МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

МАТЕРИАЛЫ

VII Международной молодежной научной конференции

«МАТЕМАТИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ИНФОРМАЦИОННЫХ, ТЕХНИЧЕСКИХ И ЭКОНОМИЧЕСКИХ СИСТЕМ»

Томск, 23–25 мая 2019 г.

Под общей редакцией
кандидата технических наук И.С. Шмырина

Томск
Издательский Дом Томского государственного университета
2019

4. *Nagle T.T., Holden R.K.* The Strategy and Tactics of Pricing // A Guide of Profitable Decision Making, 2nd edition, Prentice-Hall, Englewood Cliffs, NJ. – 1995.
5. *Comley P.* What do we know about Pricing research // WARC, Admap. – January 1997.
6. *Zenkova Zh., Krakovetckaia I.* Marketing of New Bio-Energy Drinks // Atiner conference paper series No: SME2013-0739. –2013.
7. *Zenkova Zh, Krakovetckaia I.* Marketing of New Bio-Energy Drinks // Theoretical and Empirical Reflections in Marketing. Athens Institute for Education & Research, ATINER. Greece. – 2014. – P. 179–192.

ONLINE PORTFOLIO SERVICE FOR MUSIC CONTENT CREATORS

V. Ivanov, A. Shkurkin
Tomsk State University

Introduction

With the advent of the Internet and the spread of online music platforms, free access to a huge amount of musical materials, instruments and opportunities has appeared. Music industry revenue grows nowadays [1]. It has led to the growth of people involved in music art.

There are still no acceptable ways for musicians to find each other. Such ways today are social networks and web applications primarily with the ability to download audio or video. But they are not designed specifically so that musicians can conveniently search for other users or share their skills and work.

So, the goal is to create a web-site that provides the functionality of creating an online portfolio for people involved in the creation of musical content.

This goal defines next objectives:

1. The definition of the main functionality.
2. Selection and study of development tools.
3. Creating a design layout.
4. Web application development.

1. Analysis of existing solutions

Consider some popular web applications.

«Soundcloud.com» [2]

It's an online platform for digital distribution of sound information with social networking features. Soundcloud provides users with uploading music or podcasts and some social network features like commenting and sharing. On Fig. 1 the main page of «Soundcloud.com» is represented.

«Bandcamp.com» [3]

This application also provides artists and labels with uploading music but it's more digital distribution platform than online-portfolio. Users can control how they sell their releases, setting their own prices, offering fans the option to pay more (which they do 40% of the time) and selling merchandise. The advantage of this application is advanced search and opportunity to sell music. On Fig. 2 the main page of «Bandcamp.com» is represented.

After analyzing the above and similar web-applications, it comes to conclusion:

– Applications don't provide instruments for mentioning users that participated in project.
– There is no detailed search with filters.

