НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ «КПІ»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра програмного забезпечення комп’ютерних систем

КУРСОВА РОБОТА

з дисципліни "Основи веб-програмування*"*

Виконав: Романюк Сергій Олександрович

Група: КП-51

Допущено до захисту

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1 семестр 2016/2017

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ «КПІ»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра програмного забезпечення комп’ютерних систем

Узгоджено                  ЗАХИЩЕНА "\_\_"\_\_\_\_\_\_\_\_\_2016р.

з оцінкою\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Керівник роботи

\_\_\_\_\_\_\_\_\_\_/Гадиняк Р.А./

\_\_\_\_\_\_\_\_\_\_/Гадиняк Р.А./

Аудіохостинг “The Bands”

Виконавець роботи

Романюк Сергій Олександрович

\_\_\_\_\_\_\_\_\_\_\_\_\_\_2016р.

**Вступний опис системи**

“**The Bands**” (*thebands.com*) – музикальний портал, аудіохостинг, на якому відвідувач сайту може безкоштовно слухати пісні в онлайн-режимі та завантажувати музику у форматі .mp3. Завантаження і прослуховування пісень є абсолютно безкоштовними послугами і потребують лише безкоштовної реєстрації.

Використовуючи зручну систему пошуку, користувач легко знайде саундтреки, збірники й альбоми цікавих для нього виконавців.

Музикальна колекція даного порталу включає в себе композиції різних жанрів, стилів й направлень. Нові треки, хіти, класика - кожен день щось нове з’являється на сайті.

Також, на ресурсі присутня можливість реєстрації. Користувачі, які ввійшли у систему, матимуть доступ до основного функціоналу сайту.

**Основний функціонал системи**

Даний сайт можна класифікувати як аудіохостинг і, відповідно, відвідувачі сайту будуть отримувати доступ до медіаконенту, розміщеному на ньому.

Таким чином, основними функціями, які будуть доступні любому типу користувачів, будуть:

* Прослуховування музики (аудіофайлів), яка знаходиться на сервері;
* Завантаження цих аудіофайлів;
* Перегляд структурованої інформації про різних виконавців, їх альбоми, треки;
* Пошук контенту за різними критеріями;
* Інші функції.

Також, на сайті буде присутньою реєстрація, і загалом, окрім незареєстрованих відвідувачів сайту, існуватимуть два типи користувачів: звичайний зареєстрований користувач та адміністратор сайту.

Зареєстровані користувачі отримуватимуть додаткові функції, наприклад:

* Змогу завантажувати контент альбомами, а не окремими треками;
* Інші функції.

**Опис сутностей і зв’язків бази даних**

*Опис сутностей*:

* “band” – об’єкт, який містить інформацію про певного музичного виконавця (групу чи музикальний проект).  
  Відповідні поля:
  + name
  + formed
  + members
  + genre
  + albums // amount of albums
  + description
  + albums\_array // links to ’albums’ collection
* “album” – об’єкт, який містить інформацію про певний альбом певного музичного виконавця.  
  Відповідні поля:
  + name
  + rls\_date // release date
  + artist
  + genre
  + tracks // amount of tracks
  + tracks\_array // links to ’tracks’ collection
* “track” – об’єкт, який містить інформацію про певний трек з певного альбома певного музичного виконавця.  
  Відповідні поля:
  + name
  + artist
  + duration
  + file\_size
* “user” – об’єкт, який містить інформацію про зареєстрованого користувача сайту.  
  Відповідні поля:
  + name
  + email
  + username
  + password // as hashed string
  + access\_level

*Зв’язки бази даних*:

**MongoDB**

*Зв’язки вищеописаних колекцій*:

У кожній з колекцій є унікальний ідентифікатор, генерація якого реалізована вбудованими функціями – \_id (тип – ObjectId). Слід зауважити, що ідентифікатор унікальний не тільки в межах своєї колекції, а й в межах усієї бази даних.

Колекція ‘bands’ має у собі масив документів – ‘band’, в кожному з яких є поле ‘albums\_array’ – масив, елементами якого є поля типу ObjectId, в яких містяться посилання на відповідні альбоми, які належать цій групі.

