

Software Testing

Music Application (I - Hear)



Kelompok 15

Heru Purnama (065118168)
Wita Setianingsih (065118134)
Zufar Mahasin Naufal (065118167)

Program Studi Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Pakuan
Bogor

Daftar Isi

BAB I.....	3
Pendahuluan.....	3
1.1 Ruang Lingkup.....	3
1.2 Rumusan Masalah.....	3
1.3 Tujuan Pengembangan.....	3
BAB II.....	4
Landasan Teori.....	4
2.1 Software – Testing.....	4
2.2 Definisi Website.....	4
2.2.1 Web Static.....	4
2.1.1 Web Dinamis.....	4
2.3 Database Management System - (DBMS).....	5
2.4 API - (Application Programming Interface).....	5
2.5 SDLC – Software Development Life Cycle.....	6
BAB III.....	7
Metode Analisis.....	7
3.1 Arsitektur konsep.....	7
3.2 Email Verification.....	7
BAB IV.....	8
Metode Perancangan.....	8
4.1 Flowchart / alur diagram.....	8
4.2 Database Relational.....	8
4.3 Wireframe.....	9
4.4 Mock-Ups.....	12
4.5 Use Case Diagram.....	13
BAB V.....	14
Pembahasan Project.....	14
5.1 Fitur - fitur aplikasi.....	14
BAB VI.....	22
Kesimpulan & Saran.....	22
Daftar Pustaka.....	23

BAB I

Pendahuluan.

Abstrak.

Secara garis besar musik merupakan media untuk meluapkan isi kata hati seseorang. Setiap orang tentu mempunyai tipe karakter music yang berbeda – beda, tentunya setiap music yang diputar mempunyai arti yang mendalam bagi penikmat music itu sendiri. jenis kategori music tidak hanya berfokus pada 1 titik genre saja, seiring perkembangan nya waktu & zaman music terus berkembang pesat layaknya teknologi saat ini.

1.1 Ruang Lingkup.

1.2 Rumusan Masalah.

Permasalahan – permasalahan pada aplikasi teruntuk tugas software testing ini meliputi beberapa hal di antara nya :

- a. Kekurangan – kekurangan apa saja yang terdapat pada aplikasi I – Hear ini ?
- b. Teruntuk format track, di aplikasi ini jenis format apa saja dalam pemutar audio ?

1.3 Tujuan Pengembangan.

Dalam membangun sebuah aplikasi agar terbilang matang dan tidak terpaku pada hal lain, disini kami memfokuskan titik tujuan dari aplikasi I-Hear di antaranya adalah :

- a. Memenuhi prosesur mata kuliah (Software – Testing).
- b. Memberikan fasilitas media streaming bagi pencinta musik.
- c. Menjadikan aplikasi bersifat multi-cross platform.
- d. Membentuk ke kompakan dalam sebuah team-mate.
- e. Menjadikan project ini sebagai bahan referensi yang bermanfaat bagi setiap orang ketika hendak membangun Aplikasi berbasis web.
- f. Meng-improve softskill & hardskill setiap participant dalam terlibat project.
- g. Menerapkan design interface yang minimalist sehingga tidak mempersulit user.

BAB II.

Landasan Teori.



Seiring timelinenya teknologi, kebutuhan sistem informasi sangat dibutuhkan terutama pelaku di bidang teknologi informasi. Studi kasus yang akan kami garapkan ini bertema kan Music Streaming, seperti hal nya platform musik ternama dunia layaknya I-Tunes, Spotify, Amazon Music dan sebagainya.

2.1 Software – Testing.

Software testing merupakan proses mengeksekusi program atau aplikasi dengan maksud untuk menemukan bug dari suatu perangkat lunak yang dibuat. Software testing membantu menyelesaikan aplikasi perangkat lunak atau produk terhadap bisnis dan kebutuhan pengguna. Hal ini sangat penting untuk memiliki cakupan tes yang baik untuk menguji aplikasi perangkat lunak sepenuhnya dan membuatnya yakin bahwa itu bekerja dengan baik dan sesuai dengan spesifikasinya.

2.2 Definisi Website.

sebuah kumpulan halaman pada suatu domain di internet yang dibuat dengan tujuan tertentu dan saling berhubungan serta dapat diakses secara luas melalui halaman depan (home page) menggunakan sebuah browser menggunakan URL website. Jenis kategori website terbagi menjadi 2 bagian di antaranya :

2.2.1 Web Static.

Dimulai dari pembuatan website tradisional. Kenapa disebut tradisional? Karena dulu orang kalau ngoding web pasti membuat sebuah web yang statis. Web statis bisa dibilang lebih baik dalam hal performa dari web dinamis. Kecepatan load yang dimiliki oleh sebuah web statis juga keamanan yang lebih tahan dari serangan hacker menjadikan beberapa orang masih memilih untuk membuat sebuah website statis.

2.1.1 Web Dinamis.

Web dinamis merupakan sebuah website yang memungkinkan penggunanya untuk berinteraksi secara langsung, dalam artian pengguna dapat menambah, memodifikasi, ataupun menghapus konten di dalam sebuah web tanpa harus membuka struktur kode dari web tersebut.

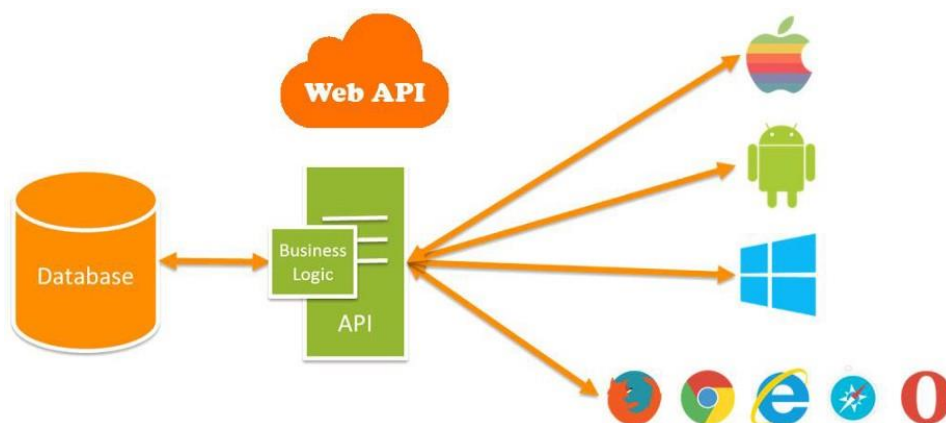
2.3 Database Management System - (DBMS).



Sistem manajemen basis data atau kadang disingkat SMD, adalah suatu sistem atau perangkat lunak yang dirancang untuk mengelola suatu basis data dan menjalankan operasi terhadap data yang diminta banyak pengguna. Contoh tipikal SMD adalah akuntansi, sumber daya manusia, dan sistem pendukung pelanggan, SMD telah berkembang menjadi bagian standar di bagian pendukung (back office) suatu perusahaan. Contoh SMD adalah Oracle, SQL server 2000/2003, MS Access, MySQL dan sebagainya. DBMS merupakan perangkat lunak yang dirancang untuk dapat melakukan utilisasi dan mengelola koleksi data dalam jumlah yang besar.

DBMS juga dirancang untuk dapat melakukan manipulasi data secara lebih mudah. Sebelum adanya DBMS, data pada umumnya disimpan dalam bentuk flat file, yaitu file teks yang ada pada sistem operasi. Sampai sekarangpun masih ada aplikasi yang menyimpan data dalam bentuk flat secara langsung.

2.4 API - (Application Programming Interface).

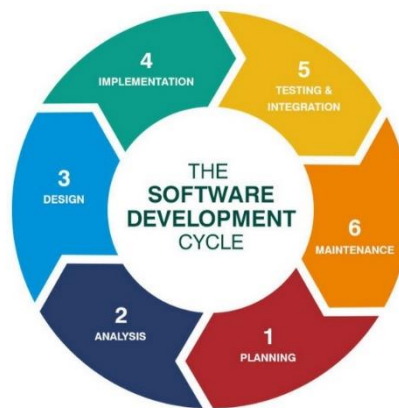


API adalah singkatan dari Application Programming Interface yaitu sebuah software yang memungkinkan para developer untuk mengintegrasikan dan mengizinkan dua aplikasi yang berbeda secara bersamaan untuk saling terhubung satu sama lain.

Fitur Web API :

- a. Mendukung banyak format teks, Seperti JSON, XML atau format apapun yang kamu inginkan, Akan tetapi kebanyakan digunakan ke dalam format JSON. Kamu juga bisa menggunakan MediaTypeFormatter untuk melakukan custom.
- b. Mendukung fitur MVC seperti routing, controllers, action results, filter, model, IOC container, dll.
- c. Web API dapat berjalan di Apache atau web server lainnya yang didukung sesuai bahasa pemrograman yang digunakan

2.5 SDLC – Software Development Life Cycle.



Pengembangan sistem (system development) dapat berarti menyusun suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada.

