('spectacle_lists.time','asc')->get();

    foreach ($test as $el){
        $dt = Carbon::parse($el->time)->locale('uk')->isoFormat('D MMMM');

        $dt2 = Carbon::parse($el->time)->locale('uk')->isoFormat('HH:mm');
        $el->time = $dt;
        $el->test_time = $dt2;
    }

    $user=auth()->user()->id;
    view()->share(['data'=>$test,'user'=>$user]);
    $pdf = PDF::loadView('kabinet.ilet')->setOptions(['defaultFont' => 'sans-serif']);
    return $pdf->download('kabinet.ilet.pdf');
}

```

Рисунок 3.10 – Функція для генерації файлу квитка у форматі PDF

3.3 Використання Web-додатку

Функціонал Web-додатку поділений на три частини. Сторінки, які містять інформацію про театр, вистави театру, та його новини, доступ до даної частини додатку мають всі бажаючі. Кабінет користувача містить в собі інформацію про білети користувача, доступ до цієї сторінки мають лише зареєстровані користувачі. Для управління контентом додатку використовується адмін-панель, доступ до якої має лише адміністратор Web-додатку.

Головна сторінка Web-додатку містить найближчу виставу (рис. 3.11), календаря з подіями театру (рис. 3.12), афішу з найближчими виставами (рис. 3.13) та останні новини театру (рис. 3.14). Всі сторінки додатку мають навігаційне меню вгорі сторінки. У підвалі сайту розміщена контактна інформація театру.

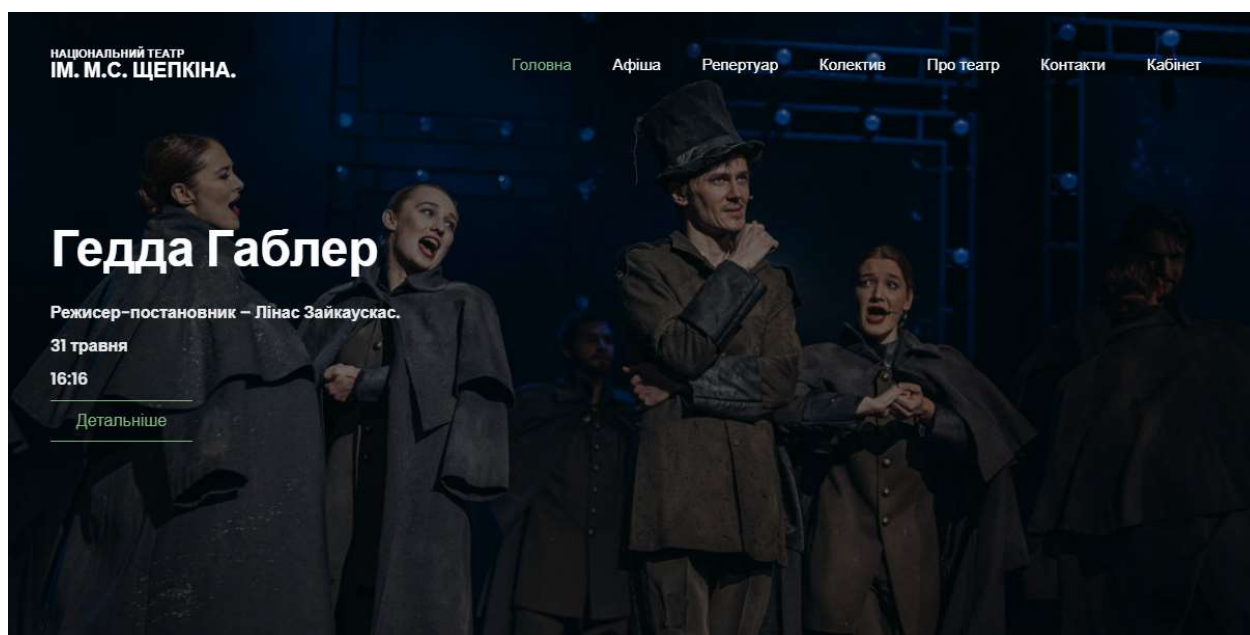


Рисунок 3.11 – Головна сторінка Web-додатку



Рисунок 3.12 – Головна сторінка Web-додатку



Рисунок 3.13 – Головна сторінка Web-додатку



Рисунок 3.14 – Головна сторінка Web-додатку

Після переходу до сторінки «Афіша» користувач має можливість перегляду афіші театру поточного місяця (рис. 3.15). При натисканні на назву вистави, користувачу буде відкрито сторінку з детальною інформацією про виставу.

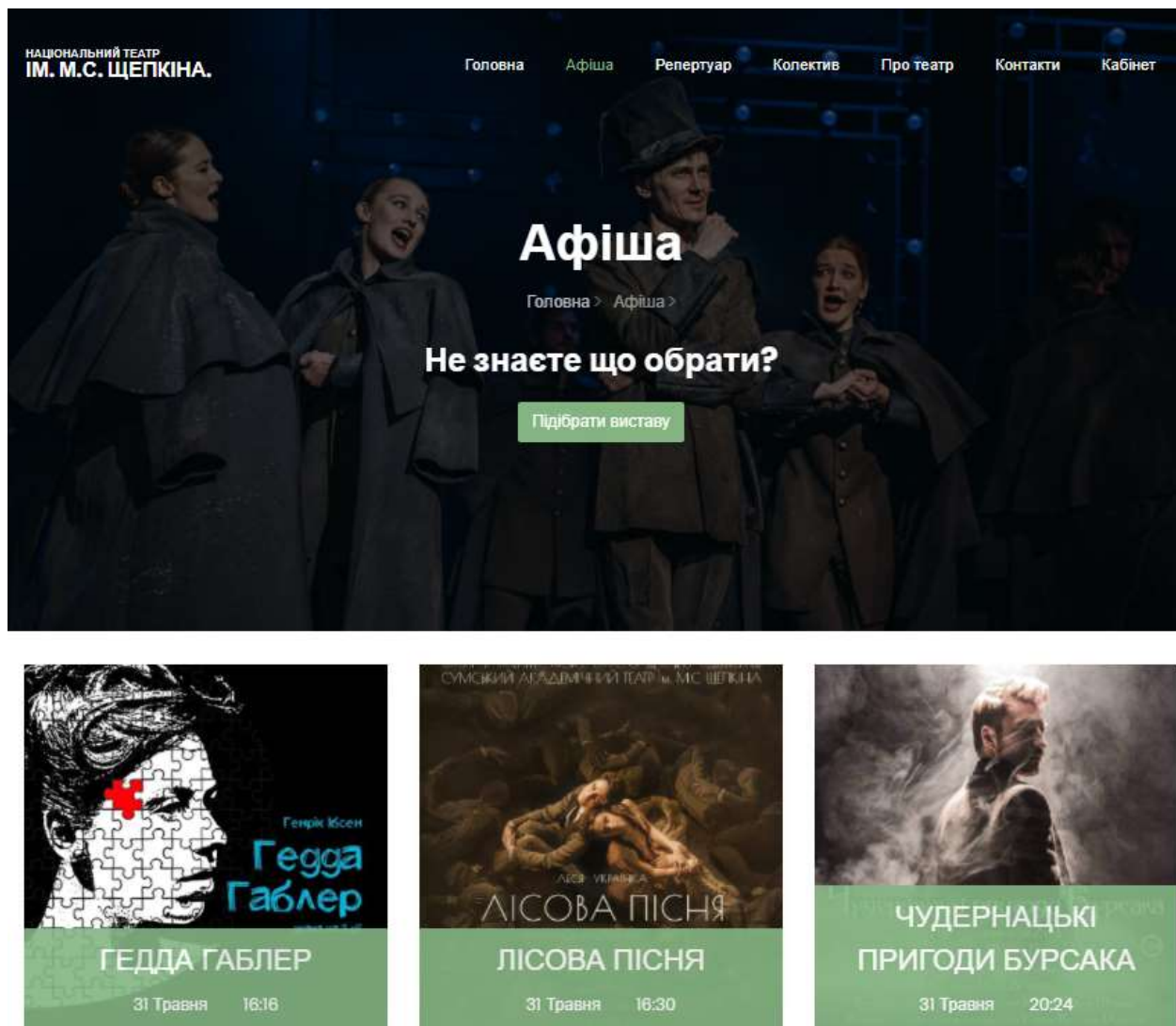


Рисунок 3.15 – Сторінка «Афіша»

Якщо користувач не може самостійно обрати виставу, він може скористатися сторінкою «Ситуативні потреби», яка містить їх перелік (рис. 3.16). При натисканні на одну з потреб, користувачу буде відкрито сторінку з переліком вистав, які задовольняють цій потребі.

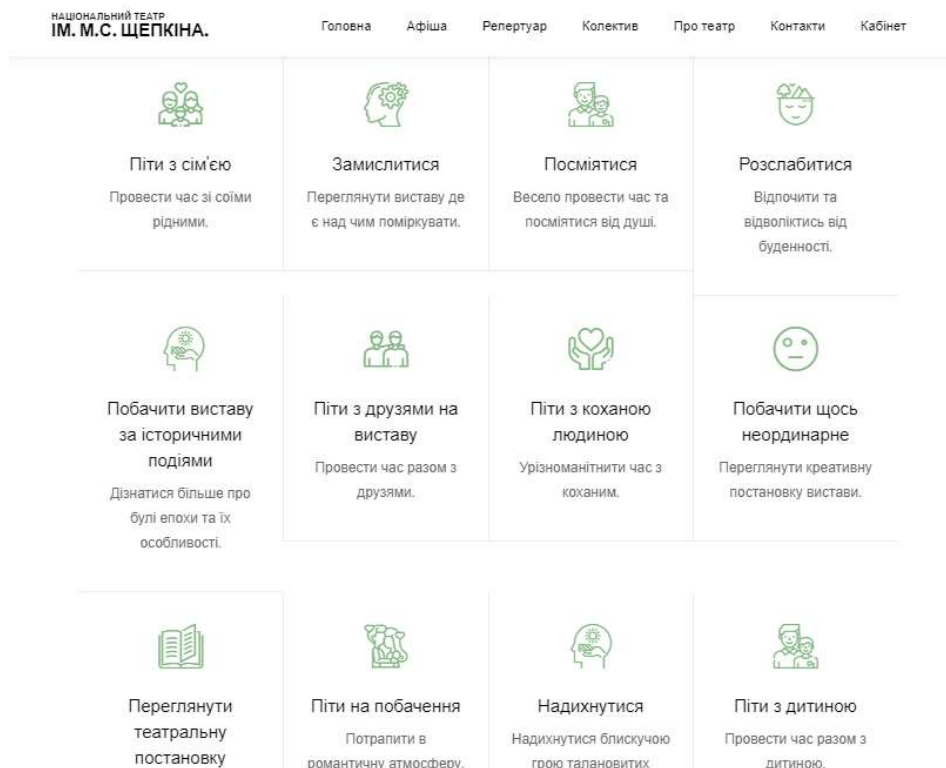


Рисунок 3.16 – Сторінка «Афіша»

Всі вистави театру, можна переглянути на сторінці репертуар (рис. 3.17). При наведенні курсору на картинку афіші вистави з'являється зелений фон, на якому зображено назву вистави. При натисканні на яку, користувачу буде відкрито сторінку з детальною інформацією про виставу (рис. 3.18). Якщо вистава буде ставитися в цьому місяці, сторінка матиме кнопку «Придбати квиток».

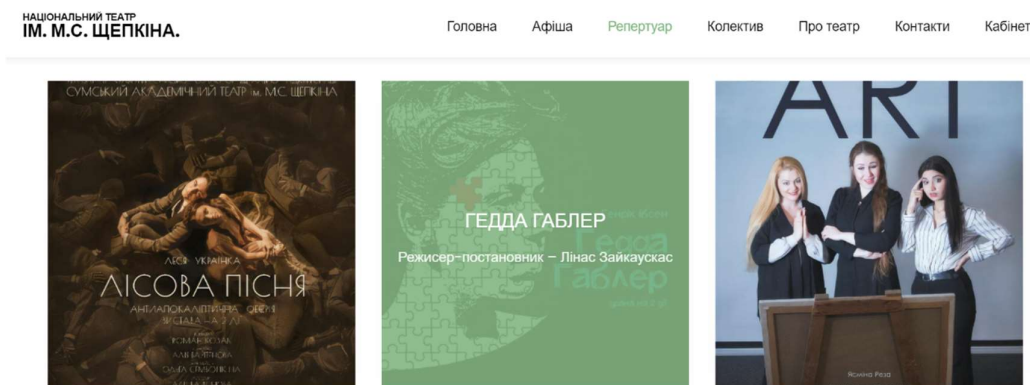



Рисунок 3.17 – Сторінка «Репертуар»

НАЦІОНАЛЬНИЙ ТЕАТР
ІМ. М.С. ЩЕПКИНА.

Головна Афіша Репертуар Колектив Про театр Контакти Кабінет



Гедда Габлер

Асистент режисера – Владислав
Калініченко Музичне оформлення – Лінас
Зайкаускас, Максим Криштоп П'єса «Гедда
Габлер» належить до найвитонченіших
шедеврів світової драматургії. «Гедда
Габлер» – загадкова п'єса – натяк.

Ситуативні потреби

[Пти з сміхом](#)

31 травня 16:16

[Купити квиток](#)

Це тонка психологічна драма, жорсткий, мікроскопічний аналіз взаємовідносин Чоловіка і Жінки у форматі «тілми для дорослих». Саме цей твір поклав початок поняттю «любовий трикутник» у всіх його безкомпромісних вимірах. Особу головної героїні шанувальники сиреситли «нікоюм Гамлетом». Її доля – трагічний і величний пошук індивідуальної свободи. У спектаклі діють три типи чоловіків – «каренінський», «моцартовський» та «обивательський». Серед жінок представлений тип «Настасії Гіліпівни» та «Сінгуроньки». Перше сценічне втілення п'єси отримала у Мюнхені, в 1891 році. І з того часу триумфально крокує різними країнами світу. Цей твір завжди викликав жвавий інтерес театральних режисерів різних поколінь, як, власне, і вся сценічна спадщина Генріка Ібсена. П'єси саме цього драматурга найчастіше прикрашають репертуарні афіші провідних театрів світу.

Дійові особи та виконавці:

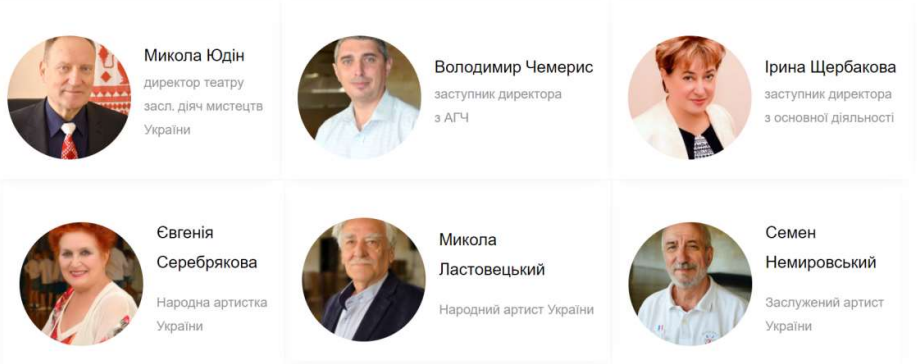
Гедда Габлер – Дана Каландарішвілі;
Йорган Тесман – Владислав Писарев;
Юліанна Тесман – Наталія Дехта;
Тевя Ельмстед – Дарія Мариніна;
Суддя Аесорс Брак – Анастасія Леміш;
Ейлерт Левбог – Владислав Калініченко;

Рисунок 3.18 – Сторінка вистави

Web-додаток має сторінки «Колектив» (рис. 3.19), «Про театр» (рис. 3.20) та «Контакти» (рис. 3.21), які містять відповідну інформацію про працівників, історію театру та контактні дані.

НАЦІОНАЛЬНИЙ ТЕАТР
ІМ. М.С. ЩЕПКИНА.

Головна Афіша Репертуар Колектив Про театр Контакти Кабінет









 Микола Юдін директор театру засл. діяч мистецтв України	 Володимир Чемерис заступник директора з АГЧ	 Ірина Щербакова заступник директора з основної діяльності
 Євгенія Серебрякова Народна артистка України	 Микола Ластовецький Народний артист України	 Семен Немировський Заслужений артист України

Рисунок 3.19 – Сторінка «Колектив»



Про театр

Сумський національний академічний театр драми та музичної комедії імені М.С. Щепкіна – провідний культурно-мистецький заклад області. За останній час перетворився на справжній форпост національної духовності та культури України. Безумовно, це велика заслуга всього колективу. Адже успіх прийшов завдяки неухильному удосконаленню репертуару та уважному ставленню до пропозицій глядачів.

Сумський Національний академічний театр – це багатoproфільний культурно-мистецький заклад, у якому з успіхом представлена класична драма, комедійна оперета сучасний авангард, різноманітні концертні програми. Це повноцінний театр з чітко визначеною стратегією зростання. У основі успішного розвитку – дух творчого експерименту, постійне вдосконалення професійної майстерності, застосування сучасних стилів і напрямків, діалог з глядачем.

Колектив театру перебуває у простійному пошуку нових ідей. За останні роки щепкінська сцена стала справжньою творчою лабораторією не лише для провідних вітчизняних, але й зарубіжних режисерів. Такий підхід цілком відповідає творчому кредо колективу, бо сучасний театр повинен випереджати смаки публіки, сприяти формуванню нових естетичних пріоритетів, знайомлячи глядача із кращими здобутками національного театрального мистецтва.

І сьогодні це відкриває можливості презентувати себе на міжнародному рівні, брати участь у різноманітних гастрольних виступах та фестивалних програмах.

Рисунок 3.20 – Сторінка «Про театр»

НАЦІОНАЛЬНИЙ ТЕАТР
ІМ. М.С. ЩЕПКИНА.

Головна Афіша Репертуар Колектив Про театр Контакти Кабінет

Контактна інформація

<p>Адреса здійснення діяльності театру: Україна, 40000, м. Суми, пл. Театральна, 1.</p>	<p>Адміністратор: +380 (542) 22-30-14 (з питань репертуару та розповсюдження квитків).</p> <p>Каса театру: +380 (542) 22-16-88 (з питань вартості квитків).</p>	<p>Email: shepkindrama@gmail.com</p> <p>Офіційний канал YouTube https://www.youtube.com/channel/UC4qIPgfm1ANQqJ8zh4vtg</p>
---	---	---

<p>Каса театру працює за наступним графіком:</p> <p>ПН: Вихідний;</p> <p>ВТ: з 12:00 до 18:30;</p> <p>СР – СБ: з 11:00 до 18:30;</p> <p>НД: з 13:00 до 18:30;</p> <p>Перерва з 14:00 до 14:30.</p> <p>Але, у будь-якому випадку, відкривається за годину до заходу.</p>	<p>Телефон "гарячої" лінії театру:</p> <p>(0542) 22-52-53, контактна особа – заступник директора з адміністративно-господарської частини Чемерис Володимир Миколайович.</p> <p>Графік особистого прийому громадян посадовими особами Національного театру ім. М.С. Щепкіна:</p> <p>Заступник директора з основної діяльності Щербанова Ірина Владиславівна – щовівторка з 15:00 до 18:30 – кабінет №3.</p> <p>Заступник директора з адміністративно-господарської частини Чемерис Володимир Миколайович – щосереда з 15:00 до 18:30 – кабінет №2.</p> <p>Головний інженер Нужиий Андрій Васильович – щоп'ятниця з 15:00 до 18:30 – кабінет №49.</p>
--	---

Рисунок 3.21 – Сторінка «Колектив»

Після перегляду інформації про виставу користувач має змогу переглянути сторінку перегляду вільних місць на виставу та обрати місце (рис. 3.22).

НАЦІОНАЛЬНИЙ ТЕАТР
ІМ. М.С. ЦЕПКИНА.

Головна Афіша Репертуар Колектив Про театр Контакти Кабінет

Чудернацькі пригоди Бурсака

31 травня 20:24

СЦЕНА

Ряд: 4 Місце:6

1 квиток · 200 грн. [Купити квиток](#)

The image shows a website interface for a theater. At the top left is the logo for the National Theatre named after M.S. Zhepkina. A navigation menu includes 'Home', 'Affiche', 'Repertoire', 'Company', 'About the Theatre', 'Contacts', and 'Cabinet'. The main content area features a poster for the play 'Чудернацькі пригоди Бурсака' (The Strange Adventures of Bursak) on the left. To the right, the play title is repeated, followed by the date and time '31 травня 20:24'. Below this is a 'СЦЕНА' (Stage) section with a grid of 18 seats (3 rows of 6). The selected seat is 'Ряд: 4 Місце:6', which is highlighted in green. At the bottom, it shows '1 квиток · 200 грн.' and a green button labeled 'Купити квиток'.

Рисунок 3.22 – Сторінка перегляду вільних місць на виставу

Для купівлі квитка користувач повинен авторизуватися (рис. 3.23). Авторизований користувач має змогу натиснути кнопку «Купити квиток». Оскільки в роботі не передбачено налаштування на сайті системи оплати, квиток одразу після натискання на кнопку «Купити квиток» вважається успішно оформленим і потрапляє до кабінету користувача (рис. 3.24 – 3.25).

Вхід Реєстрація Головна

Login

E-Mail Address

Password

Remember Me

[Login](#) [Forgot Your Password?](#)

The image shows a login form on a website. At the top right, there are links for 'Вхід' (Login), 'Реєстрація' (Registration), and 'Головна' (Home). The form itself is titled 'Login' and contains two input fields: 'E-Mail Address' with the value 'anna@gmail.com' and 'Password' with a masked value of seven dots. Below the password field is a checkbox labeled 'Remember Me'. At the bottom of the form are two buttons: a blue 'Login' button and a blue link 'Forgot Your Password?'.

Рисунок 3.23 – Сторінка авторизації користувача

Ваші квитки



Рисунок 3.24 – Сторінка «Кабінет»

Квиток №3



Лісова пісня

31 травня 16:30

Велика сцена: Партер

Ряд: 2

Місце: 12

Ціна: 150грн



Завантажити квиток

Рисунок 3.25 – Сторінка оформленого квитка

На сторінці перегляду квитка авторизований користувач має змогу завантажити квиток у форматі PDF (рис. 3.26) на електронний носій, натиснувши кнопку «Завантажити квиток».

Квиток №3

Лісова пісня

31 травня 16:30

Велика сцена: Партер

Ряд: 2

Місце: 12

Ціна: 150грн



Рисунок 3.26 – Квиток у форматі PDF

Сторінки Web-додатку мають адаптивний дизайн. Для демонстрації адаптивності сторінок, були обрані сторінки «Головна» та «Афіша».

Фрагмент сторінки «Головна» при ширині екрану від 768 до 991 зображено на рисунку 3.27.

Фрагмент сторінки «Афіша» при ширині екрану від 768 до 991 зображено на рисунку 3.28.

Фрагмент сторінки «Головна» при ширині екрану до 768 зображено на рисунку 3.29.

Фрагмент сторінки «Афіша» при ширині екрану до 768 зображено на рисунку 3.30.

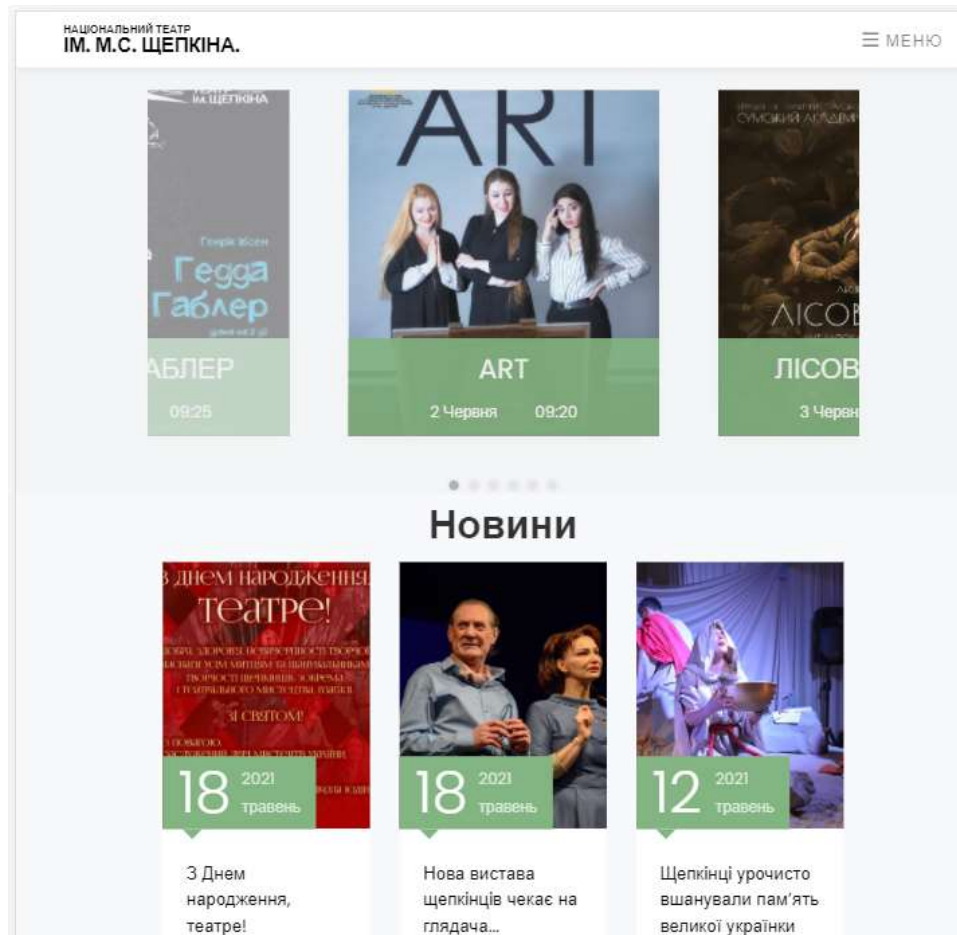


Рисунок 3.27 – Фрагмент сторінки «Головна»

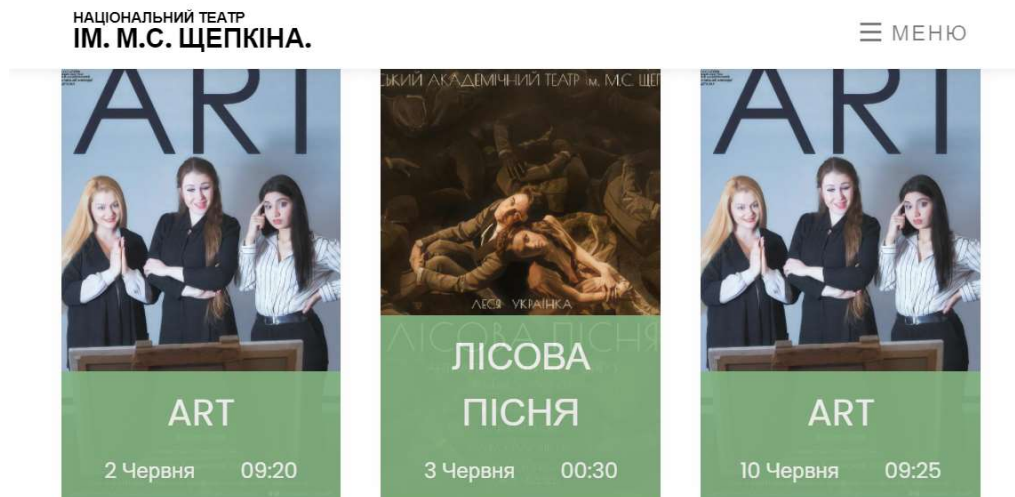


Рисунок 3.28 – Фрагмент сторінки «Афіша»

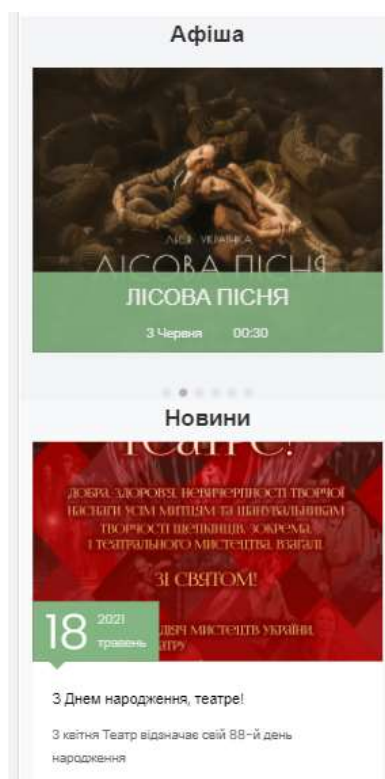


Рисунок 3.29 – Фрагмент сторінки «Головна»

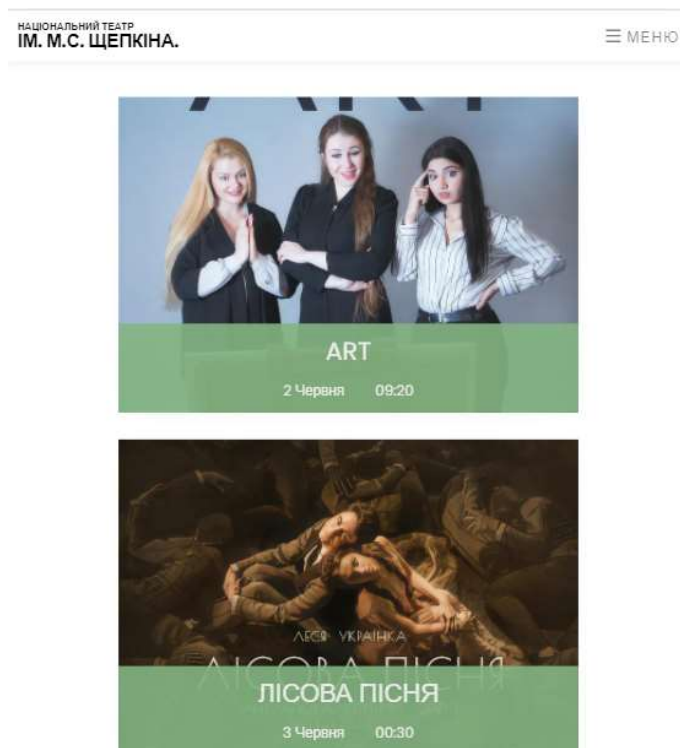


Рисунок 3.30 – Фрагмент сторінки «Афіша»

Для виконання додавання, редагування та оновлення контенту на сторінках Web-додатку існує адмін-панель, доступ до якої має лише авторизований користувач з правами адміністратора. Головна сторінка адмін-панелі (рис. 3.31) містить пункти меню для швидкого перегляду інформації про вистави, афішу театру та новини. Кожна сторінка також має навігаційне меню з лівої сторони.

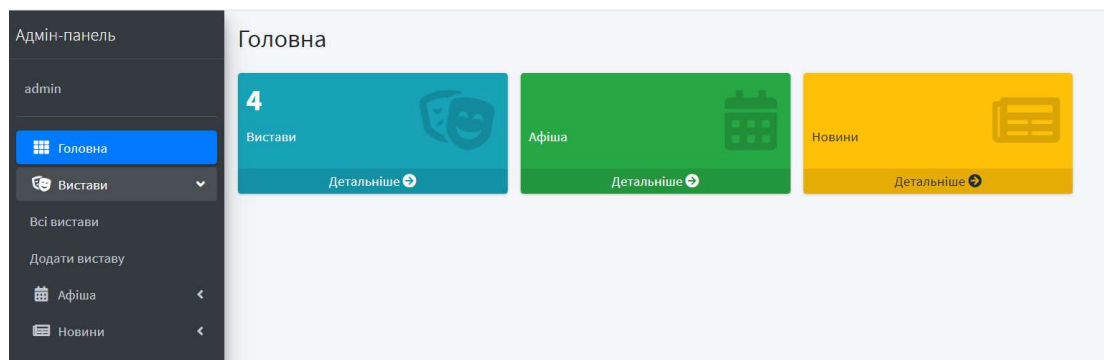


Рисунок 3.31 – Головна сторінка адмін-панелі Web-додатку

Для додавання інформації про вистави, афішу та новини театру необхідно перейти на відповідні пункти меню «Додати виставу», «Додати новини» які містять необхідні форми для додавання інформації (рис.3.32 – 3.34).

Рисунок 3.32 – Форма додавання вистави

Адмін-панель

admin

- Головна
- Вистави
- Афіша**
 - Переглянути афішу
 - Додати виставу**
 - Новини

Додати виставу

Вистава

ART

Дата та час:

mm/dd/yyyy --:-- --

Додати

Рисунок 3.33 – Форма додавання вистави до афіші

Адмін-панель

admin

- Головна
- Вистави
- Афіша
- Новини**
 - Всі новини
 - Додати новину**

Додати новину

Назва

Введіть назву вистави

Текст

File Edit View Insert Format

Paragraph **B** *I* [List Icons]

POWERED BY TINY

ФОТО НОВИНИ

Обрати зображення

Додати

Рисунок 3.34 – Форма додавання новини