Колекція ‘albums’ має у собі масив документів – ‘album’, в кожному з яких є поле ‘tracks\_array’ – масив, елементами якого є поля типу ObjectId, в яких містяться посилання на відповідні треки, які містяться у цьому альбомі, а також поле ‘band’ типу ObjectId, в якому міститься посилання на відповідну групу, якій належить цей альбом.

Колекція ‘tracks’ має у собі масив документів – ‘track’, в кожному з яких є поле ‘album’ типу ObjectId, в якому міститься посилання на відповідний альбом, якому належить цей трек.

**Опис веб-сторінок і переходів між ними зі скріншотами**

*Інформація про сайт та неіснуюча сторінка*:

* /about
* /error404

*Посилання, пов’язані з входом у систему та користувачами (/users/route), де “/route”*:

* /register
* /login
* /profile/{username}

*Пошук та інформація про контент, розміщений на сайті*:

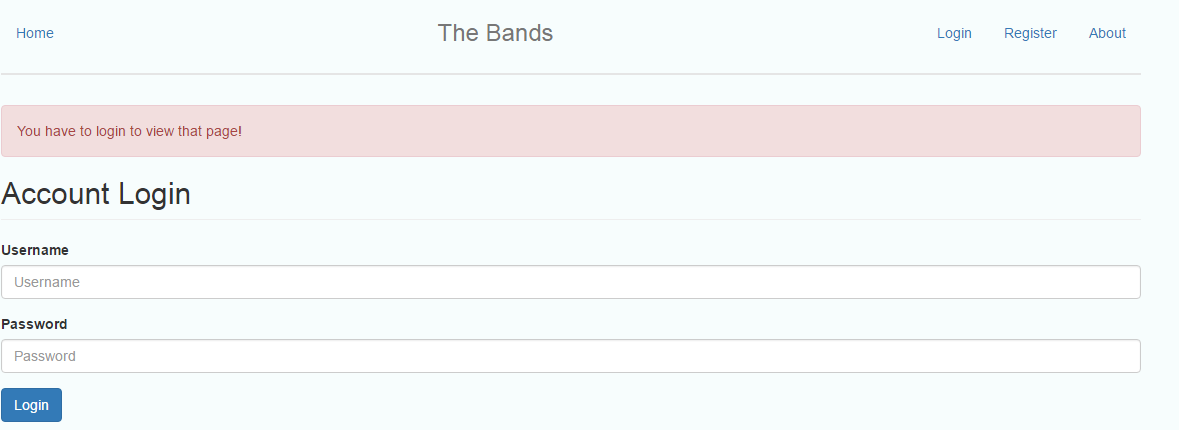
* Пошук: реалізується за заданими нижче сторінками за допомогою query string;
* /artist/{artist\_id}
* /album/{album\_id}/
* /song/{song\_id}

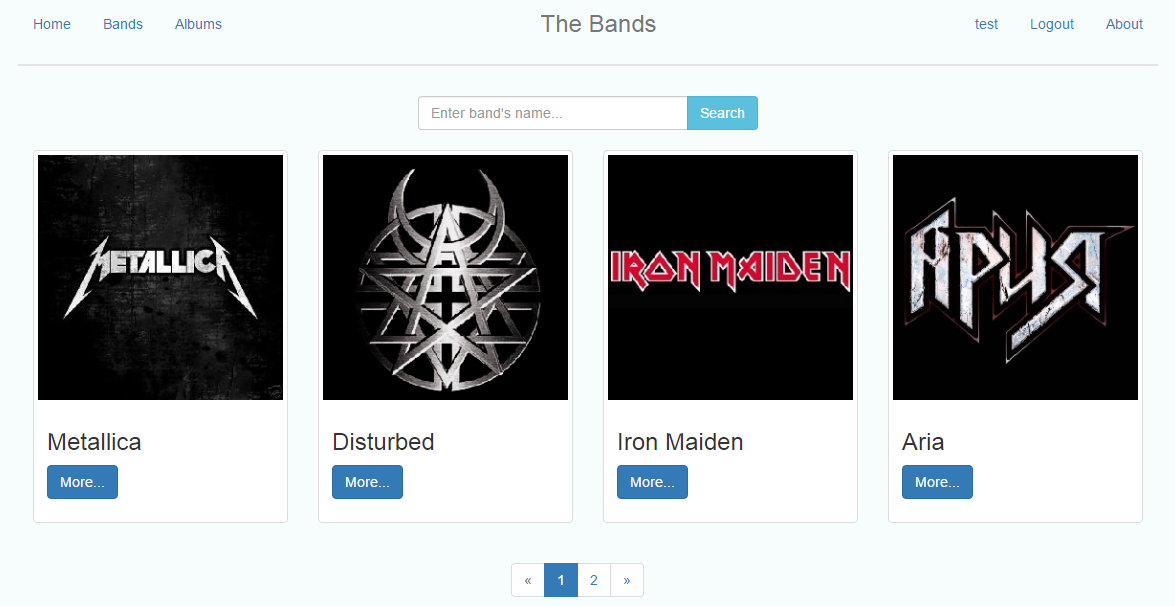
*Налаштування профілю*:

