**ЗМІСТ**

|  |  |
| --- | --- |
| 1. ТЕХНІЧНЕ ЗАВДАННЯ | 4 |
| 1.1 Підстави для розробки | 4 |
| 1.2 Призначення розробки | 4 |
| 1.3 Аналіз аналогічних систем | 6 |
| 1.4 Опис розширення  1.5 Особливості та технічна інформація | 12 |
| 2. АРХІТЕКТУРА ІНФОРМАЦІЙНОЇ СИСТЕМИ | 13 |
| 2.1 Проектування рівню «Use Case View» та його документування | 13 |
| 2.2 Проектування рівню «Logical View», моделювання поведінки та взаємодії його об’єктів | 20 |
| 2.3 Розробка прототипів інформаційної системи | 22 |
| 2.4 Програмні засоби | 29 |
| ВИСНОВКИ | 33 |
| Додаток. Код програми | 34 |

## 1. ТЕХНІЧНЕ ЗАВДАННЯ

## 1.1 Підстави для розробки

Розробка виконується на підставі рішення засідання циклової комісії про затвердження тем дипломних та курсових робіт (протокол № \_\_\_ від «\_\_\_» \_\_\_\_\_\_\_\_\_ 20\_\_ року).

*Повна назва розробки*: «Веб-розширення (веб-утиліта) для створення QR кодів».

*Коротка назва*: Fast Link.

## 1.2 Призначення розробки

*Мета роботи* полягає у формуванні системи теоретичних і практичних знань у галузі моделювання, проектування та аналізу інформаційних систем, набутті практичних навичок імітаційного моделювання складних систем, застосування мови моделювання UML та відповідного об’єктно-орієнтованого підходу до моделювання та проектування автоматизованих інформаційних систем з використанням автоматизованих методів.

Виконавець курсового проекту повинен аналізувати предметну область і описувати її з використанням мови UML; виконувати об’єктно-орієнтовану декомпозицію та об’єктно-орієнтований аналіз предметної області, використовуючи ієрархію класів, діаграми процесів та об’єктів; застосовувати різні інструментальні засоби об’єктно-орієнтованого аналізу і графічного представлення на мові UML; розробляти проект програмної системи на базі технології об’єктно-орієнтованого програмування, використовуючи відповідні діаграми мови UML; оформляти програмну документацію; проектувати архітектуру АІС з використанням засобів візуального моделювання; проектувати компоненти архітектурного рішення.

Метою даного проекту є розробка сайту Розкладу радіостанції та подальшого його розміщення в мережі Інтернет. Призначення системи – розклад радіостанції, продаж фірмових товарів та замовлення пісень.

**1.3 Аналіз аналогічних систем**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **№** | **Назва системи** | **Розробник** | **Архітектура** | **Мова реалізації** | **Функціонал** | **Посилання** |
| 1. | Люкс ФМ | luxnet | Web-application | HTML, JavaScript, PHP | * Слухати люкс фм * Перехід на головну за логотипом * Музика * Lux life * Showbiz * Розваги * Акції * Мода і краса * Вибір мови * Lifestyle * Веб камера * Кнопка «далі в ефірі» * Ефір з… * Хороші новини * вибір новин * більше новин * активні банери * кращі матеріали * Музика * ONAIR * ХІТ-ПАРАД * ВСЯ МУЗИКА * ВИКОНАВЦІ * ЗВУЧАЛО В ЕФІРІ * Lux life * Веб-камера * Ведучі * Гості * Вечірки * Програми * Розклад програм * Архів * Showbiz * Ексклюзив * Інтерв'ю * Кіно * Чутки * Кліпи * Селебріті * Монархи * Події * Розваги * Приколи * Тести * Тваринки * Про все * Написати відгук * Соц. Мережі * Карта сайту * App store * Googleplay * Пошук * Акції * Lifestyle * Секс * Подорожі * Food * Декор/дизайн * Вчені * Life * Мода і краса * Краса/Здоров'я * Спорт * Стиль * Тренди * Карта покриття * Контакти * Львів онлайн * Правила користування сайтом * Політика конфіденційності * Правила видачі призів * Структура власності * Редакційний статут * Вибір сторінок * Карта сайту * Активні банери | https://lux.fm/ |
| 2. | Закарпаття ФМ онлайн | Final Level | Web-application | HTML, JavaScript, PHP | * Всі станції * Обране * Жанри * Музика * Вибір країни * Увійти * Оцінити * Соц. Мережі * Слухати * Веб сайт * Контакти * Фейсбук * Плей лист * Встановити додаток * Зараз в ефірі * Плей лист Закарпаття фм * Вибір пісень * Залишити відгук * Вибір сторінки * Рекомендовані * Вибір мови * Вибір країни * Популярні країни * Угода користувача * Політика конфіденційності * Зворотній зв’язок * Віджети * Радіостанціям * Рівень гучності * Плеєр * Поділитися * Вибір станції * Вибір міста | https://onlineradiobox.com  /ua/zakarpattya/?lang=ru |
| 3. | Стильное радио ПЕРЕЦ FM | Yellow  Web&  Marketing | Web-application | HTML, JavaScript, PHP | * Перехідна головну за логотипом * Соцмережі * Слухати в ефірі онлайн * Пошук * Наша команда * Мапа мовлення * Проекти * Контакти * Гість в ефірі * Студія перець фм * Завантажити мобільний додаток * Акції на перець фм * Перехід за логотипом «древній Київ» * Активні банери * Передати привіт в ефірі * Мапа сайту * Інф. Про структуру власності * Активна частина сторінки в фейсбук * Концерти * Показати ще * Детальніше * Головна * Статистика | http://perec.fm/ |

## 1.4 Опис розширення

Розширення браузера (англ. Browser extension) — комп'ютерна програма, яка певним чином розширює функціональні можливості веб-браузера, підвищує його безпеку та продуктивність. В залежності від браузера і версії, цей термін може відрізнятися від подібних термінів, таких як плагін (plug-in) або доповнення (add-on). На 2021 рік плагіни застаріли для більшості браузерів, тоді як розширення широко використовуються. Найпопулярніший браузер Google Chrome має більш ніж 100 000 розширень , доступних , але більше не може не підтримувати плагінів.

Переваги розширень

* Розширення можна використовувати, щоб додавати нові функції у наявні веб-додатки.
* Ви отримуєте доречні посилання та відомості про сторінки.
* Ви отримуєте своєчасні сповіщення про певні події.
* Розширення можна використовувати як ярлики, коли вони з’являються в адресному рядку.

Синтаксис для розширень може досить відрізнятися для різних браузерів, принаймні, досить різні, щоб розширення, яке працює на одному браузері, не працювало на іншому. Що стосується інструментів інформаційно-пошукової системи, спробою обійти цю проблему була багатоміткова стратегія, запропонована в проекті Mycroft, базу даних інформаційно-пошукової системи доповнень, який працює на різних браузерах.

Для розробки використовується звичайний HTML + CSS + JavaScript, плюс API браузера, наприклад для впровадження коду розширення в кожну сторінку користувача. З 2019 року можна використовувати один і той же код для Google Chrome, Opera, Firefox, Edge, тоді як раніше наприклад розширення для Firefox не працював для Google Chrome.

QR-код (англ. quick response — швидкий відгук) — матричний код (двовимірний штрих-код). Основна перевага QR-коду — це легке розпізнавання сканувальним обладнанням (в тому числі й фотокамерою мобільного телефона), що дає можливість використання в торгівлі, на виробництві, в логістиці.