Всі наявні вистави, новини та афішу можна переглянути сторінках «Всі вистави», «Всі новини» та «Переглянути афішу» (рис.3.35 – 3.37). Дані сторінки містять кнопку для переходу на сторінки редагування контенту та кнопку для його видалення.

Всі вистави







#	Назва	Режисер		
3	ART	Владислав Писарев		
2	Гедда Габлер	Лінас Зайкаускас		
1	Лісова пісня	Роман Козак		

Рисунок 3.35 – Сторінка «Всі вистави»

Афіша















#	Назва	Дата		
4	ART	2021-06-02 09:20:00		
9	Лісова пісня	2021-06-03 00:30:00		
5	ART	2021-06-10 09:25:00		
8	Чудернацькі пригоди Бурсака	2021-06-10 11:25:00		
10	ART	2021-06-18 03:30:00		
6	Гедда Габлер	2021-06-22 09:25:00		
7	Гедда Габлер	2021-06-29 01:30:00		

Рисунок 3.36 – Сторінка «Афіша»

Всі новини

#	Назва		
2	З Днем народження, театре!		 
1	Нова вистава щепкінців чекає на глядача...		 

Рисунок 3.37 – Сторінка «Всі новини»

Для редагування необхідно обрати контент який буде редагуватися та натиснути на кнопку редагування, після чого буде відкрито форми для редагування інформації (рис. 3.38 – 3.40).

Редагувати виставуART

Назва

ART

Опис

File Edit View Insert Format

Paragraph **B** *I* [List icons]

Вистава вдало поєднує класичні режисерські прийоми драматичного мистецтва із проблематикою сучасних людських взаємовідносин. Сюжет п'єси розгортається навколо трьох подруг Паули, Франсуази та Катрін. Їхня

POWERED BY TINY

Рисунок 3.38 – Сторінка редагування вистави

Редагувати виставу

Вистава

ART

Дата та час:

06/10/2021 10:43 AM

Зберегти

Рисунок 3.39 – Сторінка редагування вистави на афіші

Назва

З Днем народження, театре!

Текст

File Edit View Insert Format

Paragraph **B** *I* [List icons]

З квітня Театр відзначає свій 88-й день народження

POWERED BY TINY

Фото новини

/files/image_2.jpg

Обрати зображення

Зберегти

Рисунок 3.40 – Сторінка редагування новини

ВИСНОВКИ

Сьогодні інформаційні технології відіграють важливу роль в нашому житті. Після важкого дня дозвілля є гарним варіантом відпочинку, наприклад перегляд фільмів або театральних вистав. Майже всі сфери людської діяльності було комп'ютеризовано, театр не став виключенням. На даний час перегляд афіші, вибір вистави для перегляду, придбання квитка на виставу можна зроби онлайн. Однак Сумський національний академічний театр драми та музичної комедії імені М. С. Щепкіна надає користувачу лише можливість перегляду афішу.

В ході виконання роботи були сформульовані мета і задачі, проведено аналіз аналогів, в процесі якого були визначені переваги та недоліки аналогічних додатків театру, для уникнення недоліків в створеному додатку, та можливих варіантів покращення додатку. Були сформульовані функціональні вимоги: можливість купівлі квитків, можливість підбору вистави за ситуативними потребами, можливість перегляду історії відвіданих вистав, адміністрування Web-додатку, перегляд афіші, інформації про виставу та митців театру, можливість перегляду інформації про театр та його новини. Також в ході роботи були обрані інструменти для реалізації. Серед засобів реалізації було обрано використання фреймворку Laravel та Bootstrap. Було проведено моделювання контекстної діаграми в нотації IDEF0 та її декомпозицію на перший та другий рівні. Були створені діаграми варіантів використання додатку, діаграма послідовності та діаграма діяльності. Було виконано проектування моделі бази даних.

Метою роботи було створення Web-додатку театру, для досягнення якої необхідно було створити можливість купівлі квитків. Дане рішення покликане спростити процес купівлі квитків, та мінімізувати контакт з персоналом театру, що є актуальним питтям під час пандемії.

Процес підбору вистави для перегляду також більше не є проблемою, оскільки додаток має перелік ситуативних потреб, які спрощують процес вибору

вистави.

Оскільки поставлені задачі були повністю виконані, в результаті роботи було створено Web-додаток підтримки діяльності Сумського національного академічного театру драми та музичної комедії імені М. С. Щепкіна.

Результати роботи були оприлюднені на науково-технічній конференції ІМА-2021. Тези доповіді розміщені у додатку Е.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бадьин, А.В. Нова ринкова ніша. Від ідеї до створення нового затребуваного продукту / А.В. Бадьин. - М.: Гаряча лінія - Телеком, 2017. - 103 с.
2. Крушельницька, Я.В. Фізіологія и психологія праці / Я.В. Крушельницька. Навч. посібник. - К.: КНЕУ, 2000. - 232 с.
3. Розробка веб-додатків, мобільних додатків та порталів [Електронний ресурс] – Режим доступу до ресурсу: <https://ittel.com.ua/informacijni-technologiyi/rozrobka-mobilnih-dodatktiv/>
4. Дунаєв, Вадим HTML, скрипти і стилі / Вадим Дунаєв. - М.: БХВ-Петербург, 2015. - 816 с.
5. Что такое Bootstrap? [Електронний ресурс] – Режим доступу до ресурсу: <https://webformyself.com/chto-takoe-bootstrap/>
6. Колісниченко, Д.Н. PHP і MySQL. Розробка веб-додатків / Д.М. Колісниченко. - М.: БХВ-Петербург, 2017. - 640 с.
7. Зандстра, М. PHP. Об'єкти, шаблони і методики програмування. / Метт Зандстра. - К.: Вільямс, 2019. - 576 с
8. Laravel — лидер среди PHP фреймворков, одобренный разработчиками [Електронний ресурс] – Режим доступу до ресурсу: <https://webformyself.com/laravel-lider-sredi-php-frejmworkov-odobrennyj-razrabotchikami/>
9. Почему Laravel — один из лучших PHP-фреймворков для стартапов и энтерпрайза [Електронний ресурс] – Режим доступу до ресурсу: <https://ru.hexlet.io/blog/posts/pochemu-laravel-odin-iz-luchshih-php-freymworkov-dlya-startapov-i-enterprayza>
10. Ярушкіна, Н. Проектування баз даних для економічних інформаційних систем в середовищі сервера даних ORACLE 7.3 / Н. Ярушкіна, Т. МЕРКУЛОВА. - Ульяновськ: УЛГТУ, 2002. - 160 с.

11. Методология IDEF0 [Электронный ресурс] – Режим доступа до ресурсу: <https://itteach.ru/bpwin/metodologiya-idef0>
12. Буч Г. Мова UML. Посібник користувача. / Г. Буч, Д. Рамбо, І. Якобсон. - М .: ДМК Пресс, 2015. - 496 с.
13. Веллинг, Л. Разработка веб-приложений с помощью PHP и MySQL / Люк Веллинг - М .: Вильямс, 2016. - 768 с.
14. Галяветдінов, Н. Основи автоматизованого проектування виробів і технологічних процесів / Н.Галяветдінов, Р. Сафін, Р. Хасаншин, П. Кайнов. - Казань: КНІТУ, 2016.
15. Маркін, А. В. Побудова запитів і програмування на SQL. Навчальний посібник / А.В. Маркін. - М .: Діалог-Міфи, 2014. - 384 с.
16. Эспозито, Д. Разработка современных веб-приложений: анализ предметных областей и технологий / Дино Эспозито. - М .: Вильямс , 2017. - 464 с.
17. Stauffer, M. Laravel: Up & Running: A Framework for Building Modern PHP Apps 2nd Edition / Matt Stauffer. - O'Reilly Media, 2019. – 544 p.
18. Minasian, J. Laravel: Master Laravel For Beginners: Get From Zero To Proficiency In The Laravel Framework / Jeromy Minasian. - Independently published, 2021. – 109 p.
19. Laravel-permission [Электронный ресурс] – Режим доступа до ресурсу: <https://spatie.be/docs/laravel-permission/v4/introduction>
20. dompdf [Электронный ресурс] – Режим доступа до ресурсу: <https://dompdf.github.io/>

ДОДАТОК А

ТЕХНІЧНЕ ЗАВДАННЯ

**на розробку інформаційної системи
«Web-додаток підтримки діяльності Сумського національного академічного
театру драми та музичної комедії
імені М. С. Щепкіна»**

А Призначення й мета створення Web-додатку

А.1 Призначення Web-додатку

Основна мета проекту полягає у розробці Web-додатку театру «Національний театр ім. М. С. Щепкіна». Призначенням якого є демонстрація основної інформації про заклад та автоматизація процесів збуту квитків.

А.2 Мета створення Web-додатку

Збільшення кількості глядачів театру за рахунок використання нового Web-додатку. Спрощення роботи персоналу з адміністрування додатку.

А.3 Цільова аудиторія

Цільовою аудиторією є працівники закладу, глядачі театру, жителі та гості міста Суми.

В Вимоги до Web-додатку

В.1 Вимоги до Web-додатку в цілому

В.1.1 Вимоги до структури й функціонування Web-додатку

Web-додаток має бути доступним в мережі Інтернет під доменним іменем musicdrama.com.ua. Розроблювальна програма повинна бути додатком, клієнтська частина якого є Web-сайтом з набором елементів керування, які забезпечують виконання функціональних можливостей

В.1.2 Вимоги до персоналу

Від персоналу не має вимагатися особливих технічних навичок для підтримки й експлуатації Web-додатку, окрім загальних навичок роботи з персональним комп'ютером і стандартним Web-браузером.

В.1.3 Вимоги до збереження інформації

Уся інформація надана у Web-додатку повинна зберігатися у базі даних реалізованій засобами системи управління базами даних MySQL.

В.1.4 Вимоги до розмежування доступу

Розроблюваний Web-додаток має бути загальнодоступним.

Відповідно до прав доступу до інформації у Web-додатку, усіх користувачів можна поділити на гостей, адміністратора та відвідувач.

Гості сайту можуть переглядати усі сторінки Web-додатку, ознайомитись з афішою, репертуаром, колективом та новинами театру.

Відвідувач крім запропонованих гостю функцій може зайти до свого кабінету, придбати квиток та передивитися історію своїх відвідувань театру.

Доступ до кабінету здійснюється після реєстрації користувача, за відповідним логіном та паролем.

Адміністратор може редагувати наповнення Web-додатку.

Доступ до панелі адміністратора здійснюється за унікальним логіном і паролем.

В.2 Структура Web-додатку

В.2.1 Загальна інформація про структуру Web-додатку

Структура Web-додатку являє собою набір сторінок, які також є пунктами головного меню.

Такими розділами є:

Головна – на сторінці зображені меню, яке відповідно дублюється на кожній сторінці сайту та карусель з афішою театру та сервісом підбору квитків. Поряд з афішою знаходиться календар подій театру. На сторінці також знаходяться новини театру.

Афіша – інформація про найближчі вистави.

Репертуар – інформація про вистави театру.

Колектив – інформація про творчий колектив театру.

Про театр – інформація про театр та його історію.

Контакти – список всіх можливих видів зв'язку з адміністрацією та касою театру, графік особистого прийому громадян посадовими особами Національного театру ім. М.С. Щепкіна .

Кабінет – особистий кабінет користувача.

В.2.2 Навігація

Для навігації, у шапці буде створена система меню. Меню необхідне для швидкого переміщення користувача до усіх доступних сторінок. Меню буде відображатися на всіх сторінках, щоб відвідувач міг в будь-який момент часу перейти до будь-якої сторінку Web-додатку.

В.2.3 Наповнення Web-додатку (контент)

Для управління контентом Web-додатку буде створена система CMS.

Заповнення та редагування контенту Web-додатку має бути зроблено через панель керування, з використанням бази даних.

Всю інформацію для наповнення Web-додатку має надавати театр.

В.2.4 Дизайн та структура додатку

Стиль Web-додатку має бути сучасним, приємним для сприйняття, у якості основних кольорів було запропоновано використати бежевий, коричневий та темно-зелений відтінки так як ці кольори переважають в оформленні залів театру.

Види і розміри шрифтів повинні бути комфортними для перегляду користувачами. Усі відео та фото мають бути чіткими та якісними. Розташування елементів на головній сторінці Web-додатку схематично зображено на рисунку А.1.

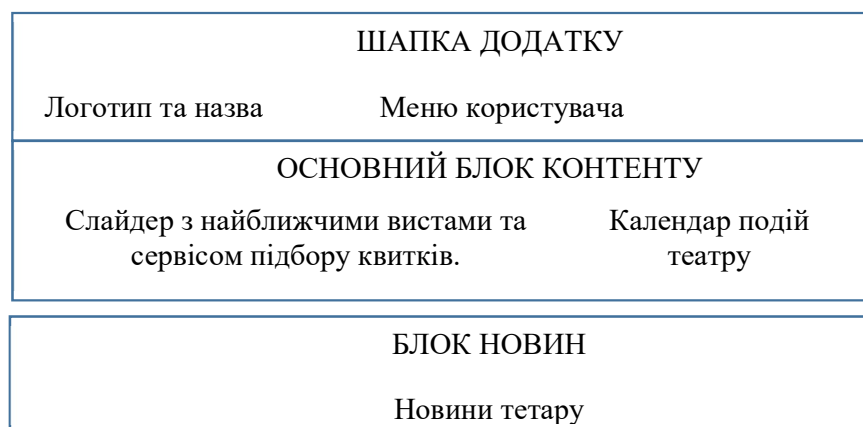


Рисунок А.1 – Схема головної сторінки

В.2.5 Система навігації (карта Web-додатку)

Карта Web-додатку зображена на рисунку А.2.

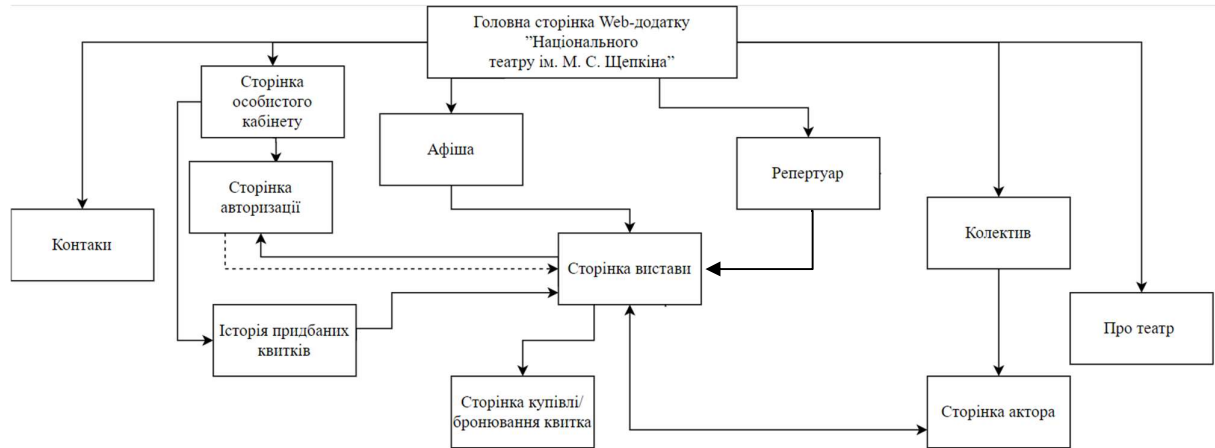


Рисунок А.2 – Карта Web-додатку

В.3 Вимоги до функціонування системи

В.3.1 Потреби користувача

Потреби користувача, визначені на основі рішення замовника, представлені у таблиці А.1.

Таблиця А.1 – Потреби користувача

ID	Потреби користувача	Джерело
UN-01	Онлайн-купівля квитка на обрану виставу	Глядач
UN-02	Перегляд своєї історії купівлі квитків	Глядач
UN-03	Перегляд опису вистав	Глядач
UN-04	Перегляд афіші	Глядач
UN-05	Інформація про працівників театру	Глядач
UN-06	Перегляд новин театру	Глядач
UN-07	Перегляд інформації про театр	Глядач
UN-08	Можливість підбору вистави за вподобаннями	Глядач
UN-09	Редагування даних	Адміністратор

В.3.2 Функціональні вимоги

На основі потреб користувача були визначені такі функціональні вимоги:

- реєстрація та авторизація користувачів;
- пошук інформації про виставу;
- пошук інформації про митців театру;
- пошук інформації про театр та новини;
- відображення результатів пошуку на Web-сторінці;
- можливість купівлі квитка шляхом заповнення онлайн-форми;
- можливість підбору вистави за вподобаннями;
- адміністрування інформації про користувачів, видалення, зміну користувацької групи, надання користувацьких прав;
- адміністрування інформації про театр, вистави, новини.

В.3.3 Системні вимоги

Даний розділ визначає, розподіляє та вказує на системні вимоги, визначені розробником. Їх перелік наведений в таблиці А.2.