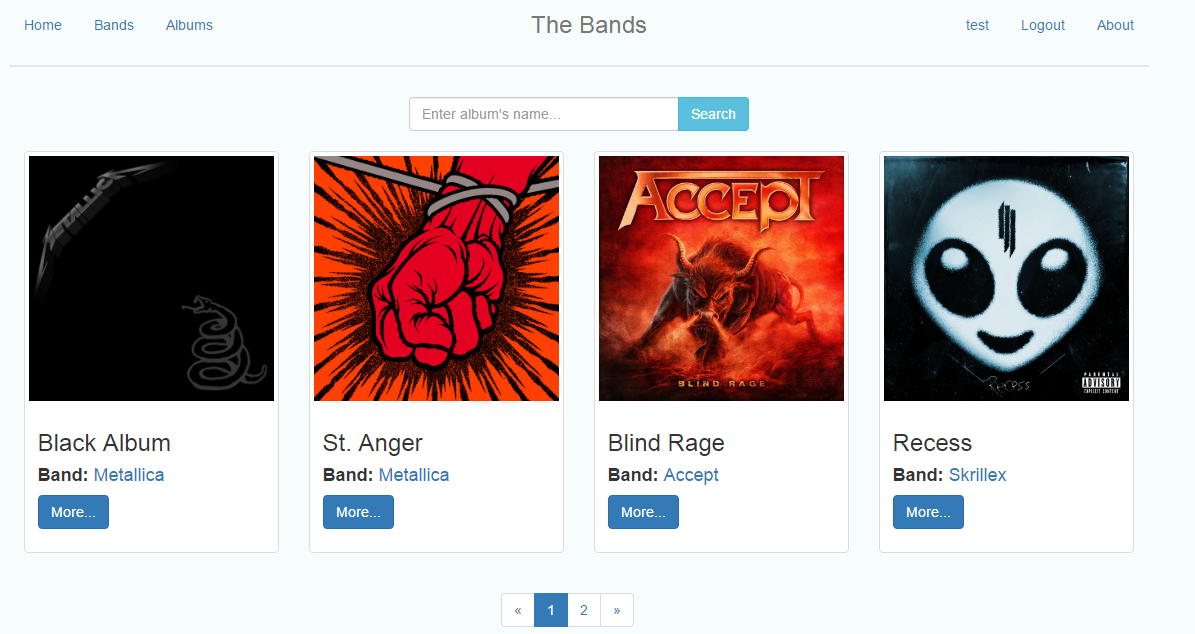
* Налаштування профілю відбувається за допомогою відправлення POST-запиту (за допомогою POST-форм) на /profile/{username}.

