**Proposed Solution**

We want the administrators are easy to manage Musix Cloud, so we build the admin panel to achieve that. The admin panel will separate to four parts, regarding Administrators management, Users management, Music management and Admin Message, so that I will divide the admin panel to four parts to develop.

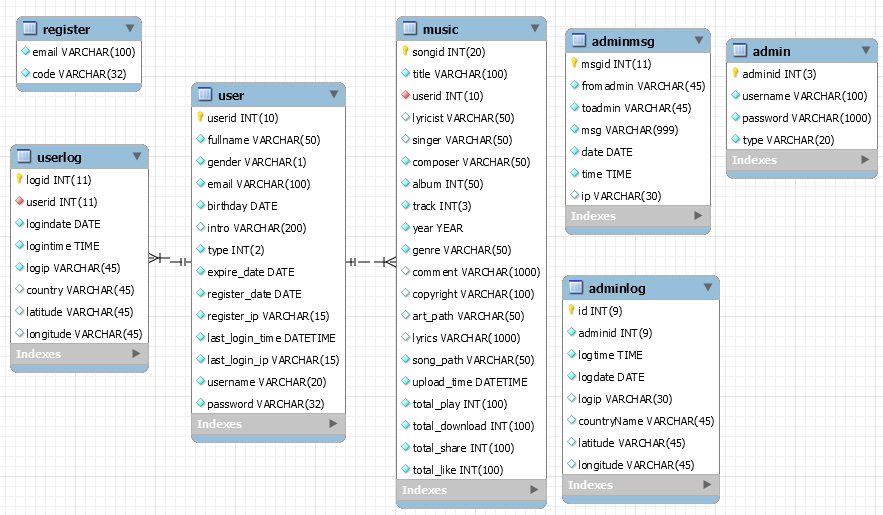
**The technically knowledge**

The knowledge that I will need to use to develop the admin panel are PHP, AJAX, jQuery, Javascript and Bootstrap.

**The function of the admin panel**

1. Administrator can add, delete and edit other Administrators username, password and the role level of Administrators. It can convenient the administrators easy to manage.
2. Each administrator login the admin panel will logging the Admin ID, Admin Username, Login Date, Login Time, Login IP address and Login location, that will ensure the administrators account are safe. If one of the admin account hack by someone that we can find out and make the reaction. The admin logging can export to xls, csv.
3. For manage the users, the Administrator can see the detail information of each users, that regarding User ID, Username, Full Name, Gender, Email, Birthday, membership type, the expire date of membership account, register date, register IP address, last login IP address ,last login time and date. The Administrators can block the users if the users break the rules.
4. Every user login will record the login logging, the logging will including the user ID, username, login date, login time, login IP address, country, latitude and longitude. The login logging can export to xls or csv.
5. For song management, administrator can delete the song that the song is not fit the requirement, for example the song lyric include some bad words, that we will delete.
6. To communication with other administrators, we build the Admin Message. All administrators can use this function to send and receive messages. Also the messages logging can export to xls and csv.

Architecture of the proposed system and roles of hardwares



Roles of users and the functions

Admin Panel

There are three type of role, including Superuser, Admin, Music Admin. Each role have their own permission.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Superuser | Admin (Normal) | Music Admin |
| Add Administrator | Y | N | N |
| Remove Administrator | Y | N | N |
| Edit Administrator | Y | N | N |
| View Administrator Logging | Y | N | N |
| View Users Detail | Y | Y | N |
| View Song | Y | Y | Y |
| Send Message to other Administrators | Y | Y | Y |
| Receive Message from other Administrators | Y | Y | Y |
| Export Administrator Logging | Y | N | N |
| Export Admin Message Logging | Y | Y | N |
| View User Logging | Y | N | N |

(Y=Yes, N=No)

Roles of users and the functions

User Panel

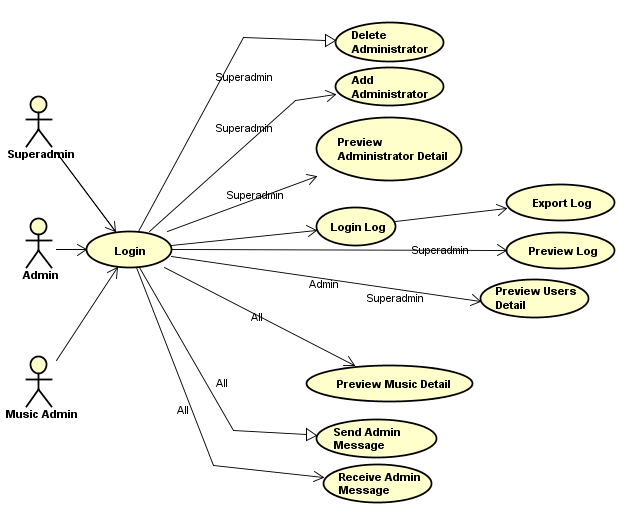
There are three types of end users to use the Musix Cloud.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Guest | Users  (Free Account) | Paid Users |
| Play Music | N | Partial | All Song |
| Upload Music | N | Y | Y |
| Download | N | N | Y |
| Music Mixer | N | N | Y |