Таблиця А.2 – Системні вимоги

ID	Системні вимоги	Пріоритет	Опис
SR-01	Наявність модуля купівлі квитків	М	Надає можливість глядачу купити квиток
SR-02	Наявність модуля авторизації користувача	М	Надає можливість глядачу авторизуватися на сайті
SR-03	База даних із купленими квитками	М	Надає можливість відображати зайняті місця в залі на ту чи іншу виставу

Продовження таблиці А.2

ID	Системні вимоги	Пріоритет	Опис
SR-04	База даних з контентом	М	Відповідає за заповнення Web-додатку контентом
SR-05	База даних з цінами на вистави, відповідно до місця в залі	М	Надає можливість подальшого відображення цін
SR-06	Наявність модуля підбору вистави	С	Надає можливість глядачу підібрати виставу за вподобаннями
SR-07	Панель адміністратора	М	Відповідає за подальше наповнення та редагування контенту адміністратором

Умовні позначення в таблиці А.2:

Must have (М) – вимоги, які повинні бути реалізовані в системі;

Could have (С) – вимоги, які можуть бути реалізовані, але вони не є центральною ціллю проекту.

В.4 Вимоги до видів забезпечення

В.4.1 Вимоги до лінгвістичного забезпечення

Web-додаток має бути виконаний українською мовою.

В.4.2 Вимоги до програмного забезпечення

Програмне забезпечення клієнтської частини повинне задовольняти наступним вимогам: Web-браузер: Internet Explorer 7.0 і вище, або Firefox 3.5 і вище, або Opera 9.5 і вище, або Safari 3.2.1 і вище, або Chrome 2 і вище.

С Склад і зміст робіт зі створення Web-додатку

Докладний опис етапів роботи зі створення Web-додатку наведено в таблиці А.3.

Таблиця А.3 – Етапи створення Web-додатку

№	Склад і зміст робіт	Строк розробки (у робочих днях)
1	Постановка цілей необхідних для досягнення певного результату	5 днів
2	Складання технічного завдання	5 днів
3	Підготовка прототипу	5 днів
4	Створення макету дизайну Web-додатку	4 дні
5	Верстка	15 днів
6	Робота над модулями для Web-додатку	20 днів
7	Робота з контентом	2 дні
8	Розміщення контенту та каталогів з фото у Web-додатку	1 день
9	Перевірка працездатності Web-додатку	2 дні
10	Завершення роботи	1 день
	Загальна тривалість робіт	60 днів

Д Вимоги до складу й змісту робіт із введення Web-додатку в експлуатацію

Зараз «Національний театр ім. М. С. Щепкіна» має свій сайт, відповідно має доменне ім'я та розміщений на хостингу. Відповідно на існуючий хостинг необхідно перенести новий Web-додаток і наповнення бази даних з подальшою їх доробкою.

ДОДАТОК Б

ПЛАНУВАННЯ РОБІТ

Щоб проєкт було успішно реалізовано, він повинен мати зрозумілу мету. Для цього доцільним є використання технології SMART. Сенс якої приховано в назві техніки.

S – Конкретність і ясність. Створення інформаційно-аналітичного Web-додатку театру «Національний театр ім. М. С. Щепкіна».

M – вимірюваність. Створити Web-додаток за 90 днів, з використанням мінімальної кількості ресурсів.

A – досяжність. Для реалізації поставленої задачі необхідні наступні навички: знання CSS та HTML, таких мов програмування як: PHP, JavaScript, та досвід роботи з реляційною базою даних MySQL.

R – важливість. Створення Web-додатку театру вплине на комфортність процесу реалізації квитків як для співробітників театру, так і для глядача, а отже відбудеться збільшення цільової аудиторії.

T – обмеженість в часі. Web-додаток повинен бути повністю реалізований в термін узгоджений між виконавцем та замовником – 90 днів.

Опис функціонування продукту проєкту

На основі поставленої мети проєкту було визначено основні рекомендації до функціоналу додатку:

- для користувача додатку обов'язковою умовою є доступ до мережі Інтернет.
- створюваний продукт має відповідати вимогам кросбраузерності та адаптивності, щоб користувачі могли користуватися ним за допомогою будь-якого пристрою (телефон, персональний комп'ютер), використовуючи браузер на їх вибір (Google Chrome, Mozilla Firefox, Internet Explorer).

– додаток повинен містити інформацію про діяльність театру та новини, можливість купівлі квитків і функцію підбору вистави за вподобаннями глядача.

Планування змісту структури робіт ІТ—проекту.

Work Breakdown Structure або Структура декомпозиції робіт – розбиття на дрібні завдання всього обсягу робіт до тих пір, поки не стане зрозуміло, як і за який час кожне із завдань можна реалізувати.

Основна мета – визначити і структурувати весь зміст проекту. Іншими словами WBS повинна містити всі роботи, заплановані в проекті.

Кожен наступний рівень декомпозиції забезпечує послідовну деталізацію змісту проекту, що дозволяє здійснювати оцінку виконаних обсягів робіт, освоєних грошей і виконання за термінами.

На нижніх рівнях пакетах робіт відповідають порівняно менші обсяги робіт. Це спрощує оцінку відсотка виконання і дає можливість більш чітко визначати дії, необхідні для досягнення цілей проекту. Даний підхід декомпозиції робіт формує необхідну основу для визначення вимірних показників (трудомісткості, вартості), а також дозволяє з високим ступенем достовірності говорити про те, що цілі, пов'язані з даними пакетом робіт можуть і будуть досягнуті. На рисунках Б.1 представлена WBS-структура Web-додатку підтримки діяльності театру.

Планування структури організації.

Наступним етапом в створенні додатку є розробка організаційної структури проекту або OBS. Організаційна структура проекту – є графічним відображенням учасників проекту (фізичних та юридичних осіб) та їхніх відповідальних осіб, залучених до реалізації проекту. На рисунку Б.2 представлено організаційну структуру проекту.

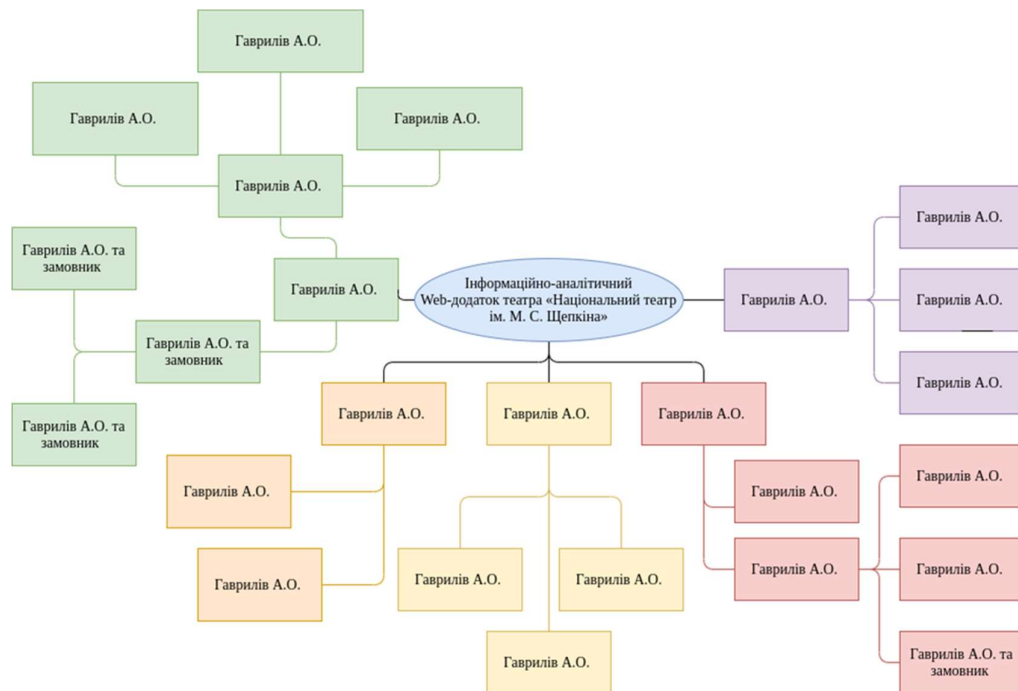


Рисунок Б.2 – OBS-структура Web-додатку

Поєднання OBS та WBS дає нам матрицю відповідальності. Вертикаль якої являє собою види робіт, а горизонталь – виконавців цих робіт. Крім цього, для того щоб показати ступінь відношення виконавця до певного виду роботи використовують умовне позначення на перетині роботи та виконавця:

- Р – бере участь у розробці;
 - В – відповідальний виконавець;
 - У – особа, що узгоджує вихідний результат.
- У таблиці Б.1 зображено матрицю відповідальності проекту.

Таблиця Б.1 – Матриця відповідальності проекту

№	Задача	Гаврилів А.О.	Нагорний В.В.	Замовник	Рецензент
1	Визначення проблеми яку допомагає вирішити створюваний додаток	В	Р		
2	Визначення цільової аудиторії	В			
3	Визначення цілей, які мають бути досягнуті, шляхом створення даного програмного продукту	В			
4	Розробка технічного завдання	В	Р	Р	
5	Затвердження технічного завдання	Р		В	
6	Розробка функціональної архітектури додатку	В	У	У	
7	Формування системної архітектури додатку	В	У	У	
8	Створення Front-end частини додатку	В	У	У	У
9	Створення Back-end частини додатку	В	У		У
10	Проектування та підключення БД	В	У		У
11	Функціональне тестування	В	У		
12	Тестування інтерфейсу	В	У	Р	
13	Тестування безпеки	В	У		
14	Тестування ергономічності	В	У	Р	
15	Створення супровідної документації	В	У		
16	Розміщення додатку на хостингу	В			
17	Проведення навчання персоналу	В		Р	

Розробка PDM мережі

Наступним етапом після створення матриці відповідальності проекту є створення системи управління даними продукту, або PDM (Product Data Management). За допомогою PDM-систем здійснюється відстеження великих масивів даних, необхідних на етапах проектування, реалізації, а також підтримки експлуатації, супроводу проектів.

Головне призначення PDM-систем – управління інформацією та полегшення доступу до даних про виріб протягом усього його життєвого циклу. Позитивний ефект досягається завдяки можливості об'єднати всі дані про виріб в єдину логічну систему. В результаті такого об'єднання всі, хто бере участь в розробці виробу, отримують розподілений авторизований доступ до проектної інформації та управління процесами проектування.

Календарний графік з виконання роботи представлений на рисунку Б.3-Б.4.

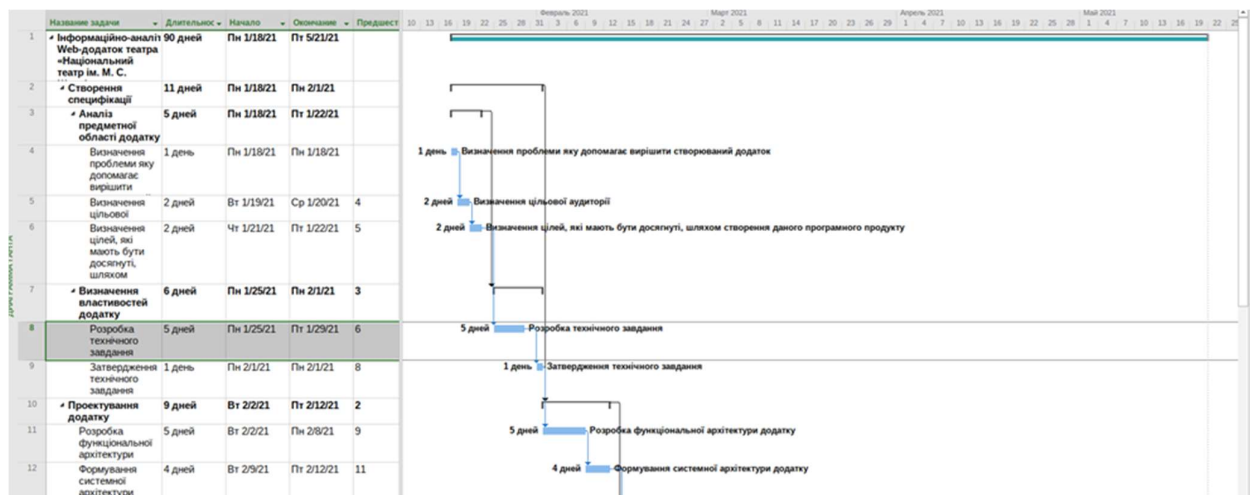


Рисунок Б.3 – Календарний графік

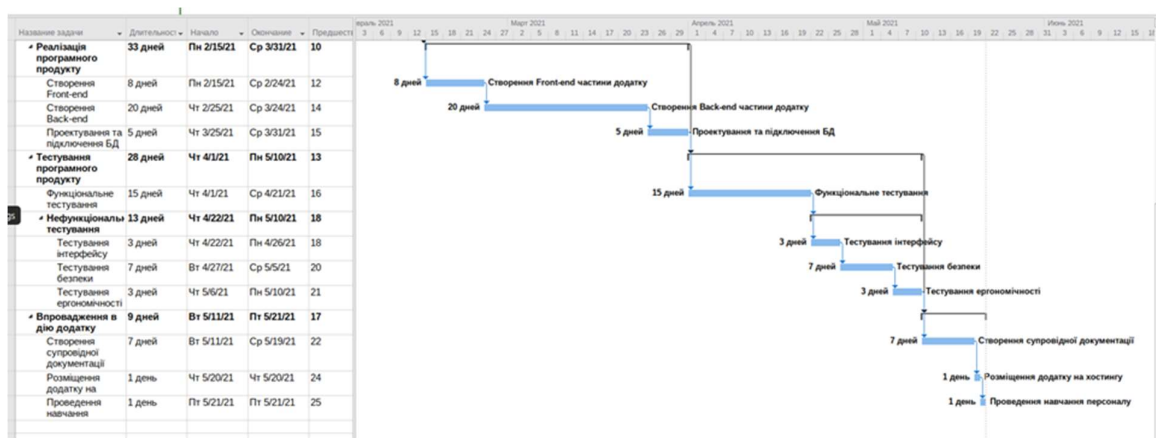


Рисунок Б.4 – Календарний графік

Управління ризиками проекту

Будь-який проект має ступінь небезпеки для його успішного здійснення, він називається – ризик. Поняття ризику характеризується невпевненістю, що пов'язана з можливістю виникнення в ході реалізації проекту несприятливих ситуацій і наслідків. Управління ризиком являє собою процес пов'язаний з аналізом ризиків (їх ідентифікація та прийняття рішення), задля максимізації позитивних та мінімізації негативних наслідків появи ризикових подій.

У таблиці Б.2 проведена класифікація ризиків.

Таблиця Б.2 – Ризики проекту

Ризик	Назва ризику	Опис ризику
1	Невиконання роботи в поставлені терміни	Недотримання термінів завершення робіт
2	Зміна вимог	Зміна об'єму та терміну роботи замовником
3	Низька якість вхідного матеріалу	Матеріал для наповнення сайту має низьку якість
4	Несумісність з апаратним забезпеченням замовника	Апаратне забезпечення замовника має недостатню потужність
5	Підтримка проекту	Замовник не впорається самостійно з підтримкою сайту

У таблицях Б.3 – Б.4 описана реєстрація ризиків та матриця впливу відповідно.

Таблиця Б.3 – Реєстрація ризиків

Ризик	Вірогідність виникнення	Ступінь впливу	Значення ризику
1	1	2	2
2	3	3	9
3	2	3	6
4	1	3	3
5	2	2	4

Таблиця Б.4 – Матриця впливу

Вірогідність виникнення	Матриця впливу		
3			2
2		5	3
1		1	4
Ступінь впливу	1	2	3

Після оцінки ризиків було складено план реагування на ризики, кий зображено у таблиці Б.5.

Таблиця Б.5 – Реакція на ризики проекту

Ризик	Назва ризику	Реакція на ризик
1	Невиконання роботи в поставлені терміни	Консультація з керівником проекту. Обговорення з замовником можливості спрощення вимог або зміни термінів.
2	Зміна вимог	Зміна термінів завершення проекту
3	Низька якість вхідного матеріалу	Самостійне створення та пошук матеріалів
4	Несумісність з апаратним забезпеченням замовника	Модифікація програми. Модернізація обладнання.
5	Підтримка проекту	Проведення додаткового навчання з персоналом, відповідальним за підтримку сайту. Модифікація програми в бік спрощення його підтримки.

ДОДАТОК В

ЛІСТИНГ КОДУ ГОЛОВНОЇ СТОРІНКИ ДОДАТКУ

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <link rel="shortcut icon" href="/images/logo20.png" type="image/png">
    <title>Національний театр ім. М.С. Щепкіна.</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <link rel="stylesheet" href="/css/eventCalendar.css">
    <link rel="stylesheet" href="/css/eventCalendar_theme_responsive.css">
    <link href="https://fonts.googleapis.com/css?family=Poppins:300,400,500,600,700,800,900"
rel="stylesheet">
    <link
href="https://fonts.googleapis.com/css?family=Playfair+Display:400,400i,700,700i,900,900i&display=sw
ap" rel="stylesheet">
    <link rel="stylesheet" href="/css/open-iconic-bootstrap.min.css">
    <link rel="stylesheet" href="/css/animate.css">

    <link rel="stylesheet" href="/css/owl.carousel.min.css">
    <link rel="stylesheet" href="/css/owl.theme.default.min.css">
    <link rel="stylesheet" href="/css/magnific-popup.css">
    <link rel="stylesheet" href="/css/aos.css">
    <link rel="stylesheet" href="/css/ionicons.min.css">

    <link rel="stylesheet" href="/css/flaticon.css">
    <link rel="stylesheet" href="/css/icomoon.css">
    <link rel="stylesheet" href="/css/style.css">
    <link rel="stylesheet" href="/css/theme-basic.css" />
    <link rel="stylesheet" href="/css/theme-glass.css" />
    <script src="/js/bundle.js"></script>
    <script src="/js/jquery.min.js"></script>
  </head>
  <body>

    <nav class="navbar px-md-0 navbar-expand-lg navbar-dark ftco_navbar bg-dark ftco-navbar-
light" id="ftco-navbar">

      <div class="container-fluid px-md-5">
        <a class="navbar-brand" href="index"><span>Національний театр </span>ім. М.С.
Щепкіна.</a>
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#ftco-nav" aria-controls="ftco-nav" aria-expanded="false" aria-label="Toggle navigation">
          <span class="oi oi-menu"></span> Меню
        </button>

        <div class="collapse navbar-collapse" id="ftco-nav">
          <ul class="navbar-nav ml-auto">
            <li class="nav-item active"><a href="index" class="nav-link">Головна</a></li>
            <li class="nav-item"><a href="afisha" class="nav-link">Афіша</a></li>
            <li class="nav-item"><a href="repertuar" class="nav-link">Репертуар</a></li>
            <li class="nav-item"><a href="kolektiv" class="nav-link">Колектив</a></li>
            <li class="nav-item"><a href="about" class="nav-link">Про театр</a></li>
            <li class="nav-item"><a href="contact" class="nav-link">Контакти</a></li>
            <li class="nav-item"><a href="home" class="nav-link">Кабінет</a></li>
          </ul>
        </div>
      </div>
    </nav>