Definisi SDLC Serangkaian aktivitas yang dilaksanakan oleh professional dan pemakai sistem informasi untuk mengembangkan dan mengimplementasikan system informasi

Siklus hidup pengembangan sistem terbagi menjadi 6 fase di antaranya :

- a. Perencanaan system.
- b. Analisis sistem
- c. Design sistem.
- d. Implementasi.
- e. Testing informasi.
- f. Perawatan / maintainance.

BAB III.

Metode Analisis.

3.1 Arsitektur konsep.

Saat menggunakan alat apa pun di "dunia nyata", Anda akan merasa lebih percaya diri jika memahami cara kerja alat tersebut. Pengembangan aplikasi tidak berbeda. Saat Anda memahami bagaimana alat pengembangan Anda berfungsi, Anda merasa lebih nyaman dan percaya diri menggunakannya.

Tujuan dari dokumen ini adalah untuk memberi Anda gambaran umum tingkat tinggi yang baik tentang cara kerja kerangka kerja Laravel. Dengan mengenal keseluruhan kerangka kerja lebih baik, semuanya terasa kurang "ajaib" dan Anda akan lebih percaya diri membangun aplikasi Anda. Jika Anda tidak langsung memahami semua istilah, jangan berkecil hati Cobalah untuk mendapatkan pemahaman dasar tentang apa yang sedang terjadi, dan pengetahuan Anda akan berkembang saat Anda menjelajahi bagian lain dari dokumentasi.

3.2 Email Verification.

Banyak aplikasi web meminta pengguna untuk memverifikasi alamat email mereka sebelum menggunakan aplikasi. Alih-alih memaksa Anda untuk menerapkan ulang ini pada setiap aplikasi, Laravel menyediakan metode yang nyaman untuk mengirim dan memverifikasi permintaan verifikasi email.

3.3 System Requirement.

Sebelum memulai nya development aplikasi kita perlu menganalisis sistem dan tools apa saja yang akan di implementasikan ke sebuah sistem agar tidak terjadi halnya sistem under requirement. urutan requirement sistem yang ditubuhkan dibawah ini adalah :

a. Minimum System.

- Laravel versi 5.5
- Php versi 5.9
- Composer versi 1.5
- MySql Versi 5.9
- Adobe XD cc 2019 versi 25. 2. 12
- Ms.visio versi 2018.

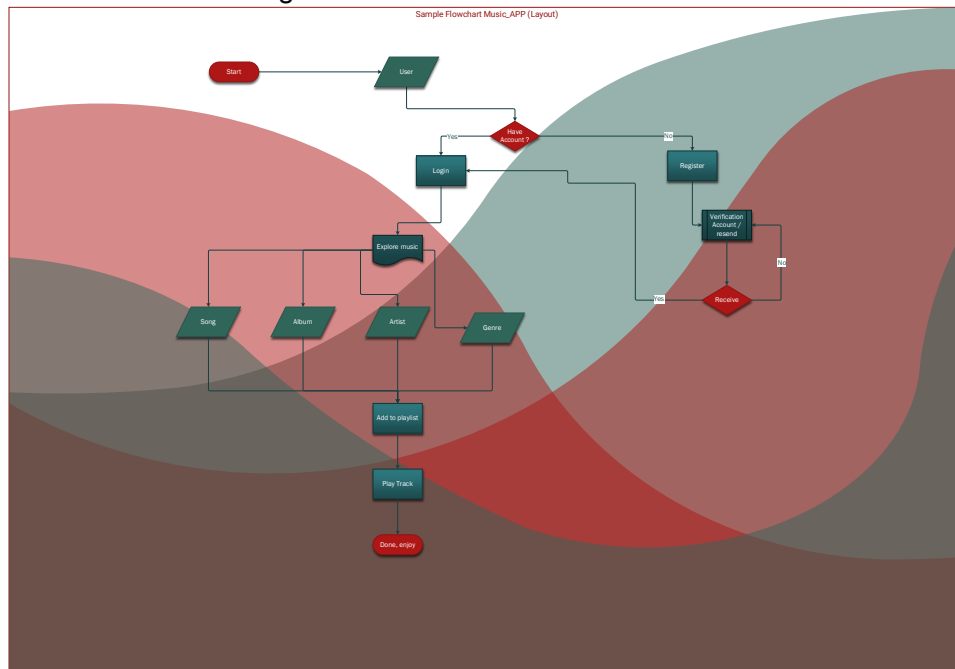
b. Recommended system.

- Laravel 5.5=> terbaru.
- Php versi 7.3+
- Composer versi 1.7>
- Mysql versi 7.3
- Adobe XD cc2020.
- Ms. Visio 2020.

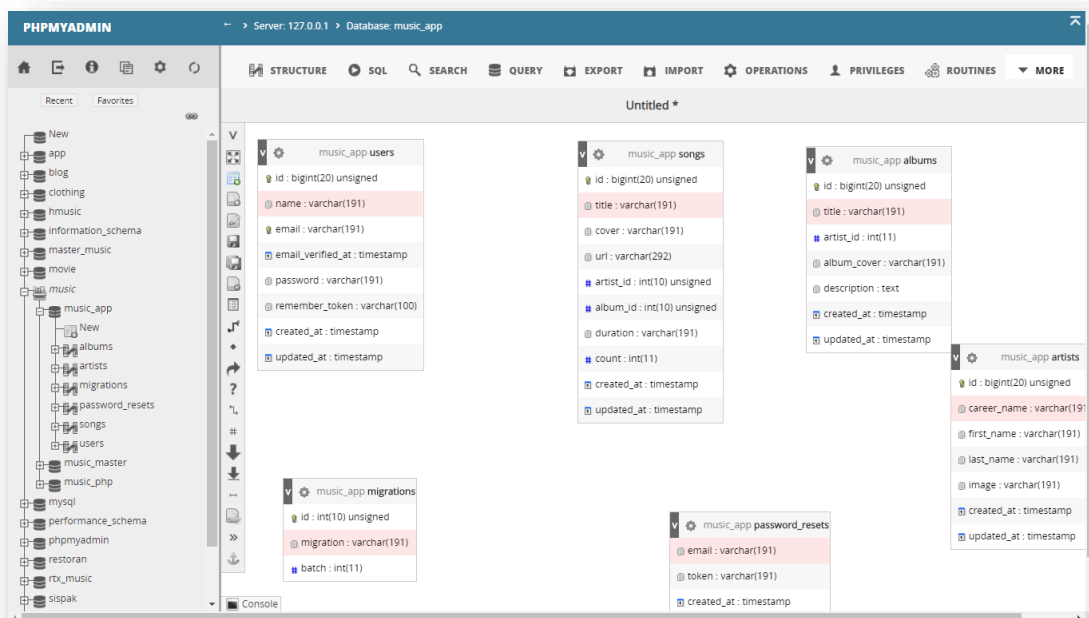
BAB IV.

Metode Perancangan.

4.1 Flowchart / alur diagram.



4.2 Database Relational.



Gambar diatas merupakan desain model database aplikasi musik yang terdiri atas 6 buah tabel diantaranya :

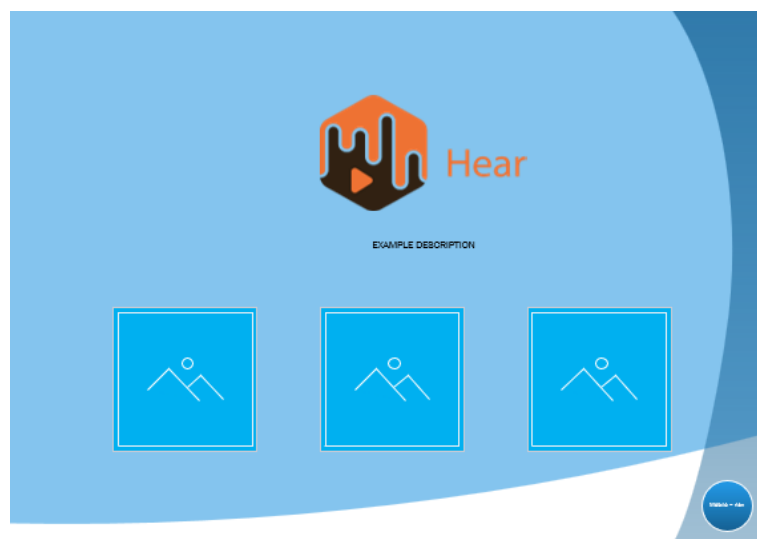
- Tabel Songs

- Users
- Password_reset
- Migration
- Albums
- Artist


Setiap entitas memiliki relasi terhadap table satu sama lain, seperti contoh tabel songs memiliki sifat one-to-many dengan table album,artist,users. Pada dasarnya sebuah aplikasi sangat erat kaitannya dengan management database karena terdapat sumber data. Dari penggarapan aplikasi ini kami sudah mereseach ke berbagai platform music mainstream bahwa aplikasi yang mereka gunakan juga menerapkan SDLC,Model prototype interface, RAD, model prosesperangkat lunak evolusioner.