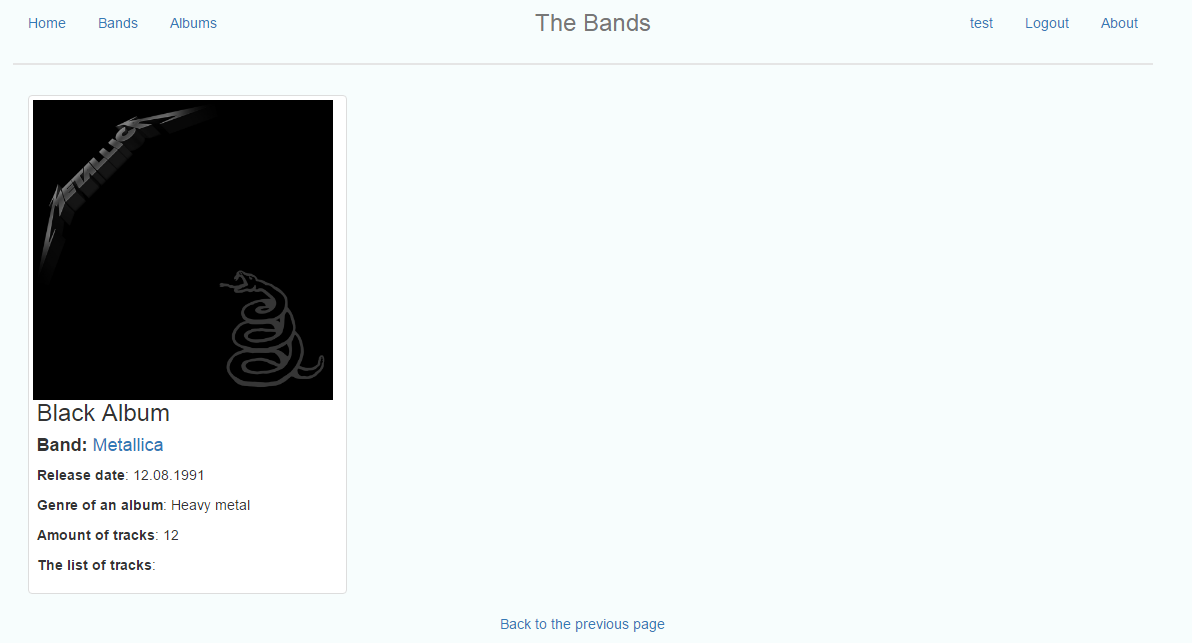
**Скріншоти**











**Опис мови програмування, фреймворку та основних бібліотек**

Back-end мова програмування: JavaScript.

Front-end мова програмування: JavaScript.

База даних: MongoDB.

* *Робота з MongoDB через JavaScript: модуль mongoose.*

Фреймворки та основні бібліотеки:

* Шаблонізатор: EJS
* Node.js
* Back-end реалізація серверу: Express
* Back-end реалізація авторизації і сесій: PassportJS
* Front-end форматування і вивід: Bootstrap
* Інші бібліотеки

**Опис роботи із RESTful API**

Можливі API-команди за певними шляхами:

* Шлях “*/api/bands*”:
  + Метод *GET* (**query string**: *?page=n* де n – ціле число): отримання деякої кількості (максимум 4 за один запит) груп за певною сторінкою (номер сторінки задається парою ключ-значення у query string**)**;
  + Метод *POST* (**body**: “bandName”: “Lol”,“bandFormed”: “1991-01-01”,“bandMembers”: “5”,“bandGenre”: “SampleGenre”,“bandDescription”: “Some Description”, “bandAlbums”: “8”): додання нової групи до списку груп (інформація задається парами ключ-значення у body);
* Шлях “/*api/filterBands*”:
  + Метод *GET* (**query string**: ?bandName=Lol&bandFormed=1991-01-01&bandMembers=5&bandGenre=SampleGenre&bandAlbums=8): отримання відфільтрованої кількості груп за певними значеннями у полях. Критерії відбору задаються парами ключ-значення у query string у довільній кількості;
* Шлях *“/api/bands/:band\_id\_param*”, де band\_id\_param – параметр URL-строки, який є унікальним ідентифікатором певної групи:
  + Метод *GET*: отримання певної групи за її унікальним ідентифікатором;
  + Метод *DELETE*: видалення певної групи за її унікальним ідентифікатором;
  + Метод *PUT* (**body**: “bandName”: “Lol”,“bandFormed”: “1991-01-01”,“bandMembers”: “5”,“bandGenre”: “SampleGenre”,“bandDescription”: “Some Description”,“bandAlbums”: “8”): оновлення інформації певної групи за її унікальним ідентифікатором (інформація задається парами ключ-значення у body).