**1.5 Особливості та технічна інформація**

На відміну від старого штрих-коду, який сканують тонким променем, QR-код визначається сенсором як двовимірне зображення. Три квадрати в кутах зображення та менші синхронізувальні квадратики по всьому коду дозволяють нормалізувати розмір зображення і його орієнтацію, а також кут, під яким сенсор розташований до поверхні зображення. Точки переводяться в двійкові числа з перевіркою контрольних сум.

Основна перевага QR-коду — це легке розпізнавання сканувальним обладнанням (в тому числі і фотокамерою мобільного телефону).

Хоча термін «QR code» є зареєстрованим товарним знаком «DENSO Corporation», використання кодів не обкладається ніякими ліцензійними відрахуваннями, а самі вони описані й опубліковані як стандарти ISO. Специфікація QR-коду не описує формат даних.

Найпопулярніші програми перегляду QR-кодів підтримують такі формати даних: URL, веб-сторінки, E-mail (з темою листа), SMS на номер (з темою), MeCard, vCard, географічні координати.

Також деякі програми можуть розпізнавати файли GIF, JPG, PNG або MID менше 4 КБ і зашифрований текст, але ці формати не отримали популярності.

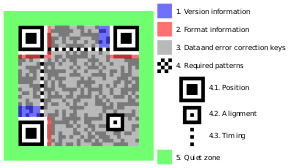


Рисунок 1 Структура QR-коду з позначеними функціональними елементами.

Найменший QR-код (версія 1) має розмір 21 × 21 піксель (без урахування полів), найбільший (версія 40) — 177 × 177 пікселів.

Максимальна кількість символів, які вміщаються в один QR-код:

* Цифри — 7089
* Цифри і букви (включаючи кирилицю) — 4296
* Двійковий код — 2953 байт
* Ієрогліфи — 1817

Також існують «псевдо кодування»: завдання методу кодування в даних, розбиття довгого повідомлення на кілька кодів тощо.

Для виправлення помилок застосовується код Ріда — Соломона з 8-бітним кодовим словом. Є чотири рівня надмірності: 7, 15, 25 і 30 %. Завдяки виправленню помилок вдається нанести на QR-код малюнок і все одно залишити його читабельним.

Щоб у коді не було елементів, здатних заплутати сканер, область даних XOR'иться зі спеціальною маскою. Коректно працюючий кодер повинен перебрати всі варіанти масок, порахувати штрафні очки для кожної області та вибрати найбільш вдалу.

**2. АРХІТЕКТУРА ІНФОРМАЦІЙНОЇ СИСТЕМИ**

**2.1 Проектування рівню «Use Case View» та його документування**

Даний рівень описує, що система повинна робити, які дії в ній виконують користувачі та яка реакція системи на дію користувача.

Для інформаційної системи «Розклад радіостанції» ми визначили акторів *Користувач, Оператор, Склад та Адміністратор* та їхні основні дії (прецеденти) з опису роботи системи.

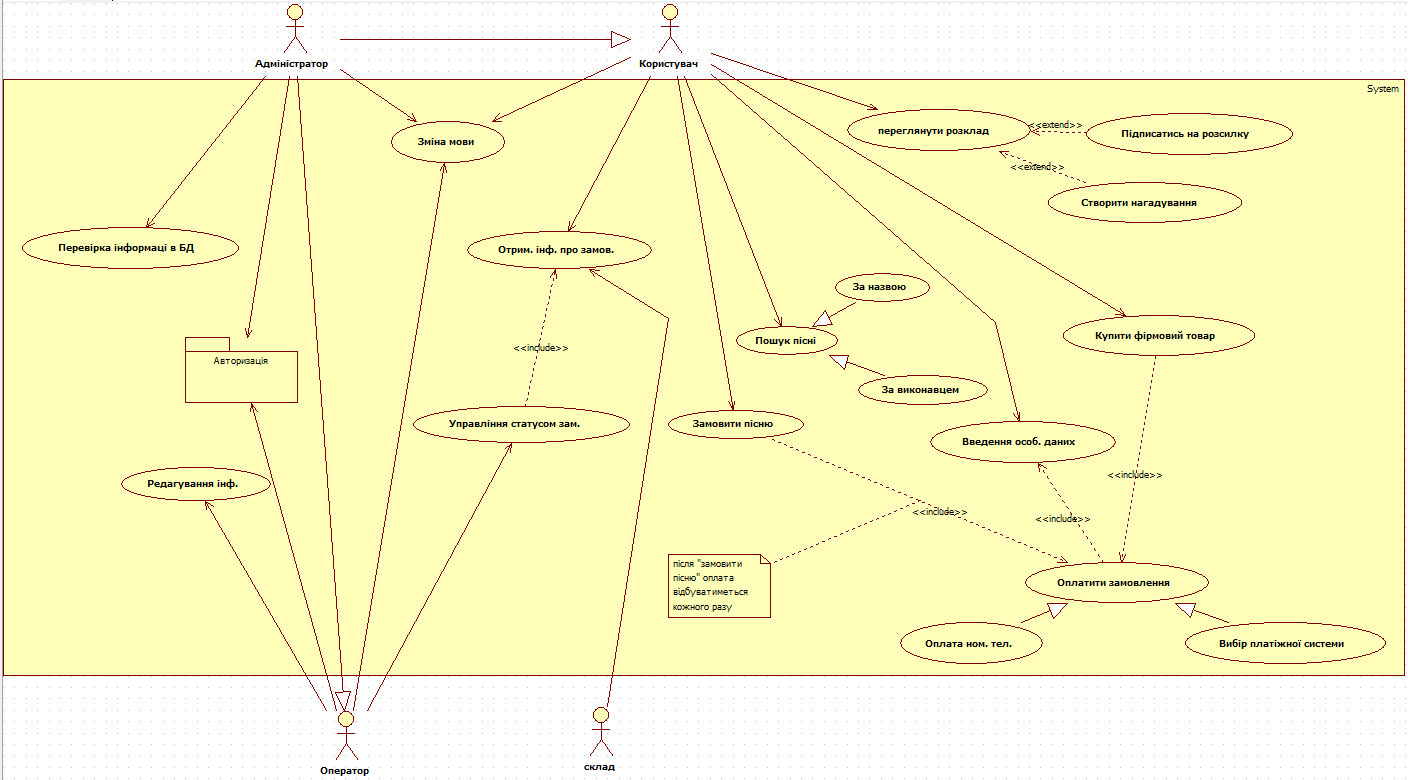


Рисунок 1 – Діаграма варіантів використання системи

1. Користувачі системи:

*Користувач* – людина, яка може переглядати розклад, шукати пісні і здійснювати замовлення товарів.

*Адміністратор* – людина, яка перевіряє актуальність даних системи.

*Оператор* – людина, яка наповнює, та модифікує дані системи.

*Склад* – людина яка надає інформацію про доставку.