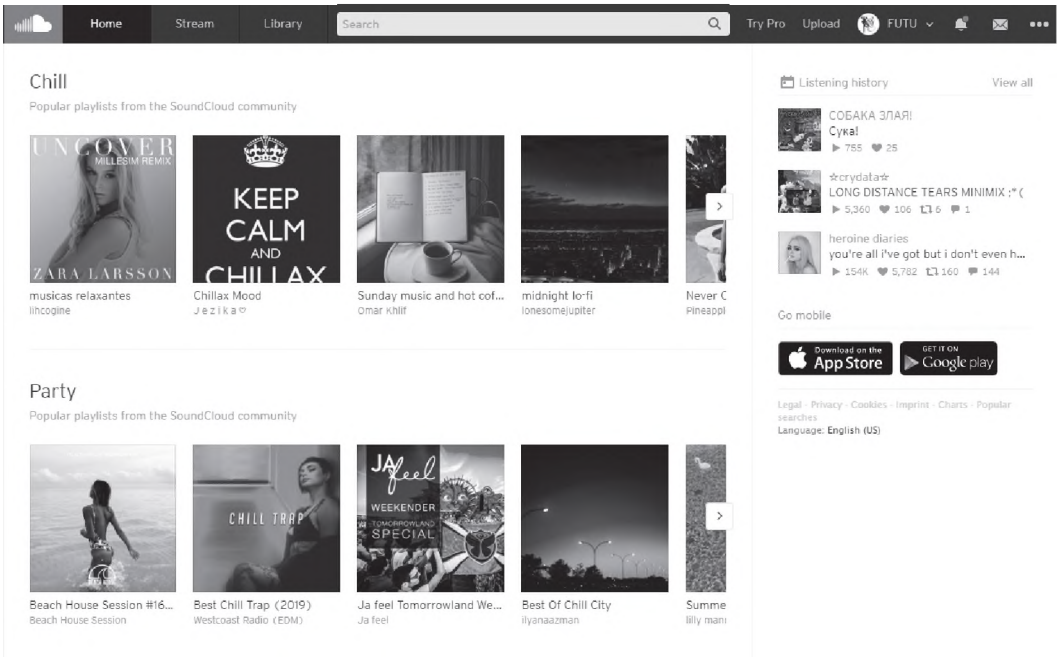

Fig. 1. The main page of «Soundcloud.com»

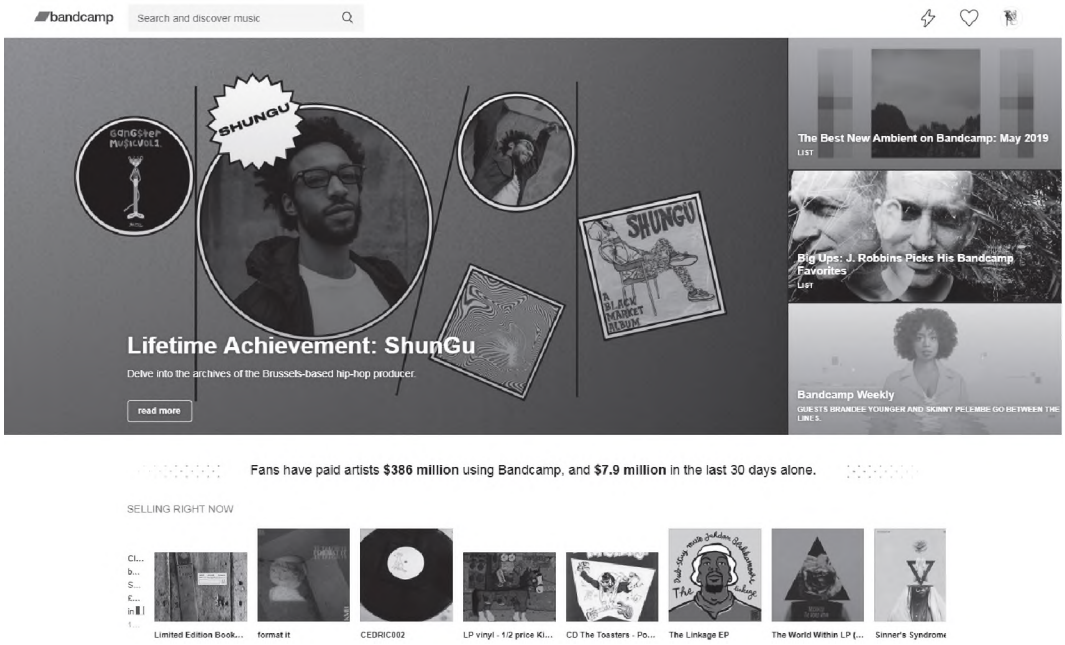

Fig. 2. The main page of «Bandcamp.com»

2. Domain modelling

The first step of web application modelling is domain modelling which shows the main details of system. Based on the fact that Javascript is not an Object Oriented Programming language It's hard to show all the relations between entities and business logic so on the Fig.3 you can see how Mongoose classes look like.

