

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC BÁCH KHOA



## ĐỒ ÁN TỔNG HỢP HƯỚNG CÔNG NGHỆ PHẦN MỀM CO3103

---

# KASHI SAGASHI

---

Lớp:	CN01	
GVHD:	Bùi Công Tuấn	
Khoa:	Khoa học và Kỹ thuật máy tính	
SV thực hiện:	Bùi Hồ Hải Đăng	2153289
	Lý Công Đăng	2152056
	Nguyễn Trang Sỹ Lâm	2152715
	Nguyễn Ngọc Hoàng Lâm	2152714
	Đặng Thành Anh	2153153

TP. Hồ Chí Minh, Tháng 9/2023

# Table of content

<b>1</b>	<b>Project Description</b>	<b>2</b>
<b>2</b>	<b>Motivation</b>	<b>2</b>
<b>3</b>	<b>Objectives</b>	<b>2</b>
<b>4</b>	<b>Functional reqs</b>	<b>2</b>
4.1	Search for song name . . . . .	2
4.2	Search for lyrics . . . . .	2
<b>5</b>	<b>Non-functional reqs</b>	<b>3</b>
5.1	Accessibility . . . . .	3
5.2	Access Security and Safety . . . . .	3
5.3	Scalability . . . . .	3
5.4	Integrity . . . . .	3
5.5	Availability . . . . .	3
<b>6</b>	<b>List of Mockup</b>	<b>4</b>
6.1	Home page . . . . .	4
6.2	Search Bar . . . . .	4
6.3	Sign up page . . . . .	5
6.4	Sign in page . . . . .	7
6.5	Admin account . . . . .	9
6.6	Submit song page . . . . .	10
6.7	Verify song page . . . . .	11
6.8	Normal account . . . . .	11
<b>7</b>	<b>Screen flow</b>	<b>12</b>
<b>8</b>	<b>Use-case diagram</b>	<b>12</b>
<b>9</b>	<b>Use-case scenarios</b>	<b>12</b>
<b>10</b>	<b>Key Technical decisions</b>	<b>13</b>
10.1	BM25 . . . . .	13
10.1.1	Giới thiệu TF-IDF . . . . .	13
10.1.2	Term frequency . . . . .	13
10.1.3	Inverse document frequency . . . . .	14
10.1.4	TF - IDF . . . . .	14
10.2	Tại sao là BM25? . . . . .	14
<b>11</b>	<b>Tại sao lại chọn làm web?</b>	<b>15</b>
<b>12</b>	<b>Sử dụng Next.js để xây dựng ứng dụng</b>	<b>15</b>
<b>13</b>	<b>Sử dụng Express.js để xây dựng ứng dụng</b>	<b>16</b>

# 1 Project Description

Dự án của chúng tôi là một nền tảng tìm kiếm nhạc kỹ thuật số cung cấp một danh mục lớn các bài hát, album và danh sách phát từ nhiều nghệ sĩ và thể loại khác nhau. Cung cấp quyền truy cập vào nhiều bài hát, giúp người dùng có thể khám phá, truyền phát và nghe nhạc từ nhiều nghệ sĩ, cả chính thống và độc lập. Người dùng có thể quản lý danh sách phát được cá nhân hóa. Chức năng tìm kiếm giúp bạn dễ dàng tìm thấy các bài hát, album hoặc nghệ sĩ cụ thể và cung cấp danh sách phát cũng như đề xuất âm nhạc dựa trên sở thích của người dùng. Đặc biệt người dùng có thể tìm kiếm bài hát theo lời bài hát, đây là chức năng rất hữu ích cho những người không biết tên bài hát mình vô tình nghe.

## 2 Motivation

Động lực đằng sau việc tạo ra dự án này là cung cấp một cách hợp pháp, thuận tiện và thú vị để mọi người truy cập và thưởng thức âm nhạc. Hình dung một nền tảng nơi người dùng có thể tìm và phát trực tuyến ngay lập tức bất kỳ bài hát nào họ muốn mà không cần phải mua từng bản nhạc hoặc album riêng lẻ. Ngoài ra, những người vô tình nghe được bài hát ở bất cứ đâu cũng có thể dễ dàng tìm ra tên bài hát đó chỉ bằng cách biết lời bài hát.