1. Дії системи:

|  |  |
| --- | --- |
| ***Переглянути розклад*** | Прецедент запускається користувачем для того, щоби переглянути розклад, за яким працює радіостанція. |
| *Primary actor* | Користувач. |
| *Stakeholders and interests* | Користувач – зацікавлений у перегляді розкладу, що пропонуються радіостанцією. |
| *Pre-conditions* | Присутнє з’єднання з мережею Інтернет. Клієнт виконав вхід на сайт. |
| *Main Success Scenario:* | 1. Користувач обирає режим показу розкладу . 2. Система відкриває розклад.   *А1. Користувач обирає «Створити нагадування»*  *А2. Користувач обирає «Підписатись на розсилку»* |
| *Extensions:* | **А1. Користувач обирає «Створити нагадування»**   1. Система видає повідомлення «Нагадування   створено».  **А2. Користувач обирає «Підписатись на розсилку»**   1. Система видає повідомлення «Ви успішно підписались». |

|  |  |
| --- | --- |
| ***Пошук пісні*** | Прецедент запускається користувачем для того, щоби знайти пісню. |
| *Primary actor* | Користувач. |
| *Stakeholders and interests* | Користувач – зацікавлений у перегляді розкладу, що пропонуються радіостанцією. |
| *Pre-conditions* | Присутнє з’єднання з мережею Інтернет. Користувач виконав вхід на сайт. |
| *Main Success Scenario:* | 1. Користувач обирає режим пошуку пісні. 2. Система відкриває форму введення даних. 3. Користувач обирає фільтр пошуку «За назвою»   *А1. Користувач обирає фільтр пошуку «За автором»*   1. Користувач вводить дані для пошуку. 2. Система обробляє запит та виводить   результат пошуку.  *А2. Результат не знайдено.* |
| *Extensions:* | **А1. Користувач обирає фільтр пошуку «За автором»**   1. Користувач вводить дані для пошуку. 2. Система обробляє запит та виводить   результат пошуку.  **А2. Результат не знайдено**   1. Система видає повідомлення «Таку пісню не знайдено». |

|  |  |
| --- | --- |
| ***Замовити пісню*** | Прецедент запускається користувачем для того, щоб замовити пісню. |
| *Primary actor* | Користувач. |
| *Stakeholders and interests* | Користувач – зацікавлений у замовленні пісні, що пропонуються радіостанцією. |
| *Pre-conditions* | Присутнє з’єднання з мережею Інтернет. Користувач виконав вхід на сайт. |
| *Main Success Scenario:* | 1. Користувач обирає режим замовлення пісні. 2. Система пропонує клієнту обрати спосіб   оплати.   1. Користувач обирає механізм оплати та   вводить всі необхідні дані оплати.   1. Система очікує перевірки   платоспроможності клієнта.  *А2. Користувач не пройшов механізм платоспроможності.*   1. Система видає повідомлення «Замовлення   пройшло успішно» та повертається на  головну сторінку. |
| *Extensions:* | **А1. Користувач не пройшов механізм платоспроможності**   1. Система надіслала банку запит на   платоспроможність клієнта.   1. Система видає повідомлення «Користувач   не пройшов перевірку платоспроможності».  Система повертається на головну сторінку. |

|  |  |
| --- | --- |
| ***Купити фірмовий товар*** | Прецедент запускається користувачем для того, щоб замовити фірмовий товар. |
| *Primary actor* | Користувач. |
| *Stakeholders and interests* | Користувач – зацікавлений у замовленні фірмового товару, що пропонуються радіостанцією. |
| *Pre-conditions* | Присутнє з’єднання з мережею Інтернет. Користувач виконав вхід на сайт. |
| *Main Success Scenario:* | 1. Користувач обирає режим покупки фірмового товару. 2. Система пропонує користувачу обрати   спосіб оплати.   1. Користувач обирає механізм оплати та   вводить всі необхідні дані оплати.   1. Система очікує перевірки   платоспроможності користувача.  *А2. Користувач не пройшов механізм платоспроможності.*   1. Система видає повідомлення «Купівля пройшла успішно» та повертається на головну сторінку. |
| *Extensions:* | **А1. Користувач не пройшов механізм платоспроможності**   1. Система надіслала банку запит на   платоспроможність клієнта.   1. Система видає повідомлення «Користувач   не пройшов перевірку платоспроможності».   1. Система повертається на головну сторінку. |

|  |  |
| --- | --- |
| ***Зміна мови*** | Прецедент запускається користувачем (мається на увазі як і звичайний користувач так і адміністратор та оператор сайту) для того, щоби змінити мову відображення сайту. |
| *Primary actor* | Користувач. |
| *Stakeholders and interests* | Користувач – зацікавлений у користуванні сайту зручною для нього мовою. |
| *Pre-conditions* | Присутнє з’єднання з мережею Інтернет. Клієнт виконав вхід на сайт. |
| *Main Success Scenario:* | 1. Користувач обирає режим зміни мови. 2. Система змінює мову відображення сайту. |

|  |  |
| --- | --- |
| ***Редагування інформації*** | Прецедент запускається оператором для того, щоб редагувати інформацію про систему. |
| *Primary actor* | Оператор. |
| *Stakeholders and interests* | Оператор – зацікавлений, щоби дії проходили успішно. |
| *Pre-conditions* | Присутнє з’єднання з мережею Інтернет. Оператор здійснив вхід у систему. |
| *Main Success Scenario:* | 1. Оператор обирає режим редагування   інформації.   1. Система відкриває сторінку оператора. 2. Система відкриває форму редагування даних. 3. Оператор вибирає дані, які необхідно відредагувати. 4. Система відкриває детальну інформацію. 5. Оператор модифікує дані та підтверджує модифікацію даних. 6. Система зберігає відредаговані дані та оновлює front-end (інтерфейс). 7. Система повертається до кроку 2 основного потоку. |

|  |  |
| --- | --- |
| ***Перевірка інформації в БД*** | Прецедент запускається адміністратором для того, щоб перевірять правильність інформації в системі. |
| *Primary actor* | Адміністратор. |
| *Stakeholders and interests* | Адміністратор – зацікавлений в правильності інформації яка використовується в системі. |
| *Pre-conditions* | Присутнє з’єднання з мережею Інтернет. Адміністратор здійснив вхід у систему. |
| *Main Success Scenario:* | 1. Адміністратор обирає режим перегляду інформації БД. 2. Система відкриває сторінку адміністратора. 3. Система відкриває БД та виводить її вміст. 4. Адміністратор вибирає дані, які необхідно перевірити. 5. Система відкриває детальну інформацію.   *А1. Адміністратор впевнений в правильності інформації.*   1. Адміністратор модифікує дані та підтверджує модифікацію даних. 2. Система зберігає відредаговані дані та оновлює front-end (інтерфейс) якщо потрібно. 3. Система повертається до кроку 2 основного потоку. |
| *Extensions:* | **А1. Адміністратор впевнений в правильності інформації.**   1. Адміністратор вибирає перехід на сторінку   адміністратора.  2) Система відкриває сторінку адміністратора. |

**2.2 Проектування рівню «Logical View», моделювання поведінки та взаємодії його об’єктів**

**2.2.1** Даний рівень описує, як система повинна бути побудована.

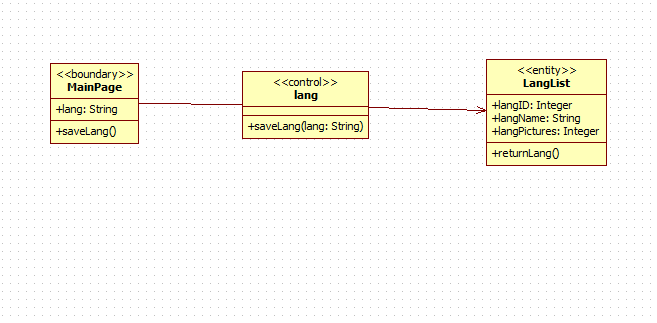


Рисунок 1 – Діаграма класів дії «Зміна мови»