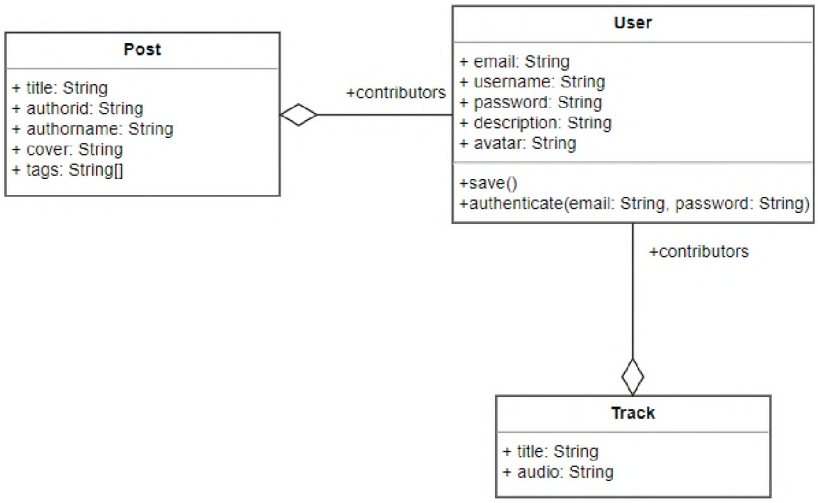

Fig. 3. Domain model of web application

Description of classes is next:

1. «Post» is a specific project and it consists of 6 attributes:
– Title of project, string;
– Authorid: id of author who created projects, string;
– Authorname: name of performer, string;
– Cover: name of project cover image file, string;
– Tags: array of string;
– Contributors: contains of id of participated users, array of string.
2. «User» consists of 5 attributes:
– Email: email of user that used to login, string;
– Username: name of user, string;
– Password: encrypted password of user, string;
– Description: description of user profile, string;
– Avatar: name of avatar image file, string.
3. «Track» represents certain song and it contains 3 attributes:
– Title: name of song, string;
– Audio: name of song audio file, string;
– Contributors: contains of id of participated users, array of string.

3. Defining of requirements

For the implementation of web application, the following requirements were formed:

1) Ability to create and manage personal portfolio page. It means that user can customize his own profile with avatar and information;

2) The function of adding project to the portfolio. When adding a project user has to upload cover of project, audio files and add information;

3) The function of mentioning persons involved in the creation of musical project;

4) Ability to search projects and content creators;

5) Home page with the latest projects of users.

4. Use case diagram

The origin and conceptual representation of the system in designing and development process is use case diagram. It helps to determine all main functional moments of interaction between system and user.

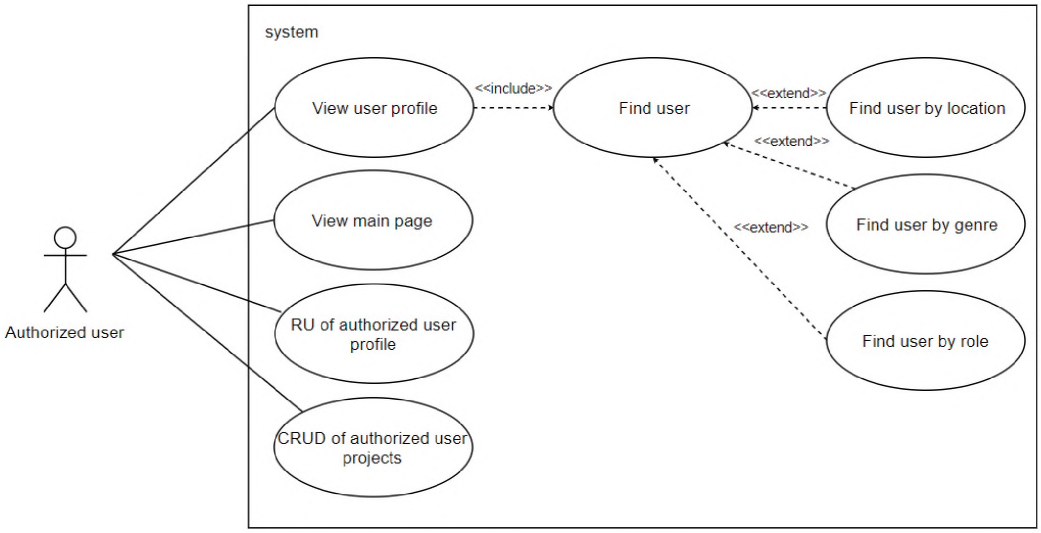

Fig. 4. Authorized user use case diagram

There are 3 roles in web-application – Authorized User (Fig. 5), Unauthorized User and Administrator (Fig. 6). The main actor of web application is authorized user so he has access to all the features of site while unauthorized user has not. The only difference between them it's that authorized user can manage his profile and create his own projects. Administrator has full rights and can manage all projects.