## 3 Objectives

Mục tiêu của chúng tôi là tạo ra một ứng dụng web cung cấp chức năng giúp người dùng tìm bài hát theo lời bài hát. Hơn nữa, chúng tôi muốn nâng cao trải nghiệm khám phá âm nhạc cho người dùng. Chúng tôi đã kết hợp các tính năng như danh sách phát được cá nhân hóa và thuật toán đề xuất để giúp người dùng khám phá âm nhạc mới dựa trên thói quen nghe của họ. Người dùng cũng có thể tạo hồ sơ của riêng mình, theo dõi những người dùng khác và chia sẻ danh sách xếp hạng. Khía cạnh xã hội này cho phép khám phá âm nhạc thông qua bạn bè.

## 4 Functional reqs

### 4.1 Search for song name

Người dùng tương tác với trang web bằng cách nhập một đoạn lời bài hát nhỏ, sau đó trang web sẽ cung cấp danh sách các bài hát có lời bài hát tương tự hoặc lời bài hát có chứa lời bài hát được nhập vào đoạn bài hát đó.

### 4.2 Search for lyrics

Người dùng tương tác với trang web bằng cách nhập một đoạn lời bài hát nhỏ hoặc bất kỳ tên bài hát nào, sau đó trang web sẽ cung cấp danh sách các bài hát có chủ đề tương tự lời bài

hát hoặc tên bài hát được nhập vào.

## 5 Non-functional reqs

### 5.1 Accessibility

Các bài hát được giới thiệu trên web đều có đầy đủ link Spotify và link YouTube, hình ảnh minh họa cho bài hát và lời bài hát đầy đủ.

### 5.2 Access Security and Safety

Khi người dùng quên mật khẩu, trang web sẽ gửi cho họ một email có liên kết để đặt lại mật khẩu.

### 5.3 Scalability

Với hệ thống ban đầu, nó có thể xử lý 500 người dùng tương tác với website và theo thời gian, con số này sẽ được mở rộng để tăng lên 10.000 người dùng. Hệ thống sẽ nỗ lực đo lường mức độ tương tác của người dùng với trang web.

### 5.4 Integrity

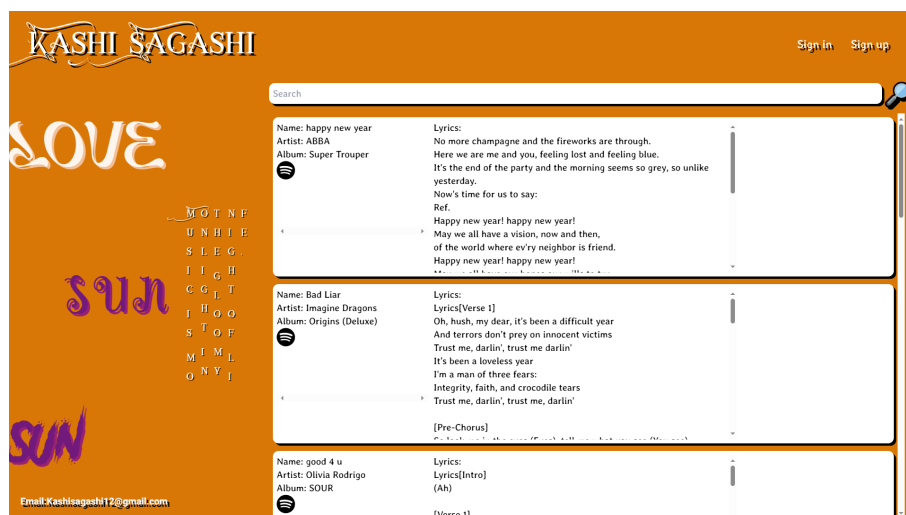
Sau khi đăng nhập, sau vài ngày người dùng không cần đăng nhập lại mà vẫn có thể sử dụng dịch vụ.

### 5.5 Availability

Người dùng có thể tương tác với website 24/7 trừ thời gian bảo trì sẽ được thông báo trước cho người dùng qua email và trên website.