4.3 Wireframe.

Desain Wireframe yang akan kami tuangkan dalam pengembangan aplikasi ini merupakan sketsa gambaran pada sebuah penempatan element web. Untuk perancangan wireframe disini menggunakan tools Microsoft Visio karena tools ini mudah di pahami.



Gambar di atas merupakan tampilan awal music app I-Hear



Sign in

[Forgot password?](#)

User ID

Password

☒ Show password

☐ Keep me signed in.


Sign in

New Account

Create new account

Sign up for SMS account

Sign – in Wireframe



Sign Up

Name

E-Mail Address

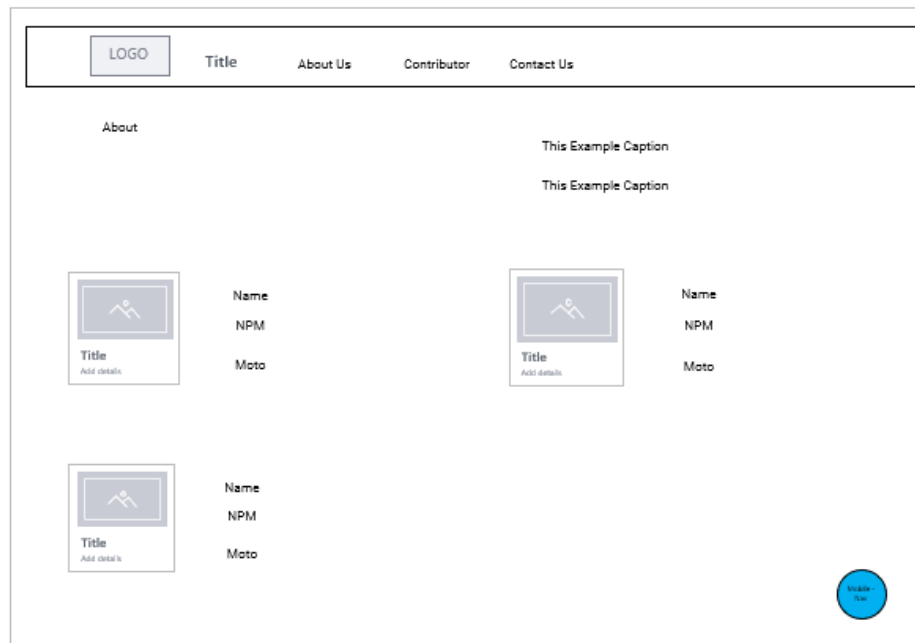
Address

Password

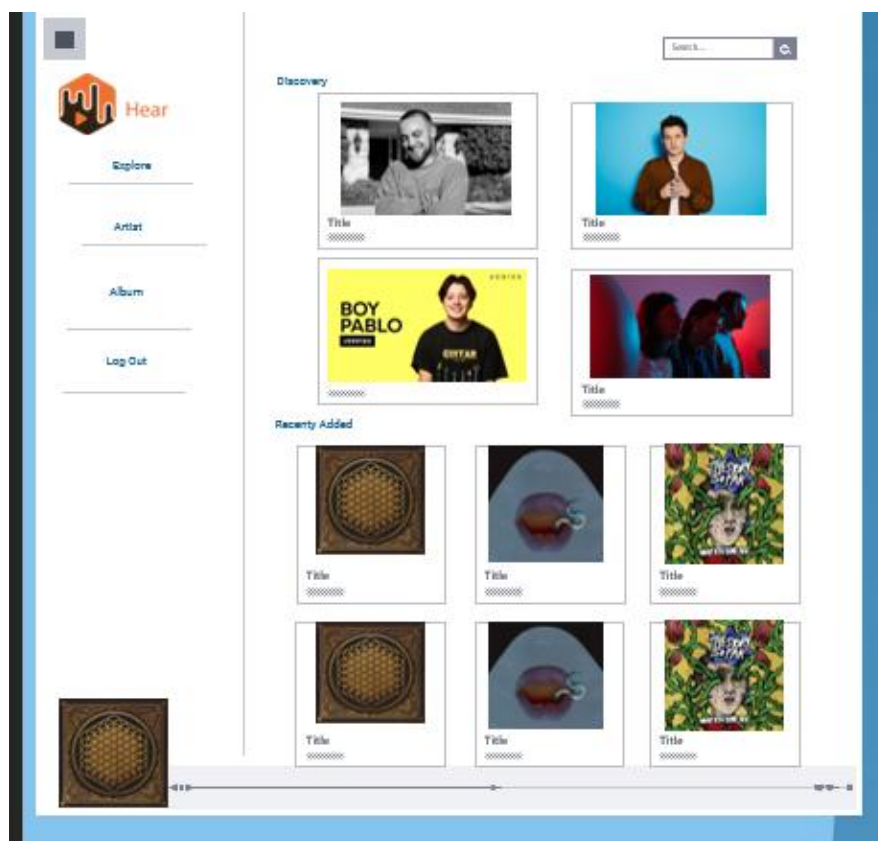
Confirm Password

Sign - Up

Sign-Up Wireframe

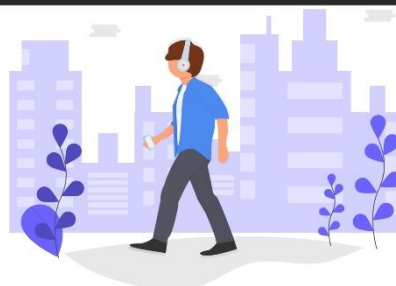
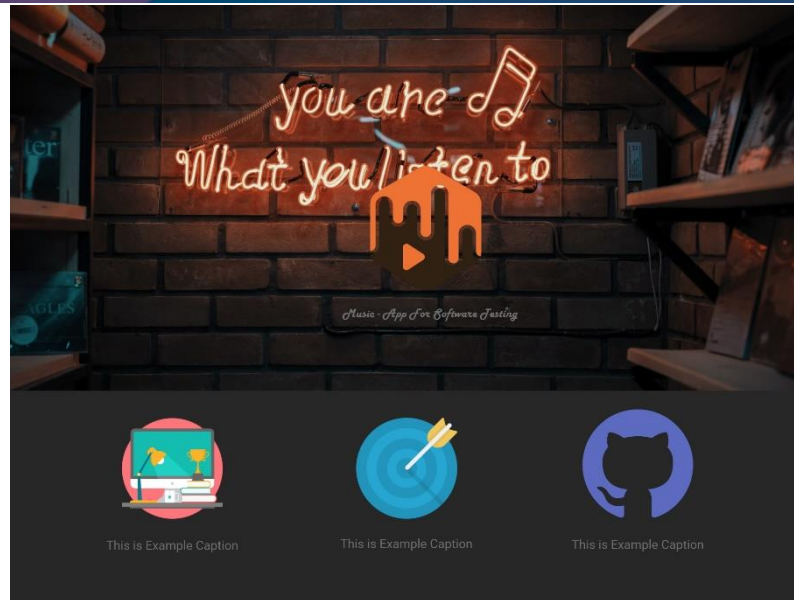
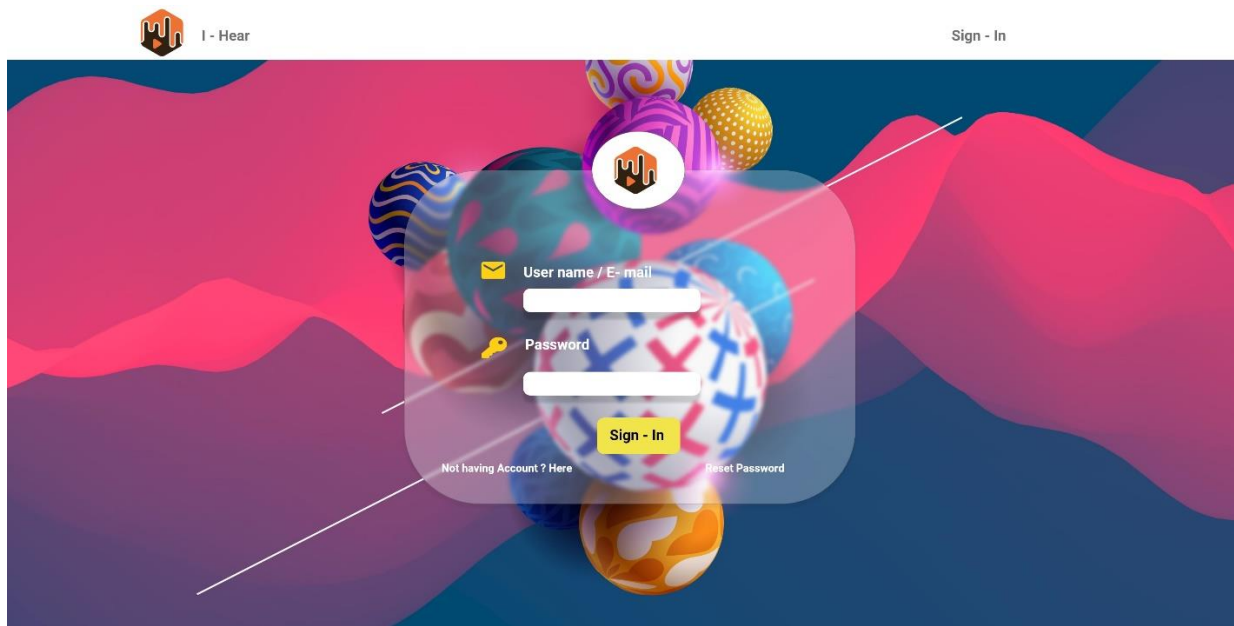