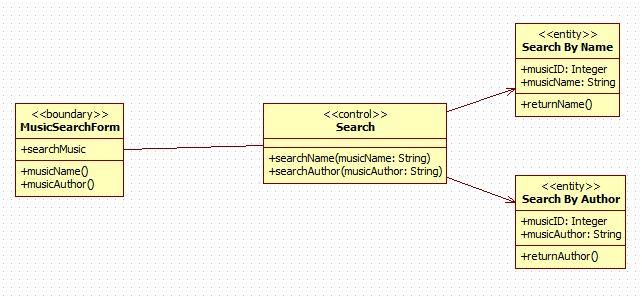


Рисунок 2 – Діаграма класів дії «Пошук пісні»

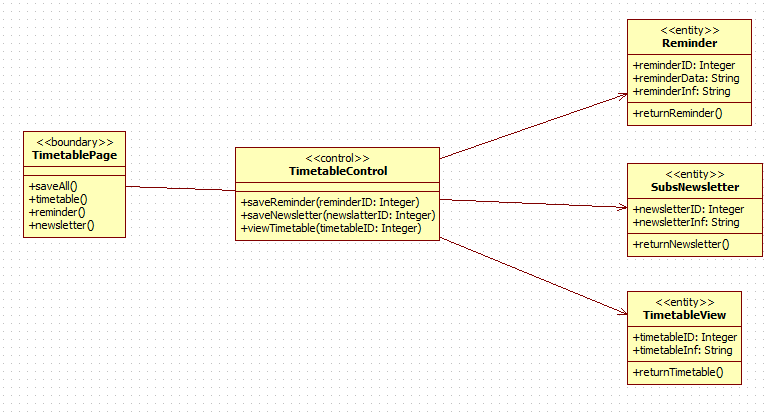


Рисунок 3 – Діаграма класів дії «Переглянути розклад»

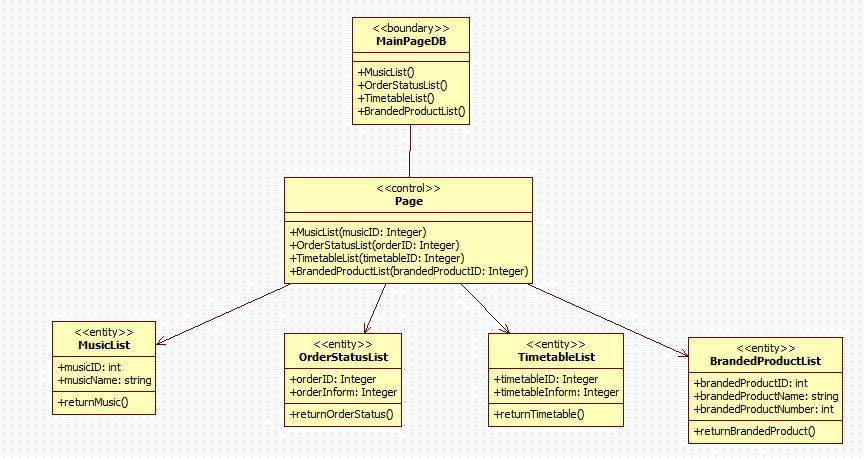


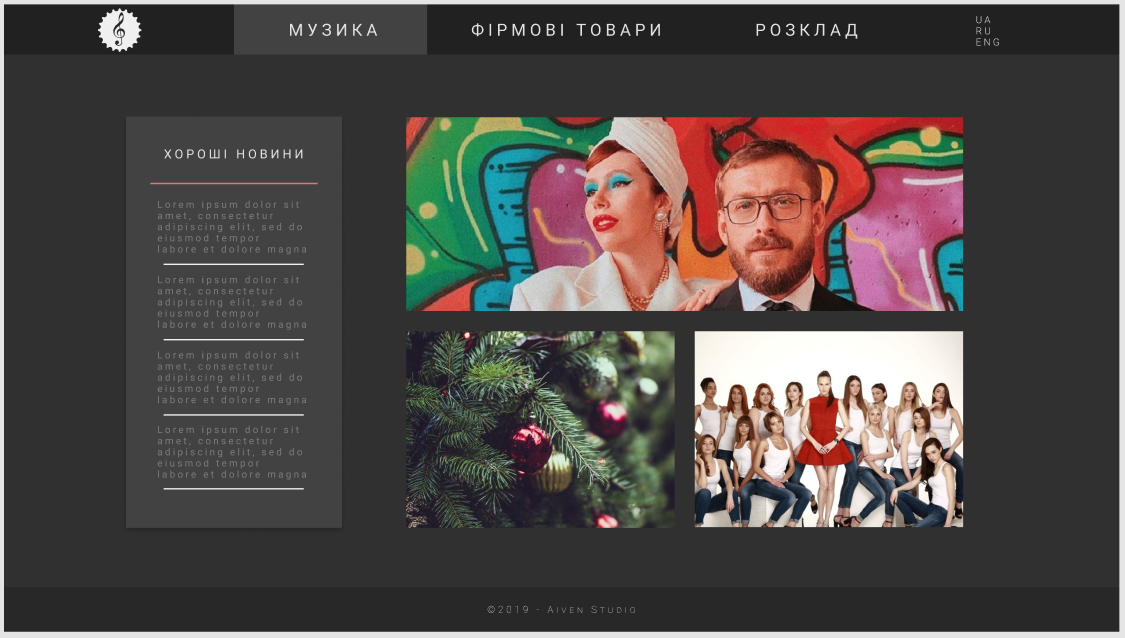
Рисунок 4 – Діаграма класів дії «Перевірка інформації в БД»