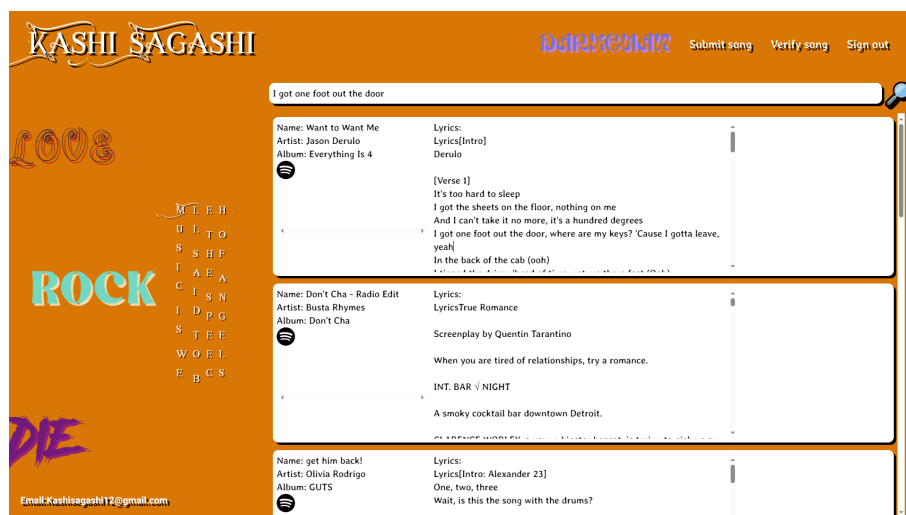
## 6 List of Mockup

### 6.1 Home page



Khi vào trang chủ của trang web, ta sẽ có giao diện như hình trên. Với thanh tìm kiếm chính giữa trang và các nút sign up, sign in nằm góc trên bên phải.

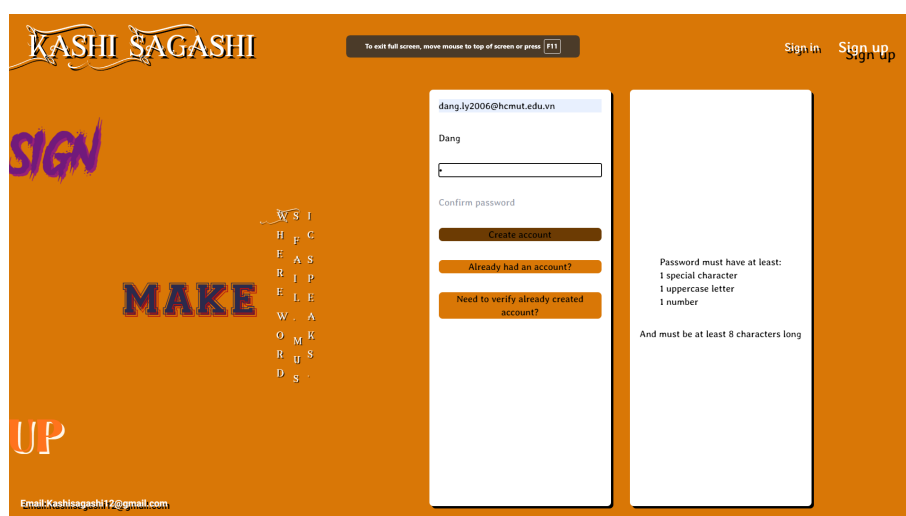
### 6.2 Search Bar



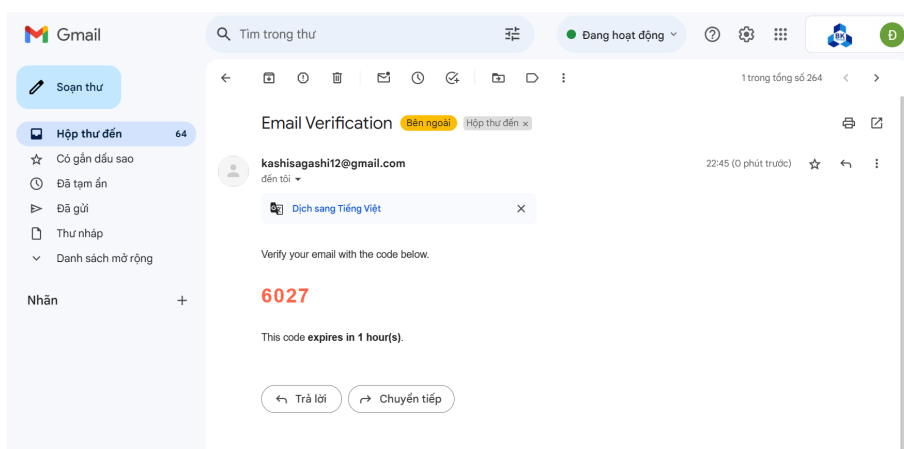


Với bất kỳ người dùng nào, dù có tài khoản hay không có tài khoản đều dùng được tính năng này. Cho phép người dùng có thể tìm kiếm bài hát theo tên hoặc theo lời. Khi có bài hát phù hợp, sẽ hiển thị các thông tin về bài hát ví dụ như lời bài hát, ca sĩ, ngày phát hành và kèm với link dẫn đến Spotify để người dùng có thể nghe.