```



```

@foreach ($data as $el)
  <div class="hero-wrap js-fullheight" style="background-image: url('/images/bg_1.jpg');" data-
stellar-background-ratio="0.5">
    <div class="overlay"></div>
    <div class="container-fluid px-md-5">
      <div class="row no-gutters slider-text js-fullheight align-items-center justify-content-
start" data-scrollax-parent="true">
        <div class="col-md-6 ftco-animate " >
          <h1 class="mb-4">{{ $el->name }}</h1>
          <h2 class="subheading">Режисер-постановник - {{ $el->producer }}. </h2>
          <h2 class="subheading">{{ $el->time }}</h2>
          <h2 class="subheading">{{ $el->test_time }}</h2>
          <div class="post-readmore"><a href="{{ route('vistava_afisha', $el-
>id) }}">Детальніше</a></div>
        </div>
      </div>
    </div>
  </div>
  @break;
@endforeach ()

```

```

<section class="ftco-section ftco-no-pt ftco-no-pb">
  <div class="container">
    <div class="row d-flex">
      <div class="col-md-6 d-flex">
        <div id="eventImage" class="img img-video d-flex align-self-stretch
align-items-center justify-content-center justify-content-md-end" style="background-
image:url(images/about.jpg); margin-top: 2rem !important; margin-bottom: 2rem !important;">
        </div>
      </div>
      <div class="col-md-6 px-5 py-5">
        <div class="row justify-content-start pt-3 pb-3">
          <div class="col-md-12 heading-section ftco-animate">
            <div id="color-calendar" style="max-height: 80%"> </div>
            <p class="subheading" style="line-height: 1.5; font-weight: 400;font-
size: 20px; padding-top: 20px;"> Вистави:</p>
          <div class="events-display"> </div>
          <script src="js/moment.js"></script>
          <script src="js/jquery.eventCalendar.js"></script>
          <script>
            var array = <?php echo $data;?>;
            let eventArray=[];
            array.forEach(function(item, i, array)
            {
              eventArray.push(
                {start: item.f_t,
                end: item.f_t,
                name:item.name,
                image: item.image ,
                url: 'afisha/vistava/'+item.id,
                })
            })
          </script>
          new Calendar({