**2.3 Розробка прототипів системи**

Прототип головної сторінки



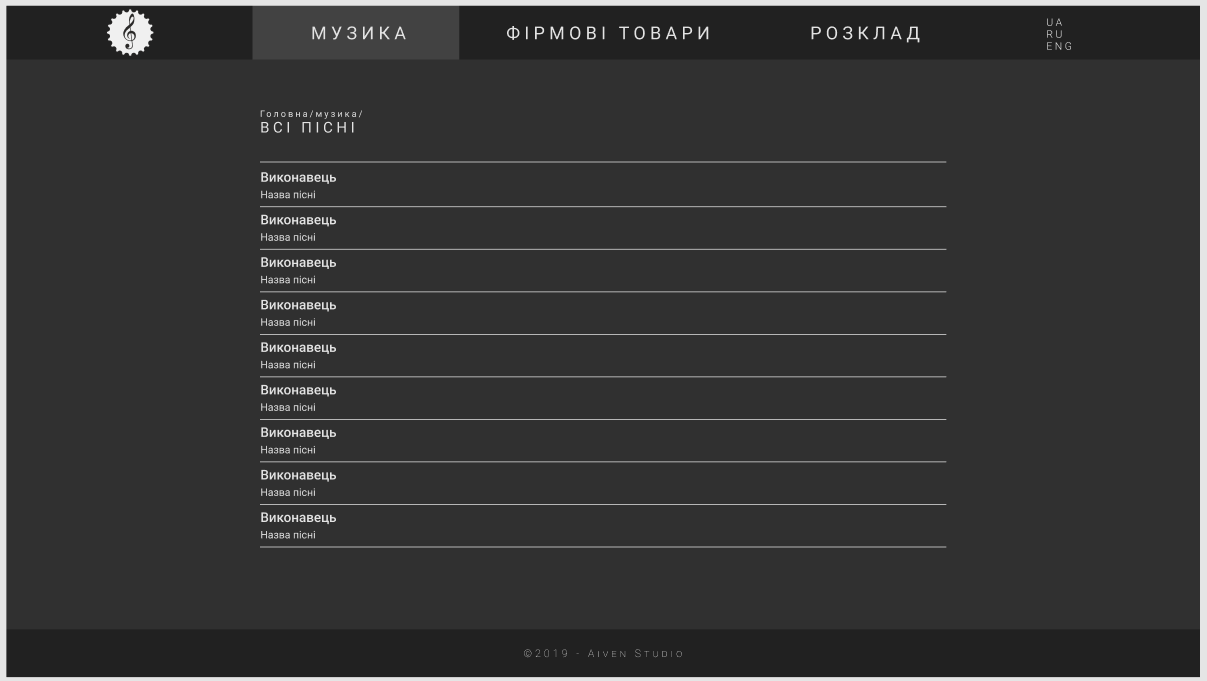
Прорисовка головної сторінки



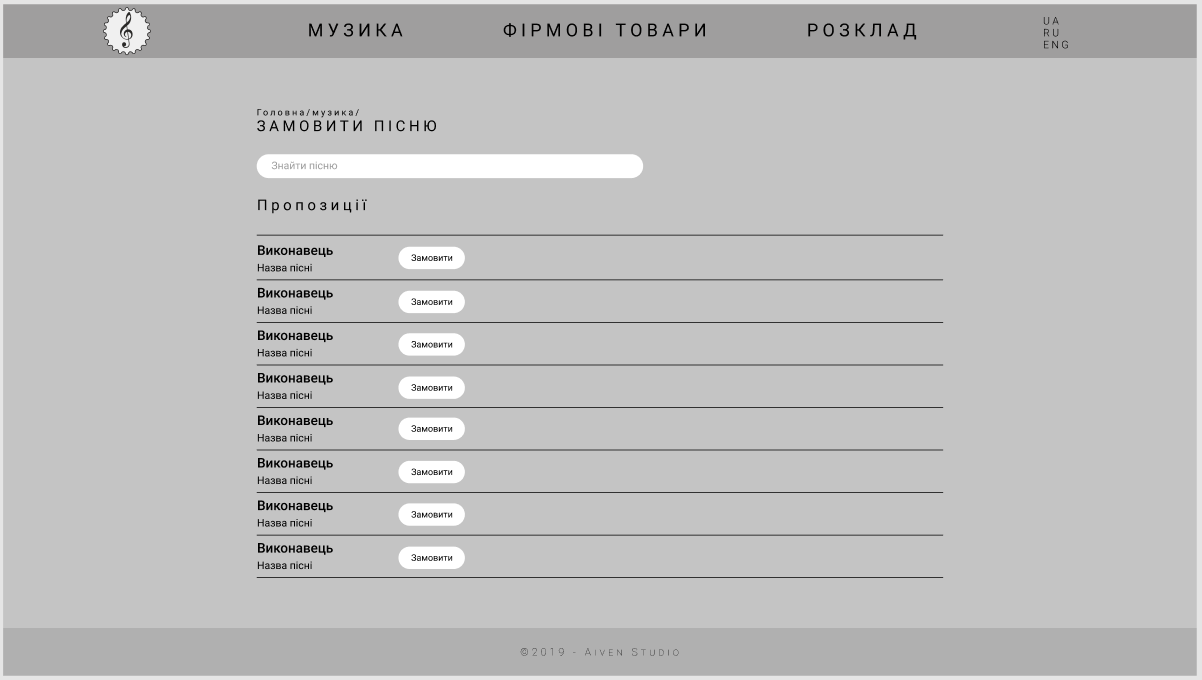
Прототип сторінки «Вся музика»



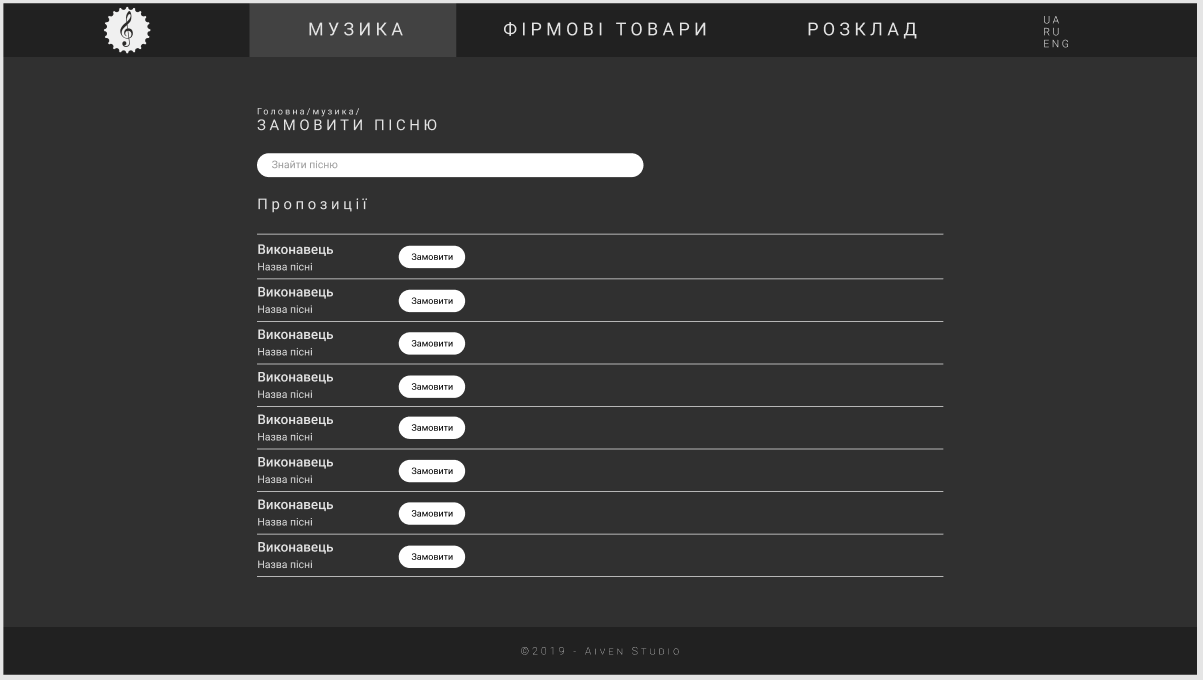
Прорисовка сторінки «Вся музика»



Прототип сторінки «Замовити пісню»