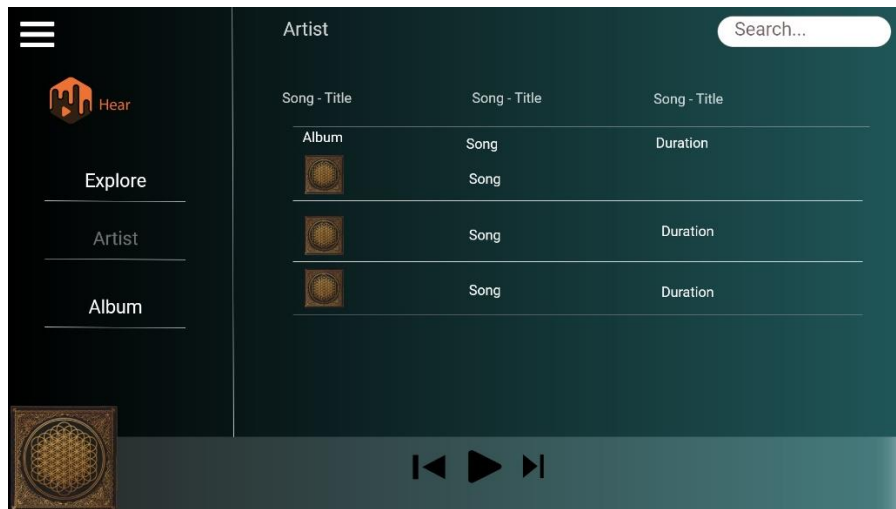
About-Page wireframe



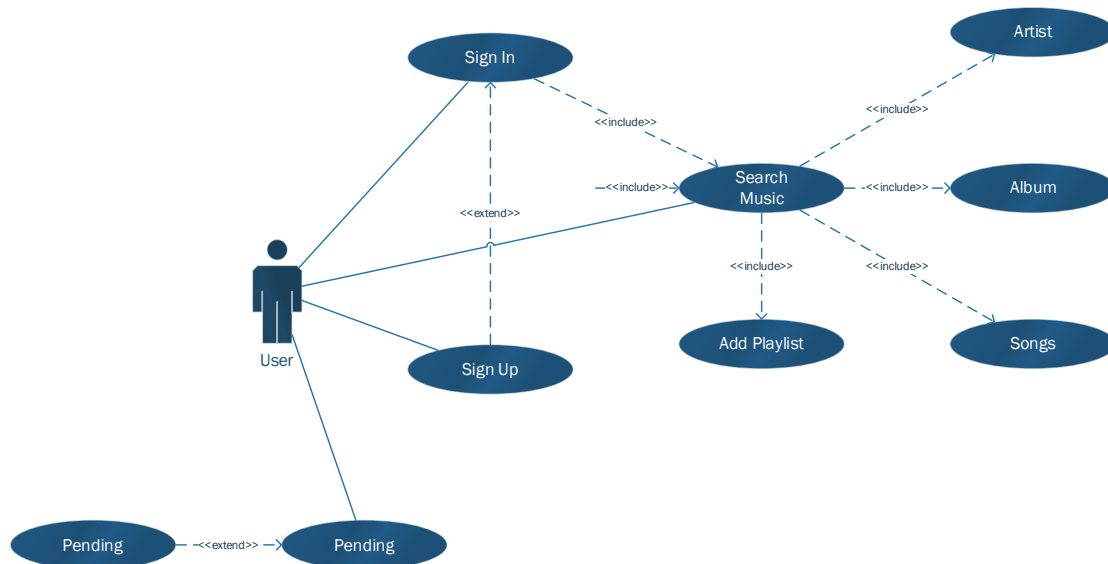
Index-page wireframe

4.4 Mock-Ups.





4.5 Use Case Diagram.



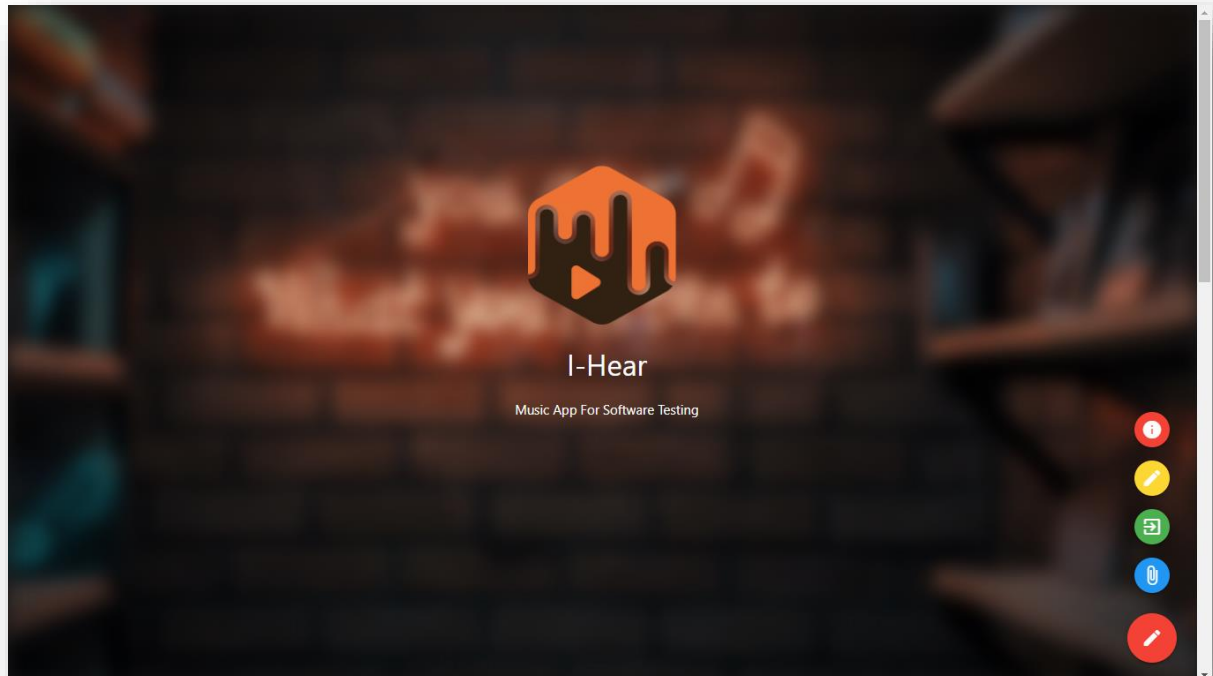
Use case diagram merupakan diagram yang menggambarkan hubungan antara aktor dengan sistem. Use case diagram bisa mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Use case diagram juga bisa digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan bisa juga mempresentasikan sebuah interaksi aktor dengan sistem. Komponen tersebut kemudian menjelaskan komunikasi antara aktor, dengan sistem yang ada. Dengan demikian, use case dapat dipresentasikan dengan urutan yang sederhana, dan akan mudah dipahami oleh para konsumen. Manfaat dari use case sendiri adalah untuk memudahkan komunikasi dengan menggunakan domain expert dan juga end user, memberikan kepastian pemahaman yang pas tentang requirement atau juga kebutuhan sebuah sistem.