```



```

<section style="background: #f4f5f7 !important;" class="ftco-section">
  <div class="container">
    <div class="row justify-content-center mb-5">
      <div class="col-md-10 text-center heading-section ftco-animate">

        <h2 class="mb-4">Афіша</h2>
      </div>
    </div>

    <div class="row">

      <div class="carousel-testimony owl-carousel ">

        @foreach ($data as $el)
          @continue($el->id == null)
          <div class="col-md-4 ftco-animate" style="max-width: 100%;">

            <div class="text text-center" style="position: absolute;z-index: 1;bottom:
            0px;background: #83b582;opacity: 0.9;width: -webkit-fill-available;margin-right: 30px;padding:
            10px;">

              <h3><a href="{{route('vistava_afisha',$el->id)}}">
              style="color: white;text-transform: uppercase;">{{ $el->name}}</a></h3>

              <span style="color: white; position:left;text-transform:
              capitalize;">{{ $el->time}} &nbsp; &nbsp; &nbsp; &nbsp; </span>

              <span
              class="text text-left" style="color: white;">{{ $el->test_time}}</span>

            </div>
            <div class="case img d-flex align-items-center justify-content-center"
            style="background-image: url(<?=$el->image?>);">

          </div>
        </div>
      @endforeach ()

    </div>
  </div>
</section>

<section class="ftco-section bg-light">
  <div class="container">
    <div class="row justify-content-center mb-5 pb-3">
      <div class="col-md-7 heading-section text-center ftco-animate">
        <a href="noviny"> <h2>Новини</h2></a>
      </div>
    </div>
    <div class="row d-flex">
      @foreach ($data as $el)
        @continue($el->id_n == null)
        <div class="col-md-4 d-flex ftco-animate" >
          <div class="blog-entry justify-content-end">
            <a href="{{route('noviny_single',$el->id_n)}}" class="block-20" style="background-
            image: url(<?=$el->image_n?>); ">
          </a>
        </div>
      </div>
    </div>
  </div>
</section>