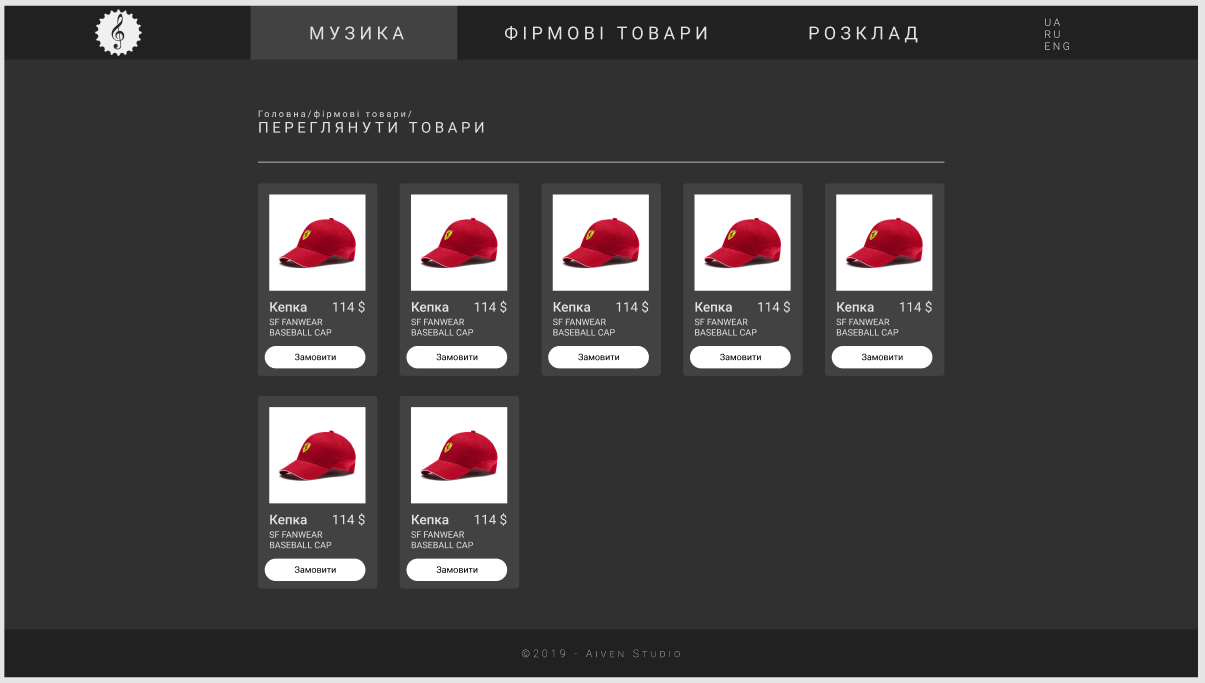
Прорисовка сторінки «Замовити пісню»



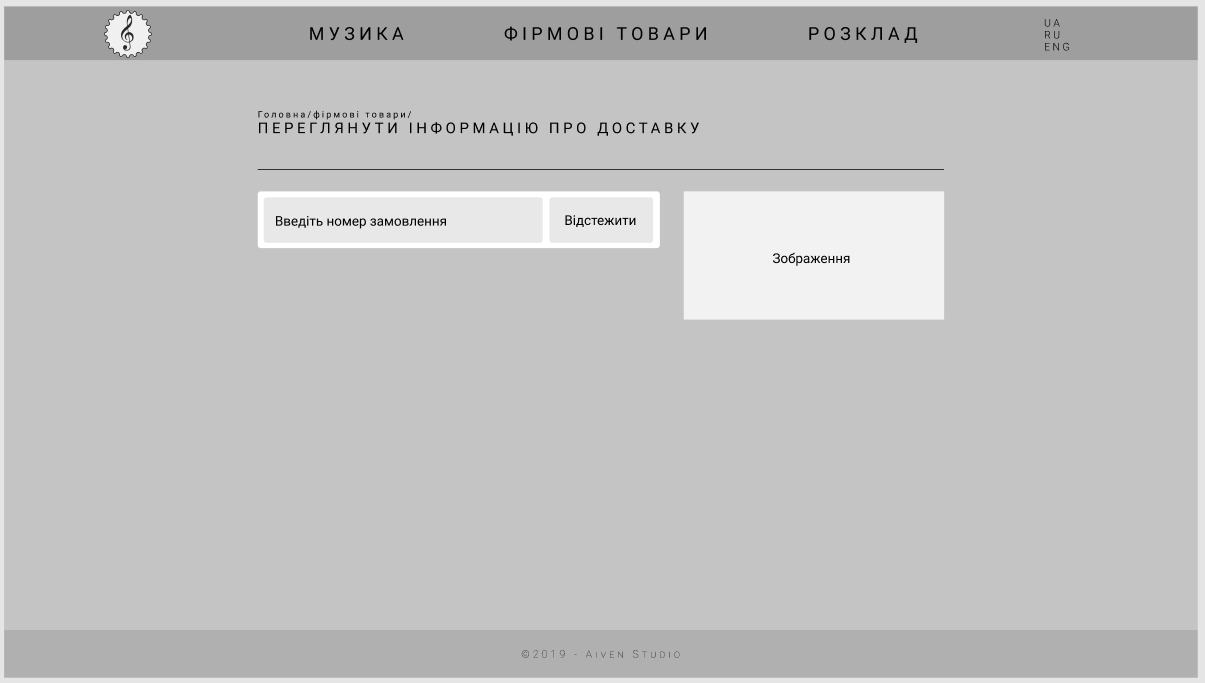
Прототип сторінки «Переглянути товари»



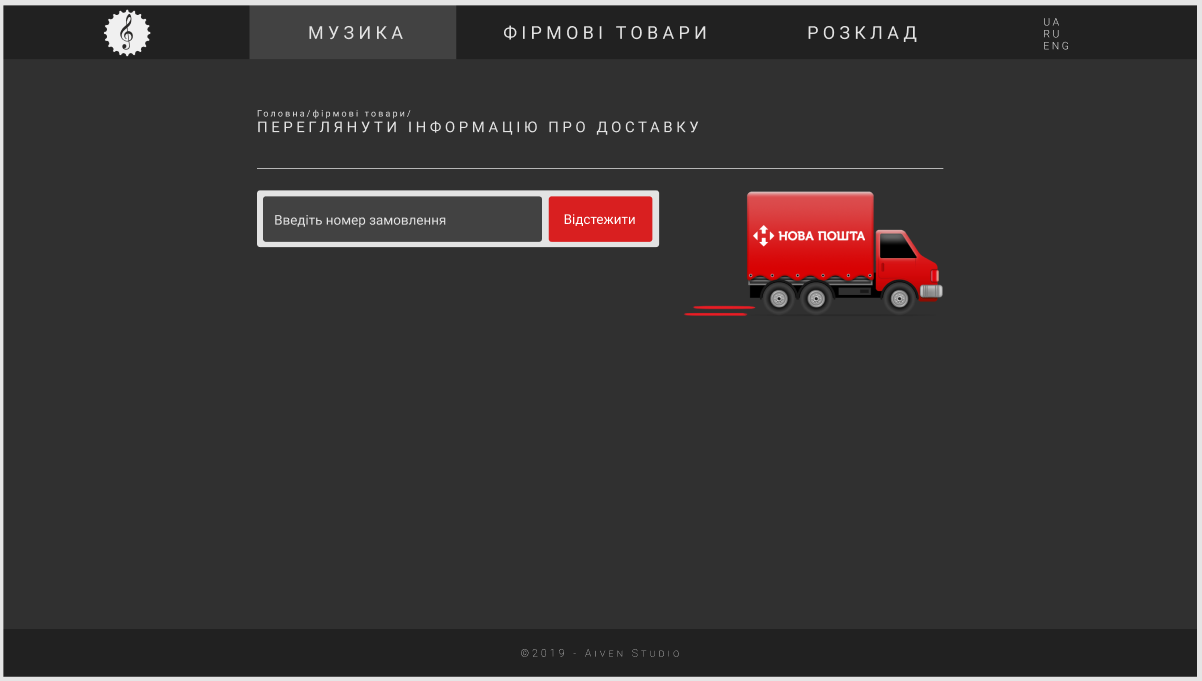
Прорисовка сторінки «Переглянути товари»