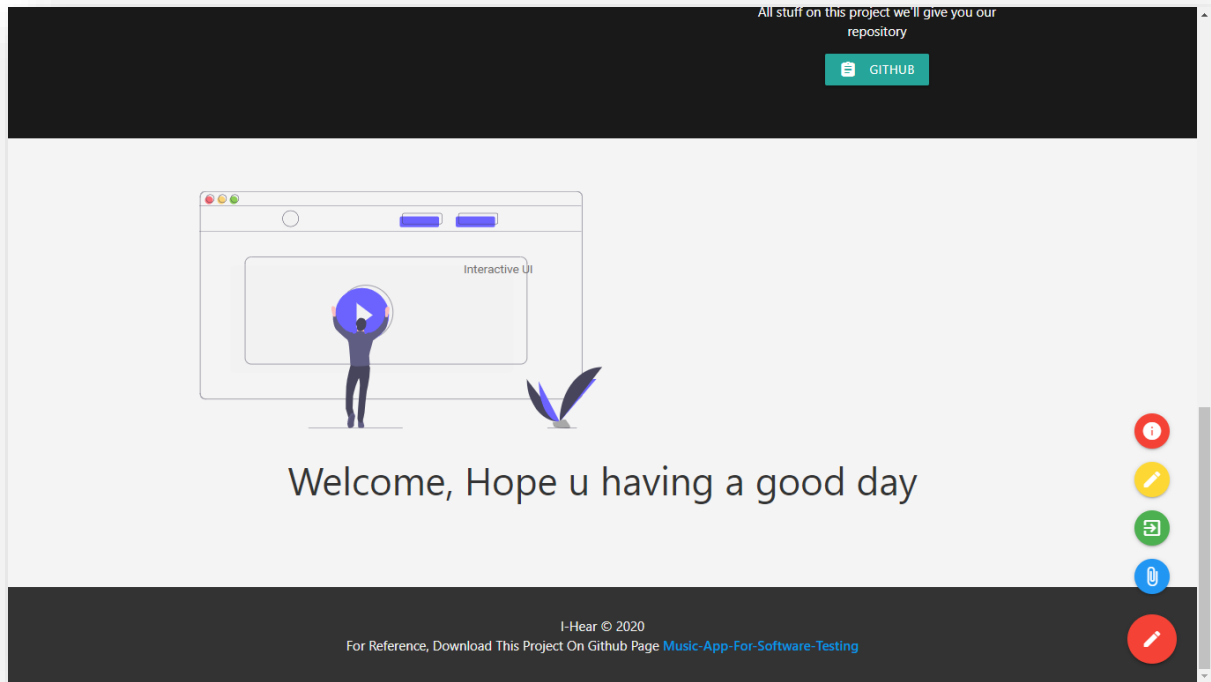
BAB V.

Pembahasan Project.

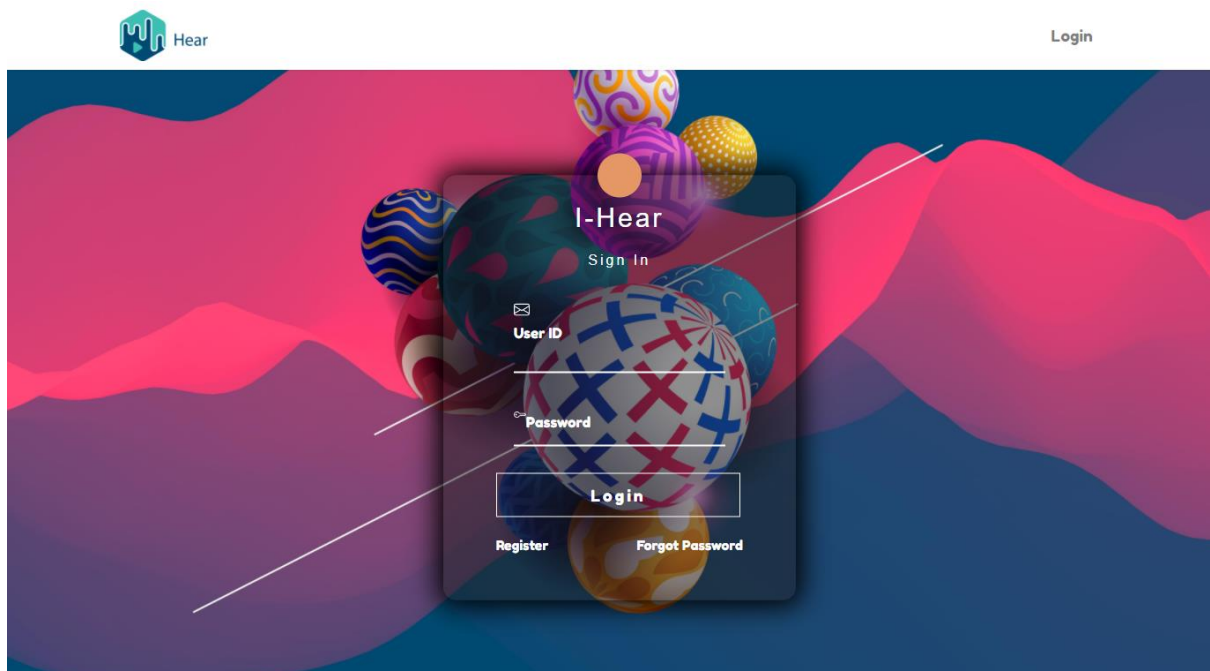
5.1 Fitur - fitur aplikasi.

❖ Verification email	Menggunakan Metode send mail ketika user membuat akun baru jika tidak memiliki akun tersedia
❖ Edit profile	User atau admin dapat mengupdate identitas profile nya dengan mengedit pada form yang sudah di sedia kan
❖ Model, View, Controller / MVC	Metode routing agar mensinkronkan url dan view template pada project
❖ Recently Added Playlist	Merupakan fitur penambahan playlist baru yang akan di tampilkan pada view index

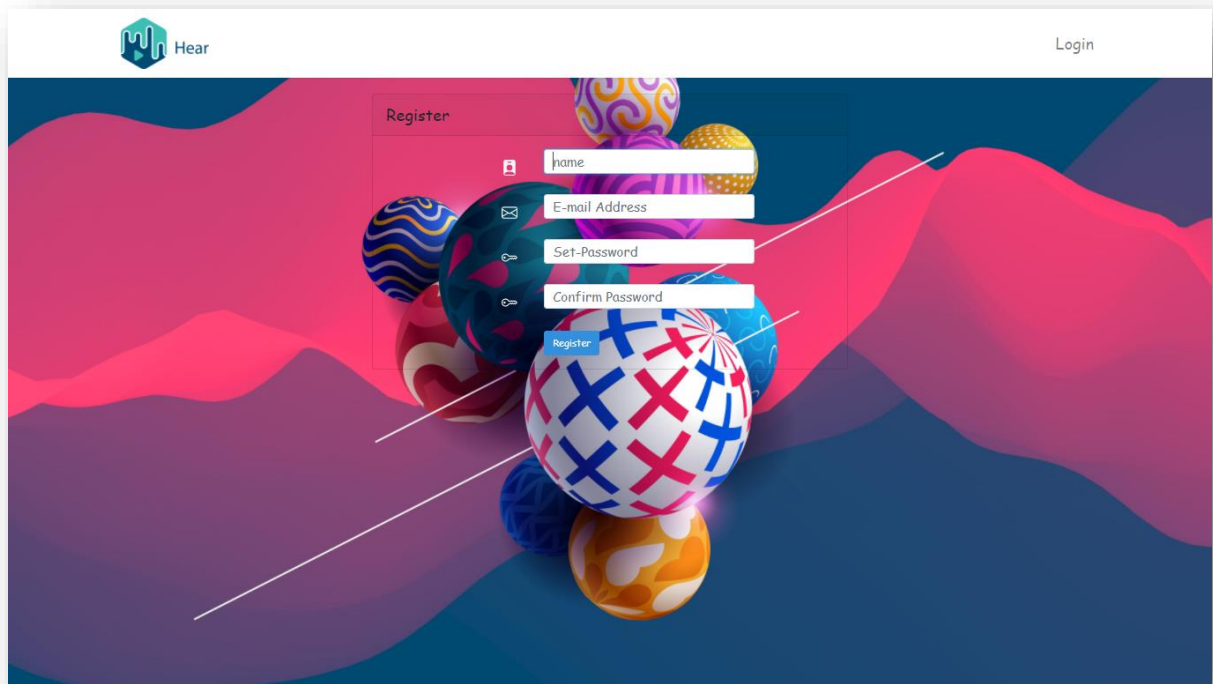




Gambar diatas merupakan tampilan welcome-page ketika user pertama kali mengakses aplikasi i - hear. Pada view welcome ini tidak terlalu banyak menggunakan deskripsi agar tidak terlalu rumit ketika dibaca oleh tester. Dibawah pojok kanan itu terdapat button berupa navigation circular yang terdiri 4 buah jenis dengan fungsi menuju link halaman yang akan dituju, misalkan urutan icon ke-1 (merah) ketika di klik menuju ke halaman 'tentang'. Pada footer telah di berikan sebuah link repository dari project ini apabila ingin mengembangkan lebih lanjut bisa mendownload folder dengan klik link berwarnabiru tersebut.