```

```

<div class="text p-4 float-right d-block">
  <div class="topper d-flex align-items-center">
    <div class="one py-2 pl-3 pr-1 align-self-stretch">
      <span class="day">{{ $el->dt_D }}</span>
    </div>
    <div class="two pl-0 pr-3 py-2 align-self-stretch">
      <span class="yr">{{ $el->dt_Y }}</span>
      <span class="mos">{{ $el->dt_M }}</span>
    </div>
  </div>
  <h3 class="heading mt-2"><a href="{{ route('noviny_single', $el->id_n) }}">{{ $el-
>name_n }}</a></h3>
  <p style="-ms-text-overflow: ellipsis;
text-overflow: ellipsis;
overflow: hidden;
-ms-line-clamp: 4;
-webkit-line-clamp: 4;
line-clamp: 4;
display: -webkit-box;
display: box;
word-wrap: break-word;
-webkit-box-orient: vertical;
box-orient: vertical;
font-size: 16px;
  line-height: 1.8;

"><?php
  echo $el->text;

?>
</p>
    </div>
  </div>
</div>
    @endforeach ()

  </div>
</div>
</section>

<footer class="ftco-footer ftco-bg-dark ftco-section">
  <div class="container">
    <div class="row mb-5">
      <div class="col-md">
        <div class="ftco-footer-widget mb-4">

          <ul class="ftco-footer-social list-unstyled float-md-left float-lft mt-5">
            <li class="ftco-animate"><a
href="https://www.facebook.com/groups/shchepkintheatre/"><span class="icon-
facebook"></span></a></li>
            <li class="ftco-animate"><a href="https://www.instagram.com/shchepkintheatre/"><span
class="icon-instagram"></span></a></li>
          </ul>
        </div>
      </div>

      <div class="col-md">
        <div class="ftco-footer-widget mb-4">
          <h2 class="ftco-heading-2">Адреса:</h2>
          <div class="block-23 mb-3">
            <ul>