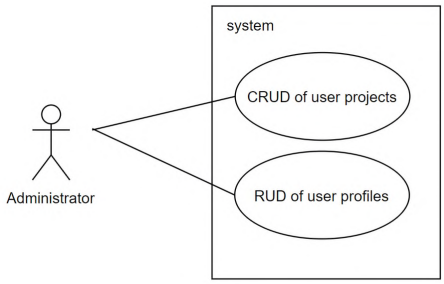

Fig. 5. Administrator use case diagram

5. Implementation tools

For server part of web application Node.js was chosen. Node.js is an open-source, cross-platform JavaScript run-time environment that executes JavaScript code outside of a browser. It gives programmer an ability to create server-side applications using Javascript. It has a huge number of additional libraries which expand opportunities for building applications and make development faster.

Express is a web application framework for Node.js designed for building web applications and APIs, which helps to organize web-application into MVC-architecture on server side. For creating a front-end template engine called Jade with UI framework Bootstrap 4 were used.

For development of database MongoDB was chosen. MongoDB – is a cross-platform document-oriented database program that classified as a NoSQL database program and it uses JSON-like documents. To manage relationships between data and to translate between objects in code and the representation of those object in MongoDB used Object Data Modeling (ODM) library Mongoose.

6. User's guide

The main page contains of user projects or releases with covers, names of projects and usernames. User can find other users, log in, sign up and click on projects for more information (Fig. 6).

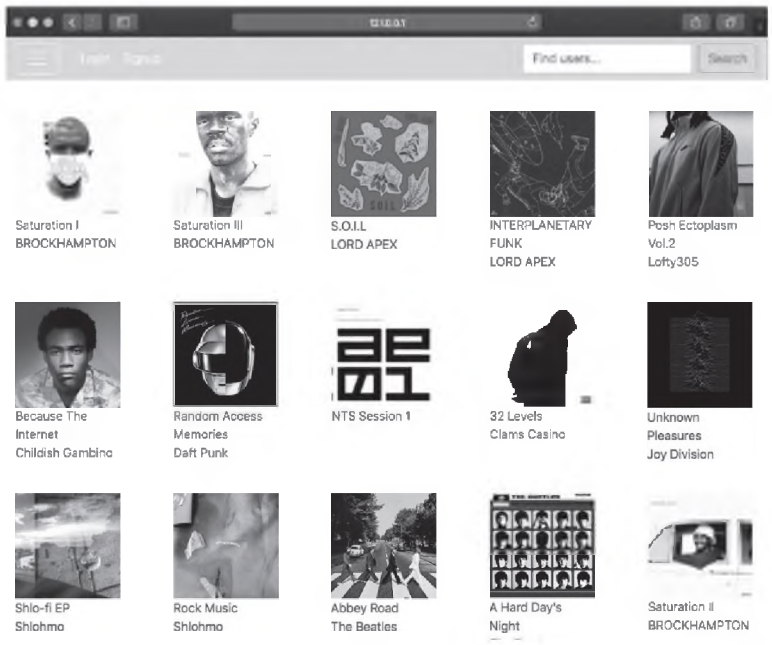

Fig. 6. The main page

The album page with elements such as name of release, name of user, all the participated users and some tags. Next to album information, we see songs with attached and playable audio files, their names and people involved in creating of song (Fig. 7).