### 6.3 Sign up page



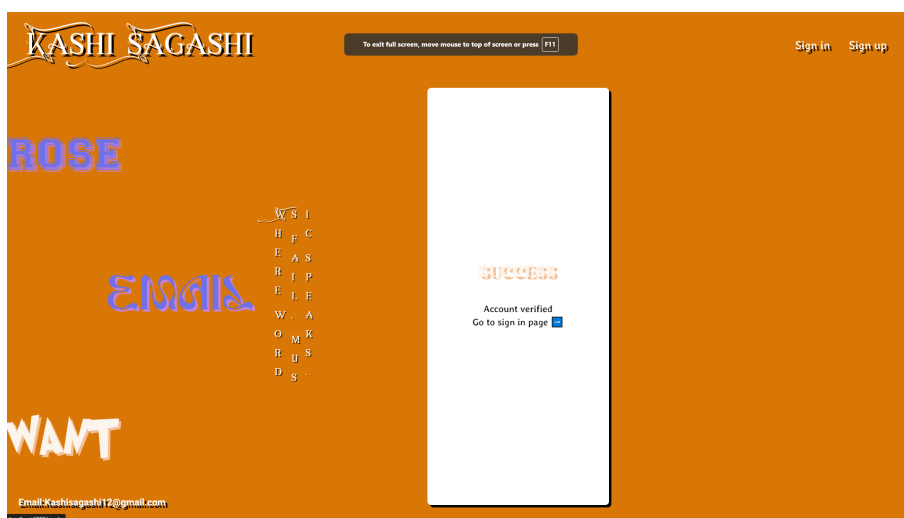
Khi người dùng đăng ký tài khoản cần làm theo các yêu cầu của web đưa ra. Sau đó, sẽ nhận được mã code qua email đã nhập.



Người dùng cần nhập code vào khung OTP để hoàn tất tạo tài khoản.

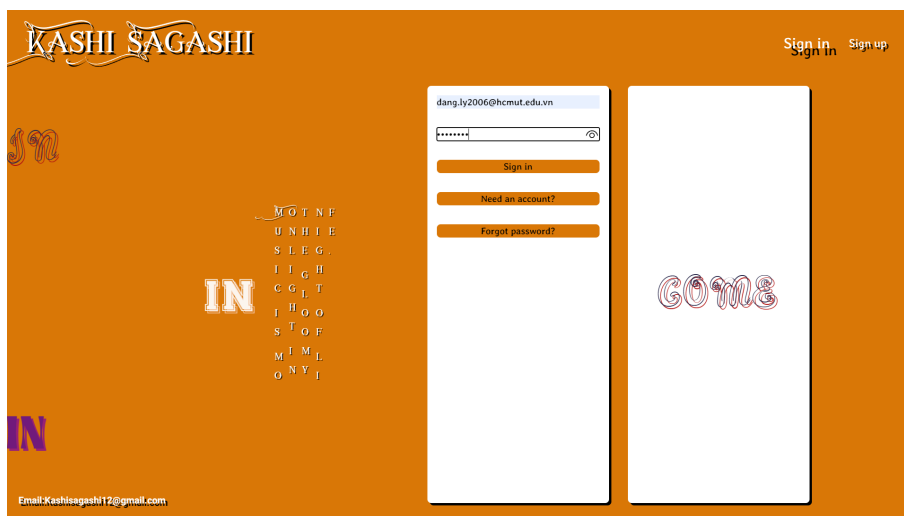


Khi hoàn tất web sẽ hiển thị thông báo như hình dưới đây.



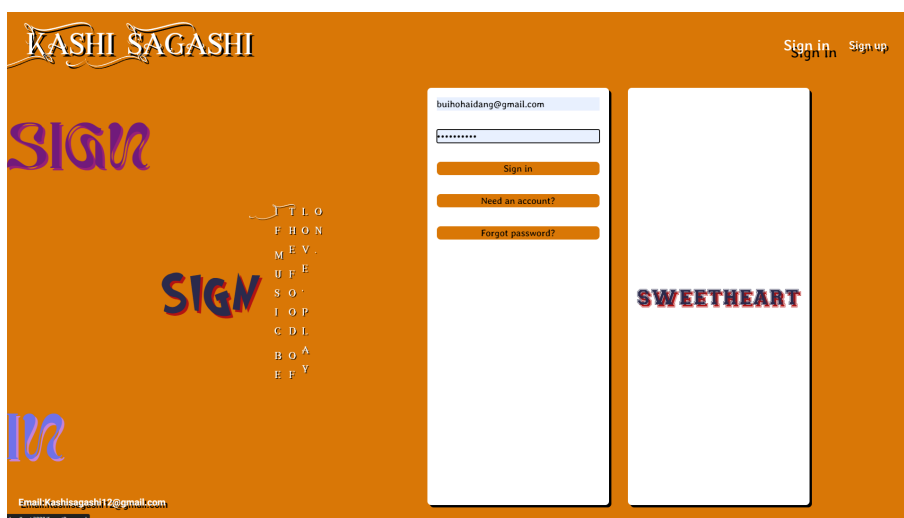
## 6.4 Sign in page

Nếu người dùng nhập thông tin tài khoản chính xác thì web sẽ hiển thị thông báo đăng nhập thành công.