```

```

        <li><span class="icon icon-map-marker"></span><span class="text">Україна, м.
Суми, пл. Театральна, 1.</span></li>
        <li><a href="#"><span class="icon icon-phone"></span><span class="text">+380
(542) 22-16-88</span></a></li>
        <li><span class="icon icon-envelope"></span><span
class="text">shepkindrama@gmail.com</span></li>
    </ul>
</div>
</div>
</div>
</div>
<div class="row">
    <div class="col-md-12 text-center">
        <p>Національний театр ім. М.С.Щепкіна © 2020. Усі права збережені.</p>
    </div>
</div>
</div>
</div>
</footer>

```

```

<div id="ftco-loader" class="show fullscreen"><svg class="circular" width="48px"
height="48px"><circle class="path-bg" cx="24" cy="24" r="22" fill="none" stroke-width="4"
stroke="#e6e6e6"/><circle class="path" cx="24" cy="24" r="22" fill="none" stroke-width="4" stroke-
miterlimit="10" stroke="#F96D00"/></svg></div>

```

```

<script src="js/jquery-migrate-3.0.1.min.js"></script>
<script src="js/popper.min.js"></script>
<script src="js/bootstrap.min.js"></script>
<script src="js/jquery.easing.1.3.js"></script>
<script src="js/jquery.waypoints.min.js"></script>
<script src="js/jquery.stellar.min.js"></script>
<script src="js/owl.carousel.min.js"></script>
<script src="js/jquery.magnific-popup.min.js"></script>
<script src="js/aos.js"></script>
<script src="js/jquery.animateNumber.min.js"></script>
<script src="js/scrollax.min.js"></script>
<script
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyBVWaKrjvy3MaE7SQ74_uJiULgl11JY0H2s&sensor=fals
e"></script>
    <script src="js/google-map.js"></script>
    <script src="js/main.js"></script>

</body>
</html>

```

ДОДАТОК Г

ЛІСТИНГ КОДУ SPECTACLE CONTROLLER

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\Spectacle;
use App\Models\Spectacle_list;
use App\Models\Noviny;
use App\Models\Actor;
use App\Models\Actor_list;
use \Carbon\Carbon;

class SpectacleController extends Controller
{
    public function afisha()
    {
        $spectacle= new Spectacle;
        $spectacle_list= new Spectacle_list;
        $now = Carbon::now();

        $test = $spectacle::select('*')
        ->join('spectacle_lists', 'spectacle_lists.id_vistavi', '=', 'spectacles.id')
        ->whereMonth('spectacle_lists.time', '=', $now->month)->where('spectacle_lists.time', '>', $now)
        ->orderBy('spectacle_lists.time', 'asc')->get();

        foreach ($test as $el){
            $dt = Carbon::parse($el->time)->locale('uk')->isoFormat('D MMMM');

            $dt2 = Carbon::parse($el->time)->locale('uk')->isoFormat('HH:mm');
            $el->time = $dt;
            $el->test_time = $dt2;
        }

        return view('afisha', ['data'=>$test]);
    }

    public function index()
    {
        $spectacle= new Spectacle;
        $noviny= new Noviny;
        $now = Carbon::now();

        $spectacle_list= $spectacle::select('*')->join('spectacle_lists', 'spectacle_lists.id_vistavi',
        '=', 'spectacles.id')->whereMonth('time', '=', $now->month)->where('time', '>', $now)-
        >orderBy('time', 'asc')->take(6)->get();

        foreach ($spectacle_list as $el){
            $dt = Carbon::parse($el->time)->locale('uk')->isoFormat('D MMMM');
            $dt2 = Carbon::parse($el->time)->locale('uk')->isoFormat('HH:mm');
            $dt3=Carbon::parse($el->time);
            $el->f_t= $dt3;
            $el->time = $dt;
        }
    }
}

```

```

        $el->test_time = $dt2;
    }

    $noviny_list= $noviny->orderBy('created_at','desc')->take(3)->get();
    foreach ($noviny_list as $el){
        $dt_Y = Carbon::parse($el->created_at)->locale('uk')->isoFormat('Y');
        $dt_D = Carbon::parse($el->created_at)->locale('uk')->isoFormat('D');
        $dt_M = Carbon::parse($el->created_at)->locale('uk')->isoFormat(' MMMM');
        $el->dt_Y = $dt_Y ;
        $el->dt_D = $dt_D ;
        $el->dt_M = $dt_M ;
    }

    $result = collect($spectacle_list)->merge($noviny_list);

return view('index',['data'=>$result]);
    }
    public function repertuar()
    {

        $spectacle= new Spectacle;
        $test= $spectacle->get()->unique('name');

return view('repertuar',['data'=>$test]);
    }
    public function kolektiv()
    {