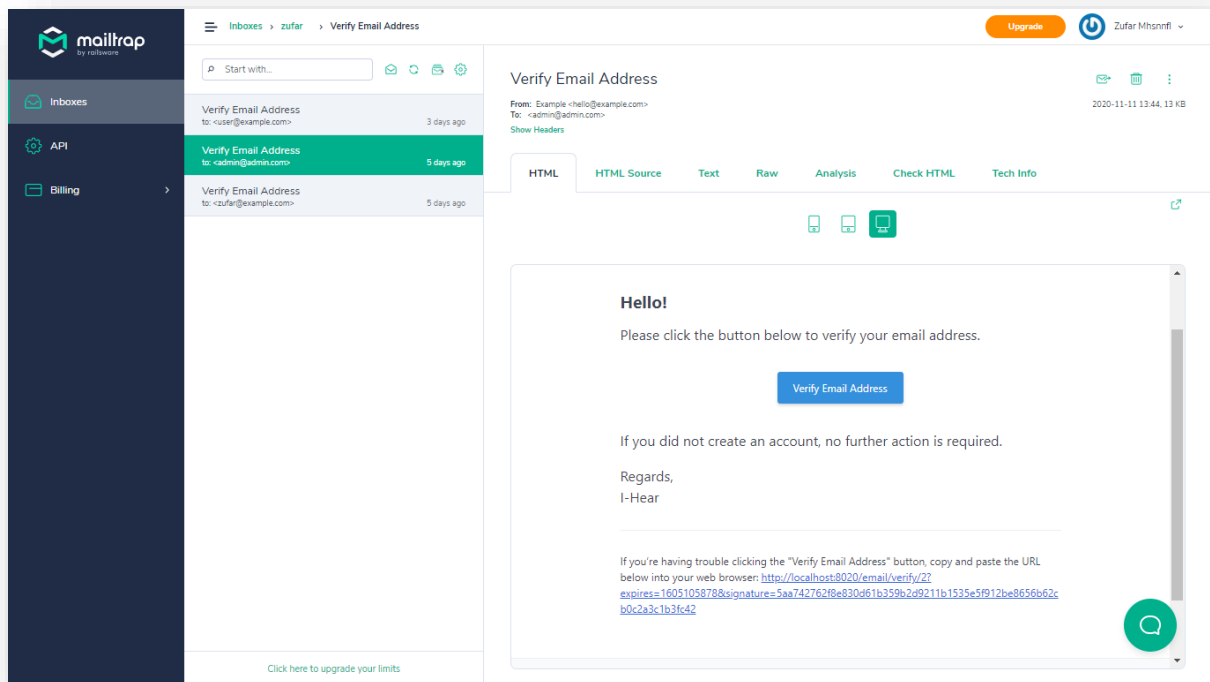
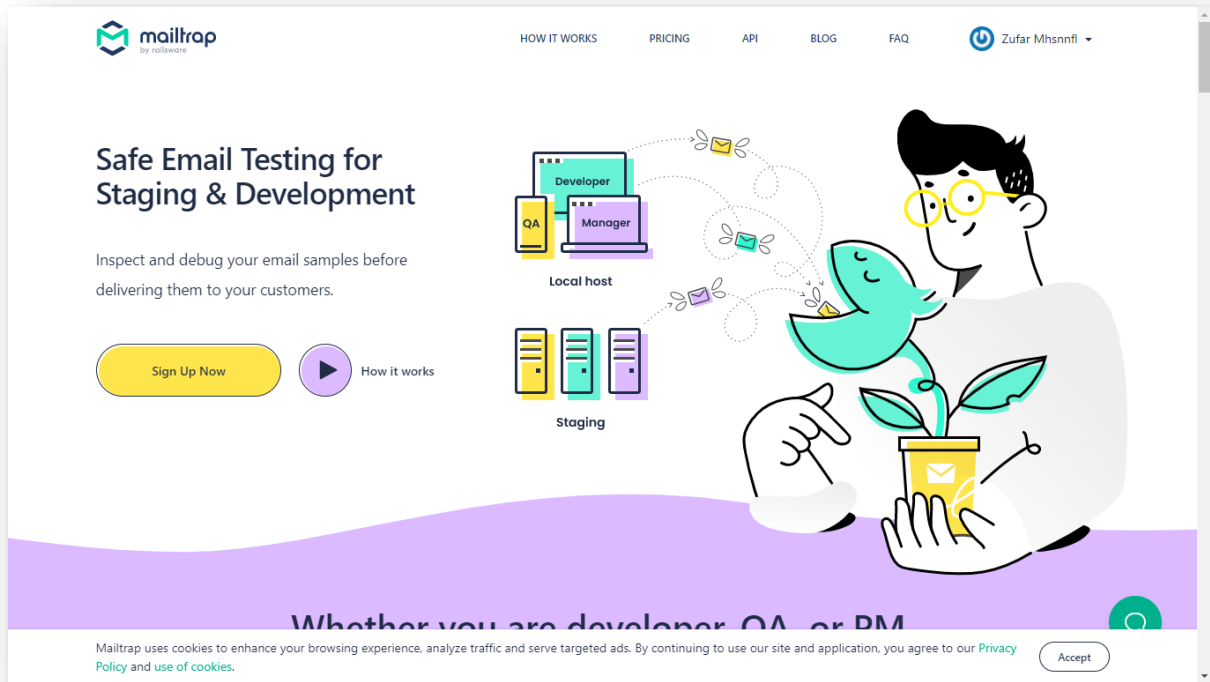
Tampilan ini merupakan view login / sign -in. Berdasarkan dari rancangan flowchart pada sistem user diminta untuk login terlebih dahulu (apabila memiliki akun tersedia) jika tidak daftar melalui form sign-up yang sudah tersedia pada form login. Pada laman sign-in sudah tervalidasi untuk setiap text input & button pada controller yang kemudian di migrate ke DBMS. Contoh validasi seperti ketika user menginputkan username & password berbeda sistem akan secara otomatis denied akun tersebut, dan juga pastikan input user & password harus sama setelah melalui proses me-registrasi akun.



The image shows a web application interface for a registration form. The background is a vibrant, abstract design with pink, blue, and purple wavy shapes, overlaid with several colorful Easter eggs. A semi-transparent white 'Register' form is centered on the screen. The form contains the following elements:

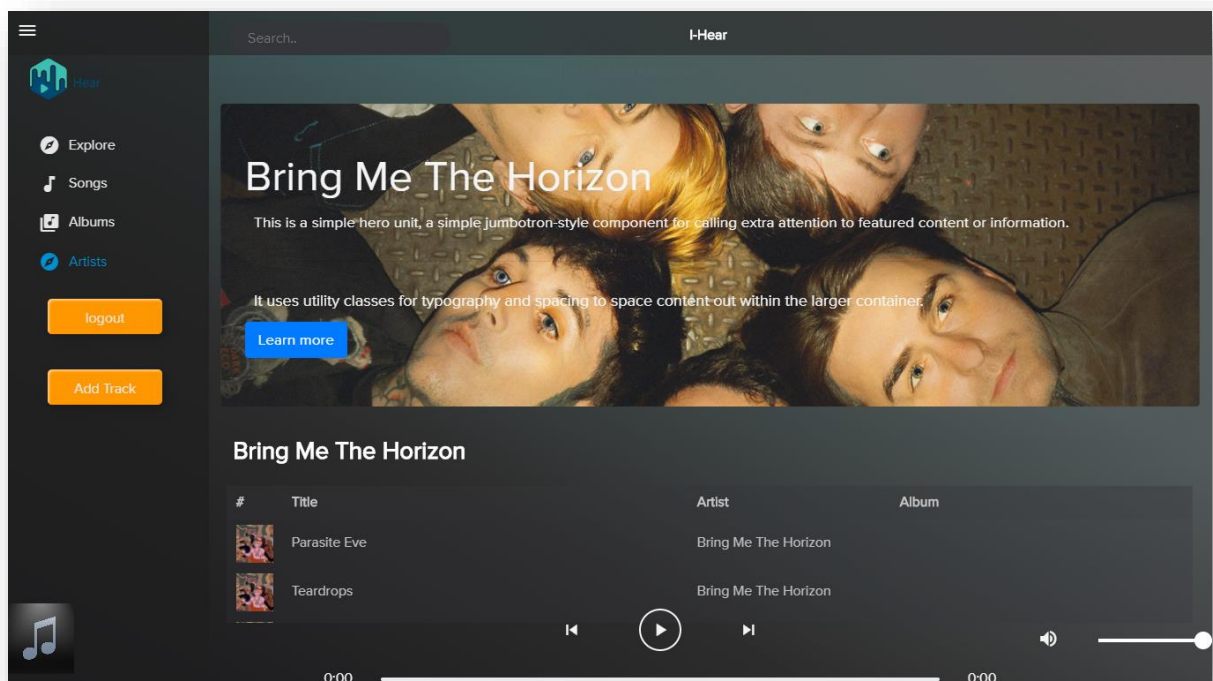
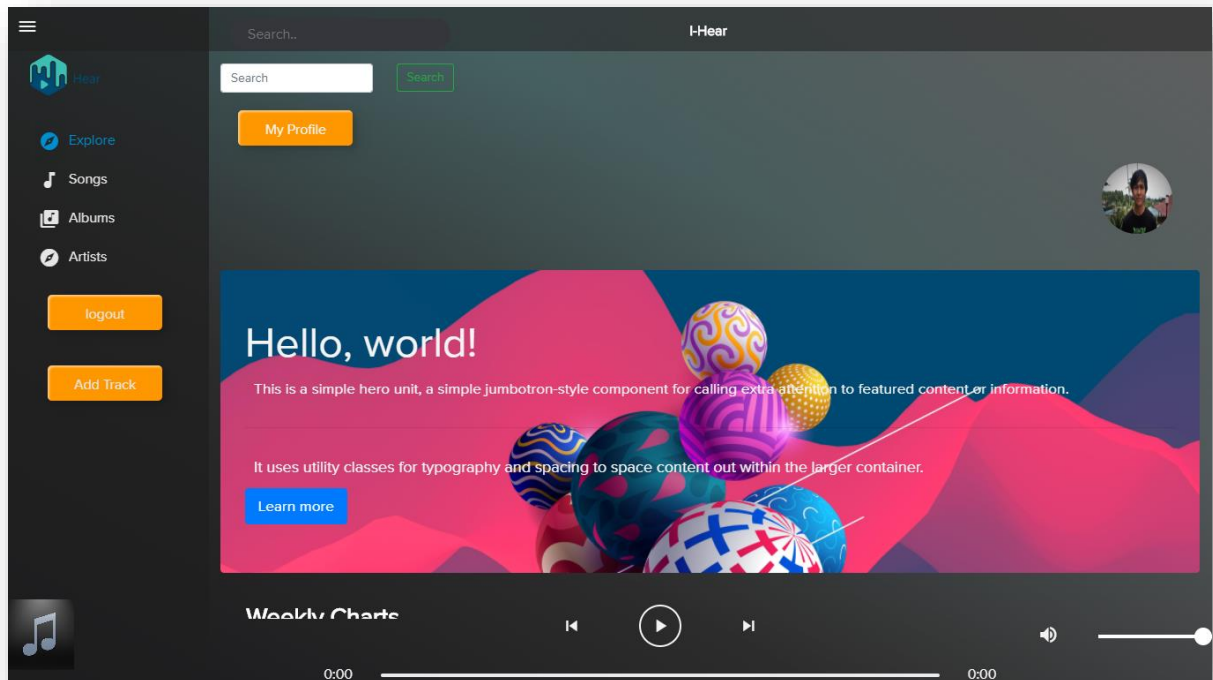
- Header:** The word 'Register' in a small, dark font.
- Input Fields:**
 - Name:** A text input field with a small person icon to its left.
 - E-mail Address:** A text input field with a small envelope icon to its left.
 - Set Password:** A text input field with a small key icon to its left.
 - Confirm Password:** A text input field with a small key icon to its left.
- Button:** A blue button with the text 'Register' in white, located below the password fields.