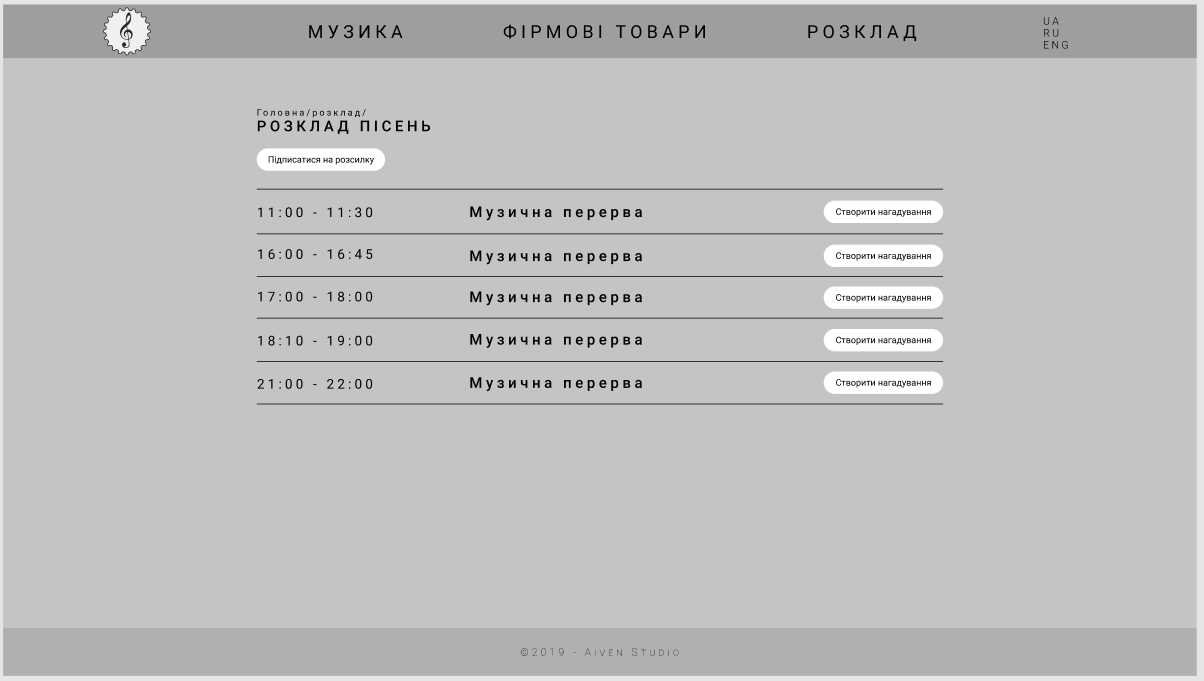
Прототип сторінки «Інформація про доставку»



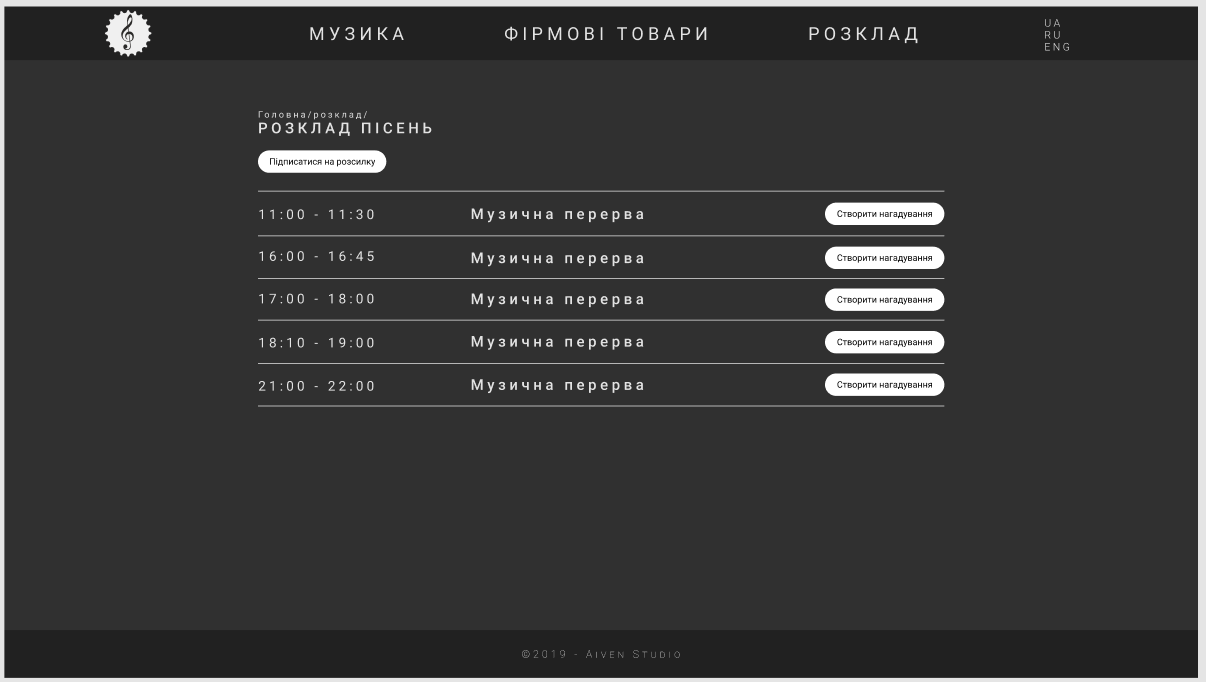
Прорисовка сторінки «Інформація про доставку»



Прототип сторінки «Розклад пісень»



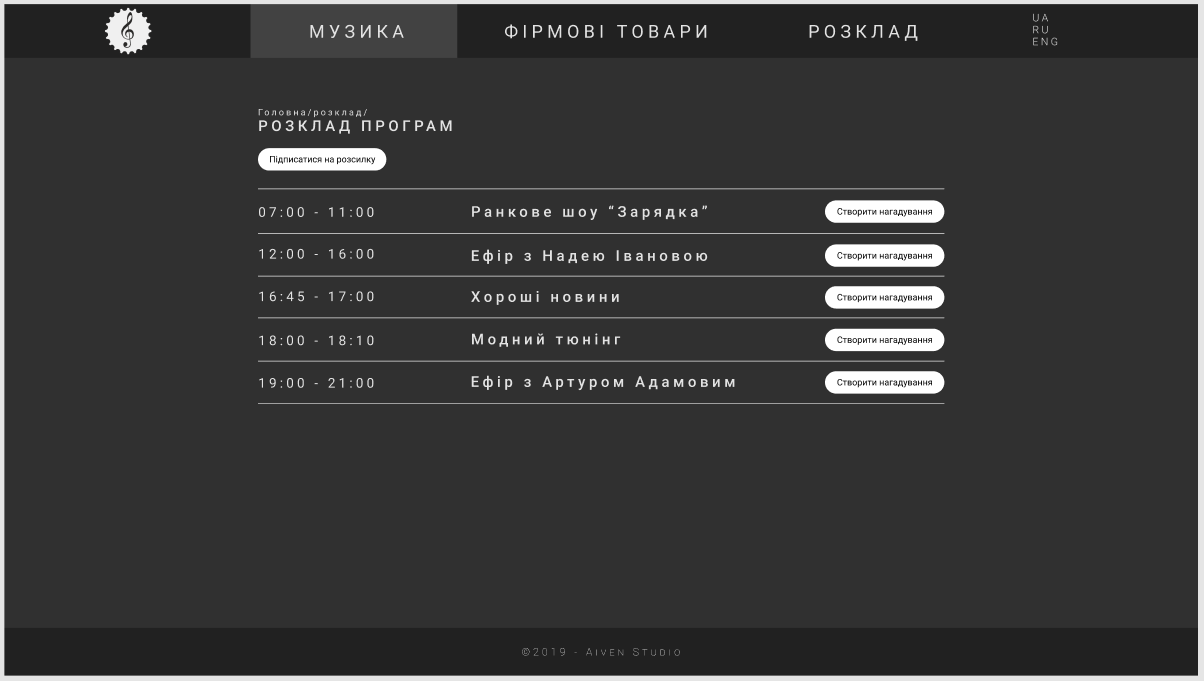
Прорисовка сторінки «Розклад пісень»