        $actor= new actor;
        $test= $actor::select('*')
        ->leftjoin('actor_lists', 'actor_lists.actor_id', '=', 'actors.id')->get()->unique('full_name');;
return view('kolektiv',['data'=>$test]);
    }

    public function vistava_afisha($id)
    {
        $spectacle= new Spectacle;
        $spectacle_list= new Spectacle_list;

        $test= $spectacle_list::select('*')->join('spectacles', 'spectacle_lists.id_vistavi', '=',
'spectacles.id')->find($id);
        $dt = Carbon::parse($test->time)->locale('uk')->isoFormat('D MMMM HH:mm');

        $now=Carbon::now();

        if($test->time < $now) {

            $test->time = null;
        }
        $test->time = $test->time ? $dt : null;
return view('vistava_afisha',['data'=>$test]);

    }
    public function vistava_repertuar($id)
    {
        $spectacle= new Spectacle;
        $spectacle_list= new Spectacle_list;

        $test= $spectacle::select('*')->leftjoin('spectacle_lists', 'spectacle_lists.id_vistavi', '=',
'spectacles.id')->find($id);

```

```

$dt = Carbon::parse($test->time)->locale('uk')->isoFormat('D MMMM HH:mm');
$now=Carbon::now();

if($test->time < $now) {

    $test->time = null;
}
$test->time = $test->time ? $dt : null;

return view('vistava_repertuar',['data'=>$test]);
}
public function situativna_afisha($id)
{
    $spectacle= new Spectacle;
    $spectacle_list= new Spectacle_list;
    $now = Carbon::now();

    $test = $spectacle::select('*')
->join('spectacle_lists', 'spectacle_lists.id_vistavi', '=', 'spectacles.id')
->whereMonth('spectacle_lists.time', '=', $now->month)->where('spectacle_lists.time', '>', $now)-
>where('spectacles.company', '=', $id)
->orderBy('spectacle_lists.time', 'asc')->get();

    foreach ($test as $el){
        $dt = Carbon::parse($el->time)->locale('uk')->isoFormat('D MMMM');

        $dt2 = Carbon::parse($el->time)->locale('uk')->isoFormat('HH:mm');
        $el->time = $dt;
        $el->test_time = $dt2;
    }

return view('situativna_afisha',['data'=>$test]);
}

public function bileti($id)
{
    $spectacle= new Spectacle;

    $test= $spectacle::select('*')->leftJoin('spectacle_lists', 'spectacle_lists.id_vistavi', '=',
'spectacles.id')->find($id);

    $dt = Carbon::parse($test->time)->locale('uk')->isoFormat('D MMMM HH:mm');
    $test->time = $dt;

return view('bileti',['data'=>$test]);
}
public function noviny()
{
    $noviny= new Noviny;

    $result= $noviny->orderBy('created_at', 'desc')->get();
    foreach ($result as $el){
        $dt_Y = Carbon::parse($el->created_at)->locale('uk')->isoFormat('Y');
        $dt_D = Carbon::parse($el->created_at)->locale('uk')->isoFormat('D');
        $dt_M = Carbon::parse($el->created_at)->locale('uk')->isoFormat(' MMMM');
        $el->dt_Y = $dt_Y ;
        $el->dt_D = $dt_D ;
        $el->dt_M = $dt_M ;
    }
}

```



```

return view('noviny',['data'=>$result]);

}
public function actor($id)
{
    $spectacle= new spectacle;

    $test=
$spectacle::select('spectacles.name','actor_lists.spectacle_id','actors.full_name','actors.position'
,'actors.info','actors.foto')
->join('actor_lists', 'actor_lists.spectacle_id', '=', 'spectacles.id')
->join('actors', 'actor_lists.actor_id',
'=', 'actors.id')
->where('actor_lists.actor_id','=', $id)-
>get();

return view('actor',['data'=>$test]);
}
public function noviny_single($id_n)
{
    $noviny= new Noviny;

    $result= $noviny->find($id_n);

return view('noviny_single',['data'=>$result]);
}
}

```

ДОДАТОК Д

ЛІСТИНГ КОДУ KABINET CONTROLLER

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\Spectacle;
use App\Models\Spectacle_list;
use App\Models\Ticket;
use App\Models\Sector;
use App\Models\User;
use App\Models\Place;
use App\Models\Zal;
use \Carbon\Carbon;
use PDF;

class KabinetController extends Controller
{
    public function index($viewer_id)
    {
        $spectacle= new Spectacle;
        $spectacle_list= new Spectacle_list;
        $now = Carbon::now();

        $test = $spectacle::select('spectacles.name', 'spectacles.image', 'spectacles.id as
spectacle_id', 'spectacle_lists.time', 'tickets.created_at', 'places.row', 'places.place', 'sectors.price
', 'sectors.name as sector_name', 'zals.name as zal_name', 'tickets.viewer_id', 'tickets.id as
ticket_id')

'spectacle_lists.id_vistavi', '=', 'spectacles.id')
->join('spectacle_lists',
'=', 'tickets.spectacle_list_id')
->join('tickets', 'spectacle_lists.id',
'tickets.place_id')
->join('places', 'places.id', '=',
'=', 'sectors.id')
->join('sectors', 'places.sector_id',
'sectors.zal_id')
->join('zals', 'zals.id', '=',
'tickets.viewer_id')
->join('users', 'users.id', '=',
->where('tickets.viewer_id', '=', $viewer_id)
->orderBy('spectacle_lists.time', 'asc')->get();

        foreach ($test as $el){
            $dt = Carbon::parse($el->time)->locale('uk')->isoFormat('D MMMM');

            $dt2 = Carbon::parse($el->time)->locale('uk')->isoFormat('HH:mm');
            $el->time = $dt;
            $el->test_time = $dt2;
        }

        return view('kabinet.index', ['data'=>$test]);
    }

    public function ticket($ticket_id)
    {
        $spectacle= new Spectacle;
    }
}

```

```

        $spectacle_list= new Spectacle_list;
    $now = Carbon::now();

    $test = $spectacle::select('spectacles.name','spectacles.image','spectacles.id as
spectacle_id','spectacle_lists.time','tickets.created_at','places.row','places.place','sectors.price
','sectors.name as sector_name','zals.name as zal_name','tickets.viewer_id','tickets.id as
ticket_id')
                                                    ->join('spectacle_lists',
'spectacle_lists.id_vistavi', '=', 'spectacles.id')
                                                    ->join('tickets', 'spectacle_lists.id',
'=', 'tickets.spectacle_list_id')
                                                    ->join('places', 'places.id', '=',
'tickets.place_id')
                                                    ->join('sectors', 'places.sector_id',
'=', 'sectors.id')
                                                    ->join('zals', 'zals.id', '=',
'sectors.zal_id')
                                                    ->join('users', 'users.id', '=',
'tickets.viewer_id')
        ->where('tickets.id', '=', $ticket_id)
        ->orderBy('spectacle_lists.time','asc')->get();

    foreach ($test as $el){
    $dt = Carbon::parse($el->time)->locale('uk')->isoFormat('D MMMM');

        $dt2 = Carbon::parse($el->time)->locale('uk')->isoFormat('HH:mm');
    $el->time = $dt;
    $el->test_time = $dt2;
    }

        $user=auth()->user()->id;
        return view('kabinet.ticket',['data'=>$test,'user'=>$user]);

    }

    public function bilet($ticket_id)
    {
        $spectacle= new Spectacle;
        $spectacle_list= new Spectacle_list;
    $now = Carbon::now();

    $test = $spectacle::select('spectacles.name','spectacles.image','spectacles.id as spectacle_id',
'spectacle_lists.time','tickets.created_at','places.row','places.place','sectors.price',
'sectors.name as sector_name','zals.name as zal_name','tickets.viewer_id','tickets.id as
ticket_id')
                                                    ->join('spectacle_lists', 'spectacle_lists.id_vistavi',
'=', 'spectacles.id')
                                                    ->join('tickets', 'spectacle_lists.id', '=',
'tickets.spectacle_list_id')
                                                    ->join('places', 'places.id', '=', 'tickets.place_id')
                                                    ->join('sectors', 'places.sector_id', '=',
'sectors.id')
                                                    ->join('zals', 'zals.id', '=', 'sectors.zal_id')
                                                    ->join('users', 'users.id', '=', 'tickets.viewer_id')
        ->where('tickets.id', '=', $ticket_id)
        ->orderBy('spectacle_lists.time','asc')->get();

    foreach ($test as $el){
    $dt = Carbon::parse($el->time)->locale('uk')->isoFormat('D MMMM');

        $dt2 = Carbon::parse($el->time)->locale('uk')->isoFormat('HH:mm');
    $el->time = $dt;
    $el->test_time = $dt2;
    }

        $user=auth()->user()->id;

```

```
        view()->share(['data'=>$test,'user'=>$user]);
$pdf = PDF::loadView('kabinet.bilet')->setOptions(['defaultFont' => 'sans-serif']);
$pdf->getDomPDF()->setHttpContext(
    stream_context_create([
        'ssl' => [
            'allow_self_signed'=> TRUE,
            'verify_peer' => FALSE,
            'verify_peer_name' => FALSE,
        ]
    ])
);
return $pdf->download('kabinet.bilet.pdf');
}

}
```

ДОДАТОК Е

ОПРИЛЮДНЕННЯ РЕЗУЛЬТАТІВ РОБОТИ

ІМА :: 2021

СЕКЦІЯ 2: Інформаційні технології проектування

Web-додаток «Національного театру ім. М. С. Щепкіна»

Гаврилів А.О., студент; Нагорний В.В., старший викладач
Сумський державний університет, м. Суми, Україна

Актуальність. Впровадження інформаційних технологій в усі сфери діяльності обумовлено стрімким розвитком науки, якісним стрибком можливостей людини, постійно зростаючими обсягами інформації, складністю процесів, що відбуваються і явищ. Мистецтво не є виключенням. Сьогодні такі процеси як купівля квитків, перегляд актуальної афіші та інформації про виставу можна зробити онлайн. Створюваний Web-додаток буде корисний як для глядачів, так і для персоналу театру, для полегшення процесу адміністрування даними.

Постановка задачі. Розробити Web-додаток театру «Національний театр ім. М. С. Щепкіна». Призначенням якого є демонстрація основної інформації про заклад та автоматизація процесів збуту квитків.

Результати. Існуючий Web-додаток театру має застарілий дизайн, що негативно впливає на його репутацію серед глядачів. Досить суттєвою проблемою є стереотипність, в більшості випадків серед молоді. Вони вважають що театр – це нецікаво. Більшість таких думок обґрунтовані шкільними походами до театру на вистави «за програмою». Люди навіть не здогадуються про існування 3D-ефектів та сучасних технологій в театрі, не кажучи вже про великий вибір різноманітних вистав. На основі цього було прийнято рішення про створення нового Web-додатку, перевагами якого є:

- можливість купівлі та бронювання квитків;
- зручний, сучасний інтерфейс;
- можливість підбору вистави з ситуативними потребами;
- можливість перегляду історії відвіданих вистав;
- можливість легкого редагування інформації на сайті.

Також він матиме таку функціональність, як перегляд афіші, інформації про виставу та митців театру. Пошук інформації про театр та його новини.

Висновки. Завдяки використанню нового Web-додатку драматичного театру користувачеві надаються можливості та переваги, які неможливі при використанні існуючої системи. Він дозволить автоматизувати процес купівлі квитків та полегшить процес адміністрування даними.