(Y=Yes, N=No)

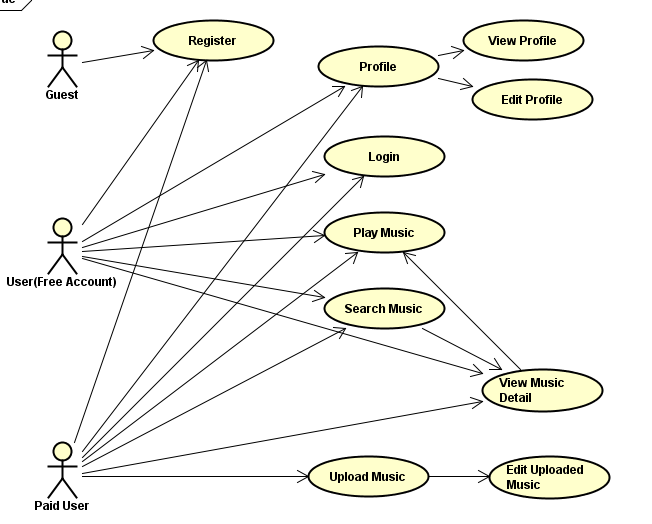
**Use Cases of the whole systems**

Admin Panel



**Use Cases of the whole systems**

User Panel



Advantages and drawbacks of the solution

Admin Panel

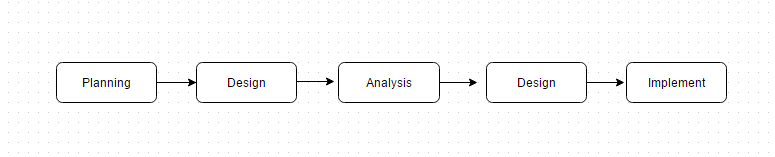
Advantages:

* The Admin panel can easy to manage the other admin, the users and the music.
* Once the user upload the song that is not fit the requirement, we can delete it easily, no need to find out the link and warning the user.
* Improve the communication between administrators.

Drawback:

* Some of the uses information will leakage once the process steps are wrong.

Non-functional Requirements



Admin Panel

* User friendly
* Provide Mobile User-interface or responsible user interface
* Smooth response

User Panel

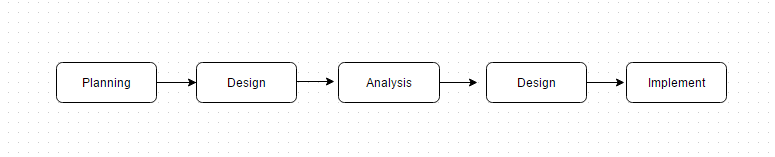
* Multi-language
* User friendly
* Provide multiple user interface
* Provide good user experience

**Software Process Model**

**Ways of Information Gathering**

1. Research, search the easiest way to develop
2. 3 Meeting per weeks
3. Survey

Methodology of Software Development Life Cycle



Plan to develop the admin panel, include the function of admin panel and the UI.

Design the User Interface

Analysis the way to develop and the order

Use Bootstrap to design the Admin user interface

Implement the planning.

**Seclude**

|  |  |
| --- | --- |
| Initial | Oct 22, 2015 |
| Admin Login (with Logging the IP Address to Database) | Oct 23, 2015 |
| User Interface Design | Oct 24, 2015 |
| Function: [Create, Delete Admin](https://github.com/rickywonghv/fypmusic/commit/74900b2840b34b79ba94cf7180befb3372e38f2e) | Oct 25, 2015 |
| Function: Edit Admin | Oct 26, 2015 |
| Add Change Password to Index | Oct 26, 2015 |
| View User Detail | Oct 26, 2015 |
| Send and receive Admin Message | Oct 27, 2015 |
| View Admin Logging | Nov 7, 2015 |
| View and delete Music Detail | Nov 9, 2015 |
| Export Admin Logging and User Logging | Nov 11, 2015 |
| Permission of function and different pages | Nov 12, 2015 |
| New Login User Interface | Nov 15, 2015 |

**Software tools, facilities and hardware needed during development**

**Software tools:**

* Sublime Text 3
* Jetbain PHP Storm
* MySQL workbench
* WAMP (Windows)
* LAMP (Ubuntu)

Hardware:

* Server

Service:

* Digital Ocean (Web server)
* Amazon AWS RDS (Database)(MySQL 5.6)

|  |  |
| --- | --- |
| Specification (server):   * 1GB Memory * 1 Core Processor * 30GB SSD Disk * 2TB Transfer | Specification (Database):  Vcpu: 1  Memory :1GiB  Storage:5GB SSD |