Jika user tidak memiliki akun tersedia, pada laman sign-up ini sudah disediakan beberapa text input untuk membuat akun baru diantara nya username, email, password dan confirm password. Setiap akun yang telah dibuatkan memiliki id token nya tersendiri.

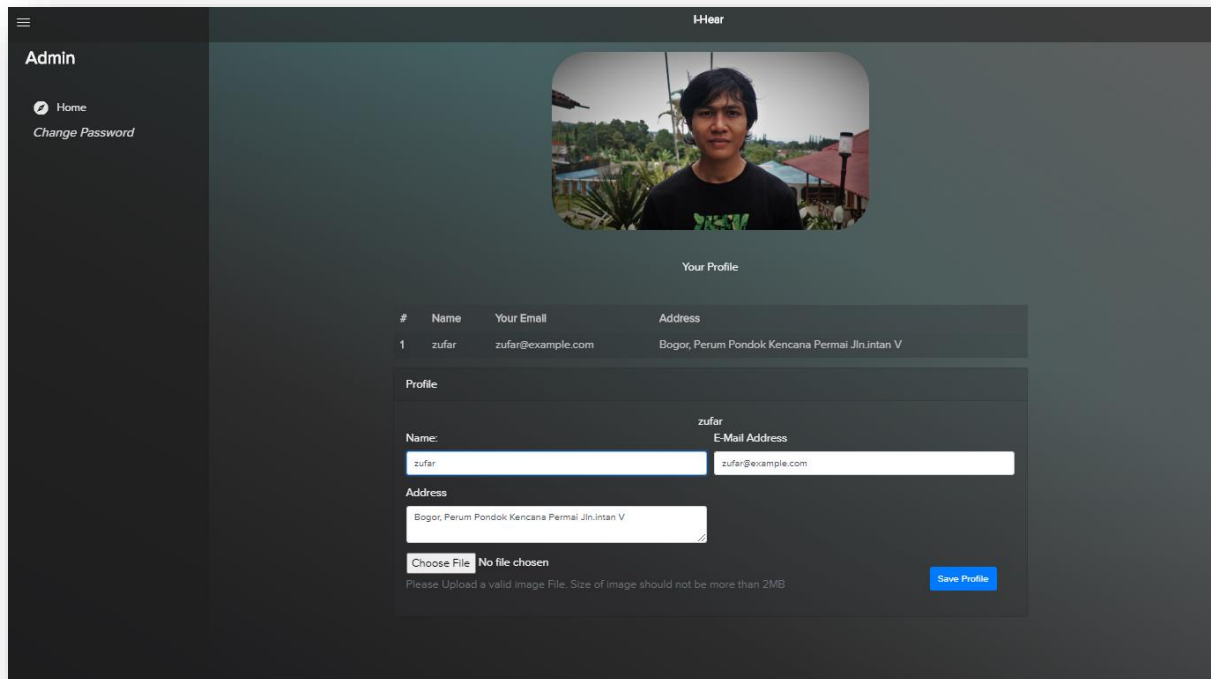


Karena aplikasi ini menggunakan metode SMTP untuk (E-mail Verification) disini akan menjelaskan sedikit mengenai tools smtp MAILTRAP.IO jadi mailtrap.io ini merupakan fake smtp server yang biasa digunakan programmer untuk mengetes suatu pengiriman email tanpa harus menggunakan akun real user. Dari laman diatas setelah proses registrasi akun user akan di minta me verifikasi akun via

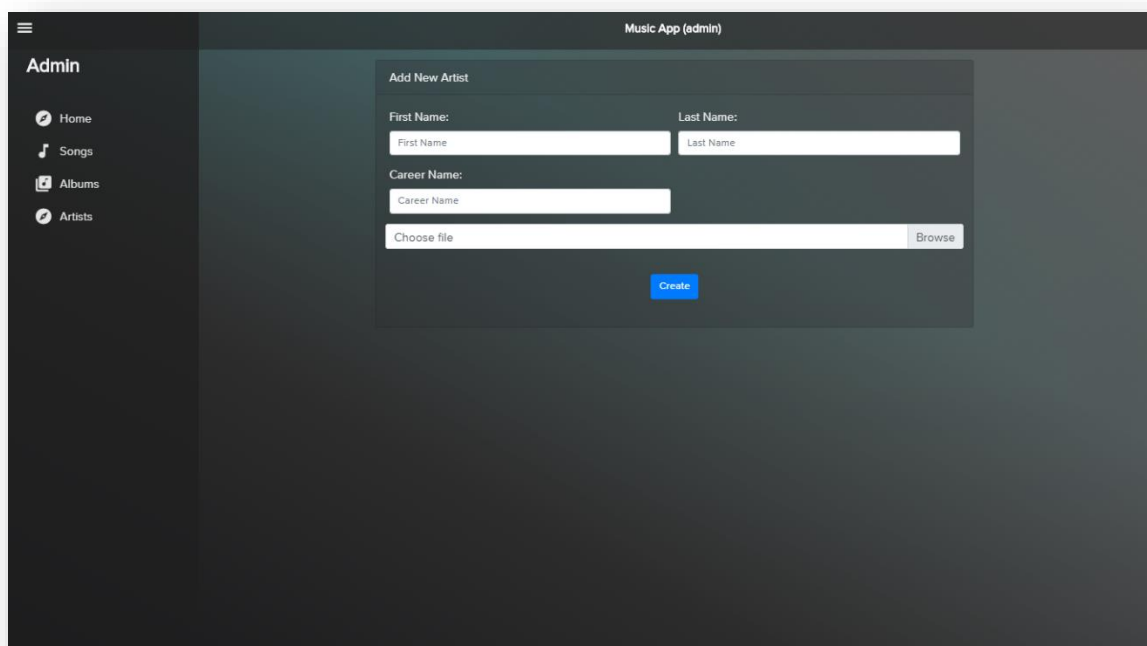
aplikasi maltrap.io, contoh seperti gambar diatas kami meverifikasi akun admin untuk memverifikasi data nya agar valid.