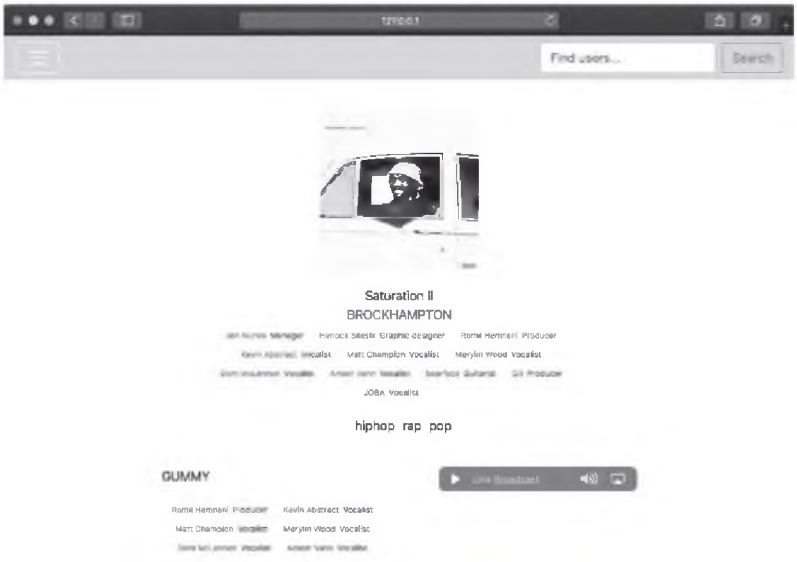

Fig. 7. Album page

Adding a project page provides with text fields such as title, tags, participated user names. Also users have to upload cover of project and choose roles of mentioned users. In

user profile users have ability to upload their avatars, to add information about them and contacts. Projects of user are presented under description block. (Fig. 8 and Fig. 9).