Прототип сторінки «Розклад програм»



Прорисовка сторінки «Розклад програм»



**2.4 Програмні засоби**

**StarUML** - програмний інструмент моделювання, який підтримує UML

(Уніфікована мова моделювання). StarUML орієнтований на UML версії 1.4 і

підтримує одинадцять різних типів діаграм, прийнятих в нотації UML 2.0. він активно підтримує підхід MDA (Модельна керована архітектура), реалізуючи концепцію профілів UML. Середовище розробки StarUML чудово налаштовується відповідно до вимог користувача і має високий ступінь розширюваності, особливо в області своїх функціональних можливостей. Використання StarUML, одного з провідних програмних інструментів моделювання, гарантує досягнення максимальної продуктивності і якості програмних проектів.

Інструмент UML, який адаптується до користувача.

StarUML надає максимальну ступінь адаптації середовища розробки користувача, пропонуючи налаштування параметрів, які можуть впливати на методологію розробки програмного забезпечення, проектну платформу і мову.

Справжня підтримка MDA.

Розробка методології опису архітектури програм - важливий процес, який буде

розвиватися ще 10 або більше років. Намір OMG (Асоціація об'єктно-орієнтованого управління) полягає в тому, щоб використовувати технологію MDA (Модельна керована архітектура) для створення переносних моделей, на базі яких можна автоматично формувати платформо-незалежні моделі або програмні коди. StarUML повністю підтримує стандарт UML 1.4, нотацію UML 2.0 і концепцію профілів UML, дозволяючи створювати платформо-незалежні моделі. Користувачі можуть легко документувати їх за допомогою простих документних шаблонів.

Чудова розширюваність і гнучкість.

StarUML забезпечує чудову розширюваність і гнучкість. Він надає механізм

аддінів (add-in), щоб розширювати свої функціональні можливості. Цей механізм розроблений спеціально, щоб надавати доступ до всіх функцій моделі / мета-моделі за допомогою COM Automation і розширювати меню і набір властивостей елементів. Також, користувачі можуть створювати власні підходи і механізми згідно зі своїми власними методологіями. Програма може також бути інтегрована з будь-якими зовнішніми інструментальними

засобами.

Точна відповідність стандарту UML.

StarUML строго дотримується специфікації UML, розробленої OMG для

моделювання програм. Слід розуміти, що використання нерегулярного синтаксису UML в інформаційних проектах через 10 або більше років може привести до сумних наслідків. StarUML максимально відповідає стандарту UML 1.4 і слідує нотації UML 2.0 на основі стійкої мета-моделі.

Відкритий формат програмної моделі.

На відміну від багатьох існуючих програм, які використовують власні

неефективні формати файлу моделі, StarUML оперує файлами в стандартному форматі XML. Коди, написані в легких для читання структурах і форматах, можуть бути легко змінені з допомогою синтаксичного аналізатора XML. З огляду на факт, що XML є світовим стандартом, це, звичайно, велика перевага, яка гарантує, що програмні моделі залишаться корисними більше ніж через десятиліття.

Справжня підтримка MDA.

StarUML дійсно підтримує профілі UML. Це максимізує розширюваність UML, роблячи моделювання на UML застосовуваним навіть в області фінансів, оборони, електронної комерції, страхування і аеронавтики. Насправді можна створювати платформо-незалежні моделі (PIM), а платформо залежні моделі (PSM) і виконувані коди можуть бути завжди автоматично згенеровані на їх основі.

Застосування методологій і платформ.

StarUML використовує концептуальний підхід, який можна застосувати до будь-яких методологій / процесів. Легко створюються не тільки моделі під засоби розробки для конкретних платформ типу .NET або J2EE, але також і для інших основних структур програмних моделей (наприклад модель подання 4 + 1, і т.д.).

Чудова розширюваність.

Всі функції StarUML реалізовані відповідно до Microsoft COM. Будь-яка мова, який підтримує COM (Visual Basic Script, Java Script, VB, Delphi, C ++, C #, VB.NET, Python, і т.д.), може використовуватися, щоб викликати StarUML або розробляти інтегровані доповнення (Аддіни).

Програмна функція перевірки моделі.

Користувачі можуть допускати помилки в процесі моделювання. Такі помилки можуть дорого обійтися, якщо вони не будуть виправлені до заключної стадії формування коду. Щоб запобігти такі ситуації, StarUML автоматично перевіряє модель програми, розроблювану користувачем, полегшуючи раннє виявлення помилок і сприяючи бездоганній і повній розробці програмного забезпечення.

Корисні аддіни.

StarUML включає багато корисних аддінів з різними функціональними

можливостями: генерація вихідних текстів на мовах програмування, конвертація вихідних текстів в моделі, імпорт файлів Rational Rose, обмін модельної інформацією з іншими програмними засобами, з використанням XMI, підтримка шаблонів проектування. Ці аддіни надають додаткові функції, збільшують продуктивність, гнучкість і функціональну сумісність моделей.

**Figma** — крос платформний, векторний графічний редактор від компанії Figma. Призначений для проектування та створення інтерфейсів, також створює інтерактивні прототипи. Працює у двох форматах: у браузері та як клієнтський додаток на десктопі користувача.

Ключовою особливістю Figma є те, що вона зберігає онлайн-версії файлів, з якими працював користувач, та не має офлайн-версій. Figma є безкоштовною для індивідуальних користувачів і платною для фахових команд.

Даний редактор підходить як для створення простих прототипів і дизайн-систем, так і складних проектів (мобільні додатки, портали). Стійка модель розвитку Figma, спростила співробітництво в усьому процесі створення цифрових продуктів для дизайнерів, розробників, менеджерів і маркетологів.

До початку 2019 року Figma вийшла на 1 мільйон зареєстрованих користувачів, ставши серйозним конкурентом для традиційних графічних редакторів і засобів прототипування. Може відкривати популярні графічні формати, зокрема: Scalable Vector Graphics (SVG), SKETCH, також експортувати у ці формати PNG, JPG та SVG.

**ВИСНОВКИ**

В результаті виконання курсового проекту було спроектовано інформаційну систему Розклад радіостанції.

Отримано та закріплено знання з об’єктно-орієнтованого аналізу та моделювання (проектування) на візуальній мові UML.

За допомогою CASE-засобу StarUML було спроектовано базові діаграми UML, які застосовуються на всіх етапах концептуального, логічного та фізичного проектування архітектури автоматизованих інформаційних систем. Крім того було спроектовано графічний інтерфейс автоматизованої інформаційної системи на базі платформи Figma та розроблено динамічний макет за допомогою платформи Figma.