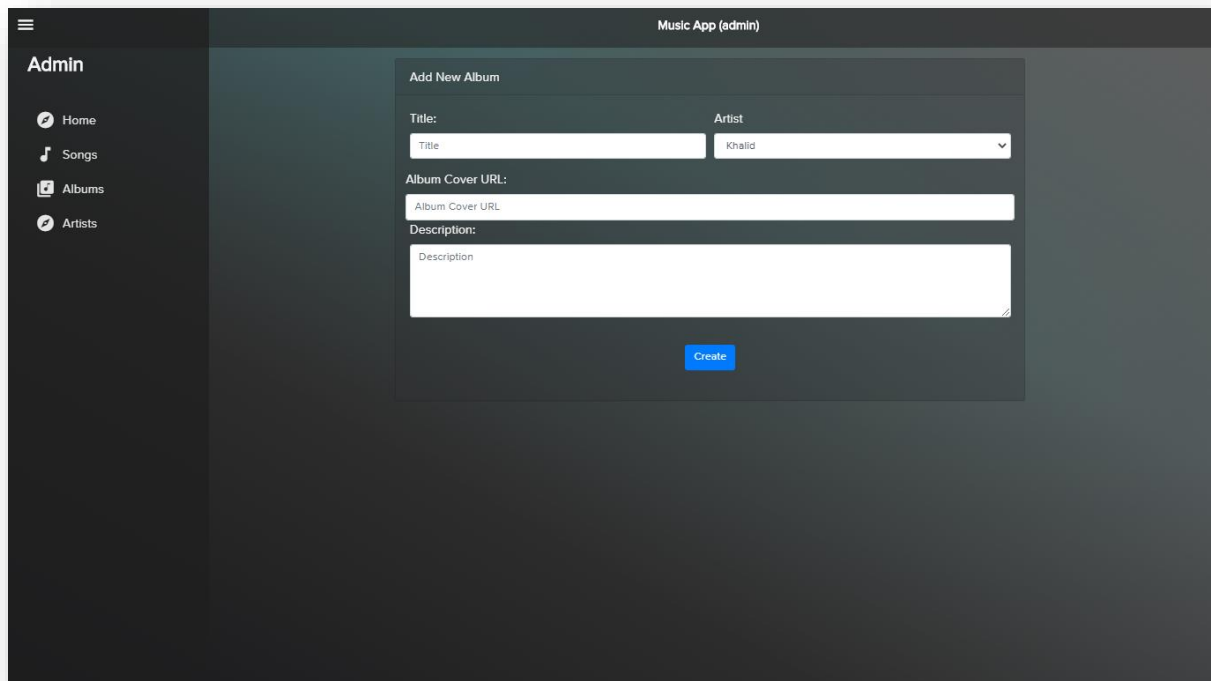
Proses verifikasi telah berhasil, setelah itu user akan di redirect ke halaman utama program (/page/index). Dari laman index kami menyajikan avatar sebagai identitas user nya.



Setiap aplikasi terutama aplikasi streaming seperti halnya amazon music, spotify, i-tunes dsb pasti memiliki fitur edit profile di dalamnya sistemnya, sama seperti aplikasi i-hear disini kami menyajikan juga untuk update profile user. Gambar diatas merupakan view edit profile dimana di dalamnya berupa table dengan kami @foreach setiap attribute nya, sedangkan dibawah nya kami tambahkan form edit dan beserta button form action untuk mengubah picture avatar nya.



Gambar diatas merupakan view untuk menambahkan artist.



The image shows a web application interface for a music app, specifically the 'Admin' section. The header bar is dark grey with a hamburger menu icon on the left and the text 'Music App (admin)' on the right. A dark sidebar on the left contains the word 'Admin' at the top, followed by four menu items: 'Home' (with a house icon), 'Songs' (with a musical note icon), 'Albums' (with a CD icon), and 'Artists' (with a person icon). The main content area is a light grey background. In the center, there is a white modal box titled 'Add New Album'. Inside this modal, there are four input fields: 'Title' (a text input), 'Artist' (a dropdown menu with 'Khalid' selected), 'Album Cover URL' (a text input), and 'Description' (a larger text area). A blue 'Create' button is located at the bottom right of the modal.

Music App (admin)

Admin

- Home
- Songs
- Albums
- Artists

Add New Album

Title:

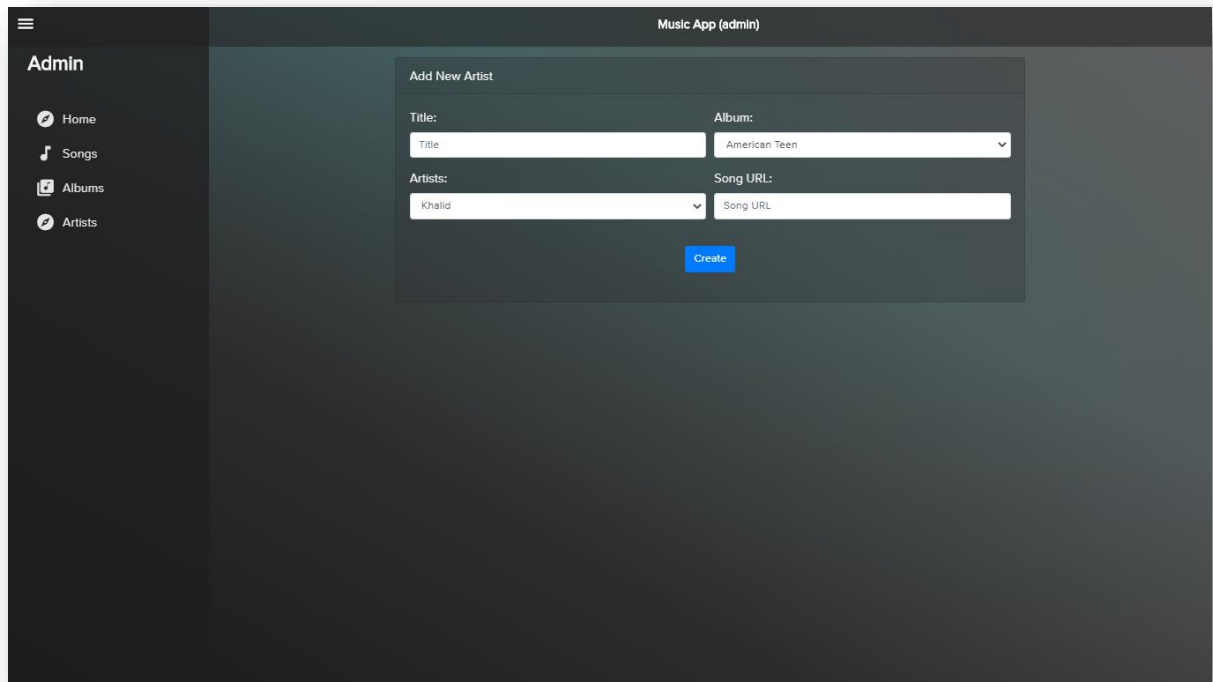
Artist:

Album Cover URL:

Description:

Create

View add album untuk menambahkan album track.



Ini adalah view add song untuk menambahkan soundtrack pada view index (halaman utama) program

BAB VI.

Kesimpulan & Saran.

Kesimpulan

Pembuatan aplikasi music I HEAR berbasis website bertujuan untuk memenuhi tugas Software Testing serta untuk memberikan fasilitas media streaming bagi para pencinta musik. Secara garis besar, berdasarkan hasil perancangan dan pembuatan aplikasi Music Application (I Hear) yang telah dilakukan, dapat disimpulkan hal-hal sebagai berikut:

1. Dengan menggunakan aplikasi ini, pengguna dapat mendengarkan music dari berbagai genre music secara gratis melalui website
2. Aplikasi ini memberikan kemudahan bagi para pengguna, dimana mereka dapat mengupdate profile dan dapat menambahkan playlistnya sendiri

Saran

Untuk itu penulis mencoba memberikan saran yang dapat dijadikan bahan pertimbangan untuk penelitian selanjutnya, yaitu diantaranya:

1. Pada saat pengembangan aplikasi selanjutnya, aplikasi ini dapat digunakan dalam seluruh sistem operasi android serta dapat digunakan secara offline.
2. Pada saat pengembangan aplikasi selanjutnya, dibangun notifikasi bagi pengguna apabila ada lagu/album terbaru dalam platform agar pengguna lebih update terhadap lagu-lagu terbaru.

Dalam pembuatan ini perlu waktu yang cukup dalam mengembangkan, semoga aplikasi ini bisa dikembangkan lagi agar aplikasi bisa terus update dan menambahkan fitur baru.

Daftar Pustaka.

<https://laravel.com/docs/5.8/releases>

<https://getbootstrap.com/docs/4.5/about/overview/>

<https://materializecss.com/>

https://github.com/ZufarNaufal/music_app