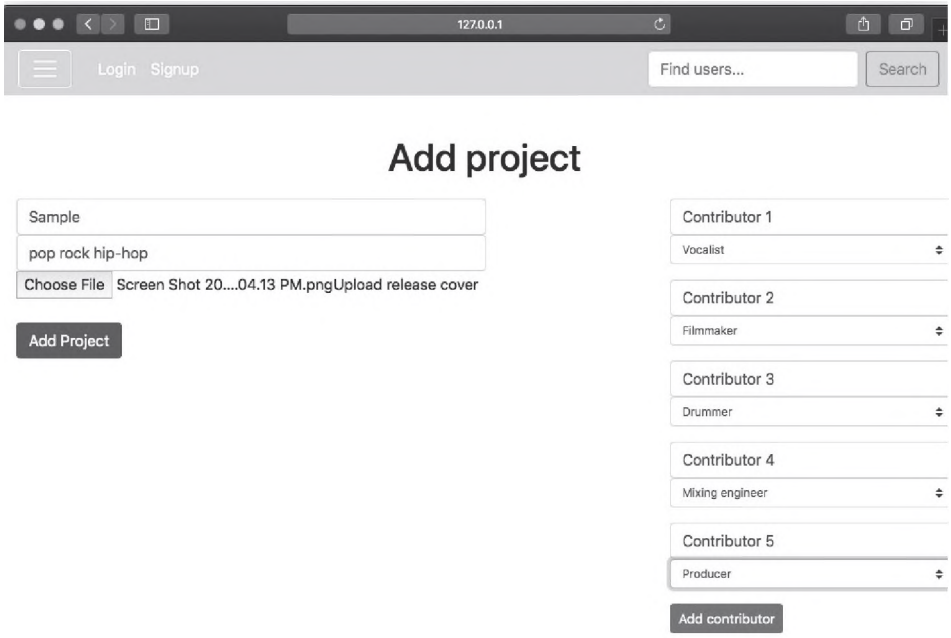

Fig. 8. Adding a project

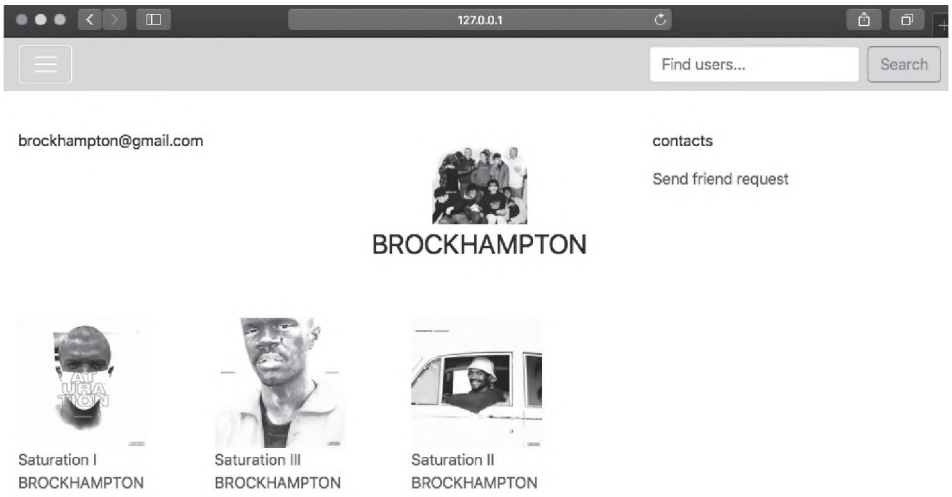

Fig. 9. User's profile page

Conclusion

The goal of this project was development of a web application that allows people involved in music to create online-portfolio to which they can attach their projects or be marked as participating in the creation of project. Also, this application has advantages compared to already existing solutions.

All objectives, which were defined in the beginning, are done. Application satisfies all requirements which were formed before implementation. So In that way, the goal and all objectives were accomplished.

REFERENCES

1. Global Music Report 2018 – IFPI, representing the recording industry worldwide. [online] Available at: https://ifpi.org/downloads/GMR2018.pdf [Accessed 31 May 2019]

2. Soundcloud.com – Music and podcast streaming platform. [online] Available at: URL: https://soundcloud.com/ [Accessed 31 May 2019]

3. Bandcamp.com – Music distribution service. [online] Available at: URL: https://bandcamp.com/ [Accessed 31 May 2019]

DETECTION OF UNDERLYING SURFACE, HORIZONTAL AND VERTICAL EDGES OF THREE-DIMENSIONAL OBJECTS BASED ON LASER SCANNING DATA

Y.L. Kostyuk, M.I. Litovchenko, A.A. Pichugina

Tomsk State University

Introduction

Laser scanning technology is widely used in various industries and solves a lot of production problems. The result of applying this technology is a three-dimensional point cloud. There is a lot of different software for handling obtained laser scanning data. An overview of a number of such software products is given in [1]. Each of these software products has various implemented algorithms for detection objects in point cloud, which are distinguished by detection accuracy, computational complexity, input data, user parameters, etc. So each algorithm has its own approach for handling point cloud and could be applied to detection certain objects.

A lot of algorithms are based on the idea of Hough Transform [2,3]. The main problem of using Hough Transform for detection objects is high computational complexity, which depends on how compound the object is. Therefore, actual task is to develop more efficient detection algorithms for some type-object.

1. Point cloud structuring

The first step includes preparation of point cloud for further research. Proposed approach of point cloud structuring in [4] allows to reduce the computation complexity of most detection algorithms to a linear order $O(n)$.

The space of point cloud is divided on voxels which mean cells in three-dimension spaces with regular size *d*. Each point of cellular partition uniquely belongs to a certain voxel: the point coordinates (x_i, y_i, z_i) are divided on cell size *d* and obtained indexes define the voxel. The list of entered points with their quantity are formed for each voxel.

2. Detection of the underlying surface

Underlying surface includes all components referring to the earth`s surface. In a simple case the underlying surface means the terrain surface. So the points reflected from the terrain surface have the lowest *Z* coordinates. Using obtained data structure it is easy to detect dead-level underlying surface by scanning the low level voxels.

If the surface presents ambiguous terrain, then it is necessary to use algorithm from [5]. Succinctly describing of algorithm is demonstrated as follows:

- consistently searching for cells in point cloud structuring that have voxel on zero level and collect them into list;
- checking eight neighboring cells for each item from certain list. If a neighboring cell has a voxel whose lower level is not more than 1 from current cell, then it is added to the end of the list.

The model of detected surface is represented by flat triangulation on selected points (Fig. 1). Handled points are deleted or ignored from point cloud for further research.