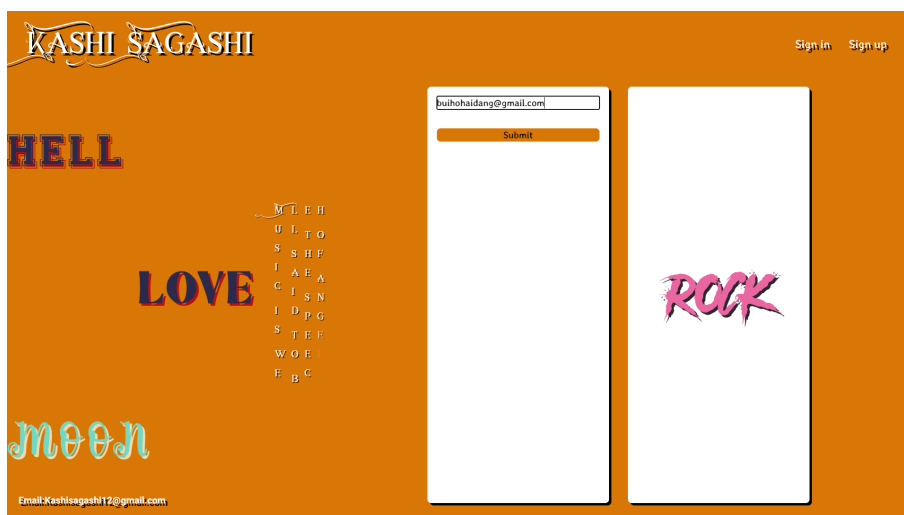


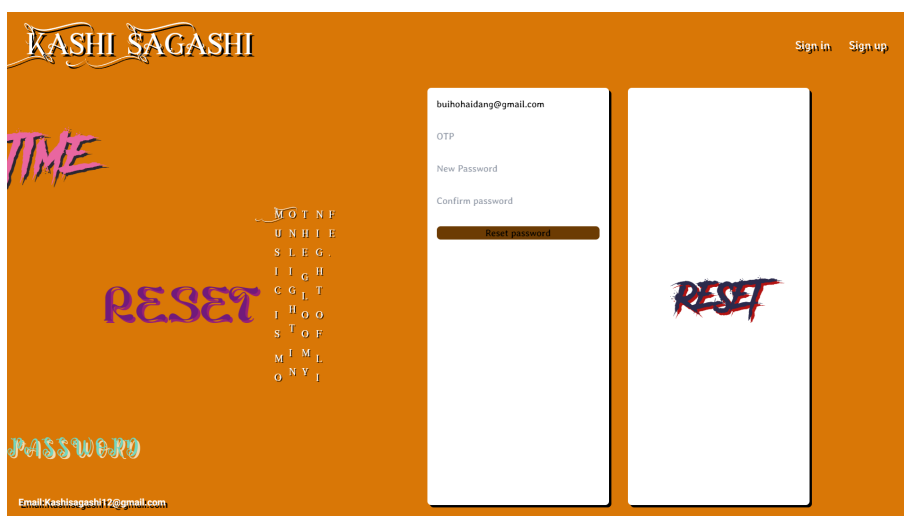
Trong giao diện đăng nhập, người dùng khi quên mật khẩu hoặc chưa có tài khoản có thể dễ dàng giải quyết điều này. Sau khi người dùng ấn vô quên mật khẩu thì màn hình sẽ xuất hiện giao diện như dưới đây.





Người dùng cần làm theo chính xác những gì mà trang web yêu cầu để có thể đổi lại mật khẩu tài khoản của mình.





KASHI SAGASHI

Sign in Sign up

TIME

RESET

PASSWORD

Email: KashiSagashi12@gmail.com

buihohaidang@gmail.com

OTP

New Password

Confirm password

Reset password

Sau khi đổi hoàn tất sẽ xuất hiện thông báo đổi mật khẩu thành công.



KASHI SAGASHI

Sign in Sign up

TIME

RESET

PASSWORD

Email: KashiSagashi12@gmail.com

SUCCESS

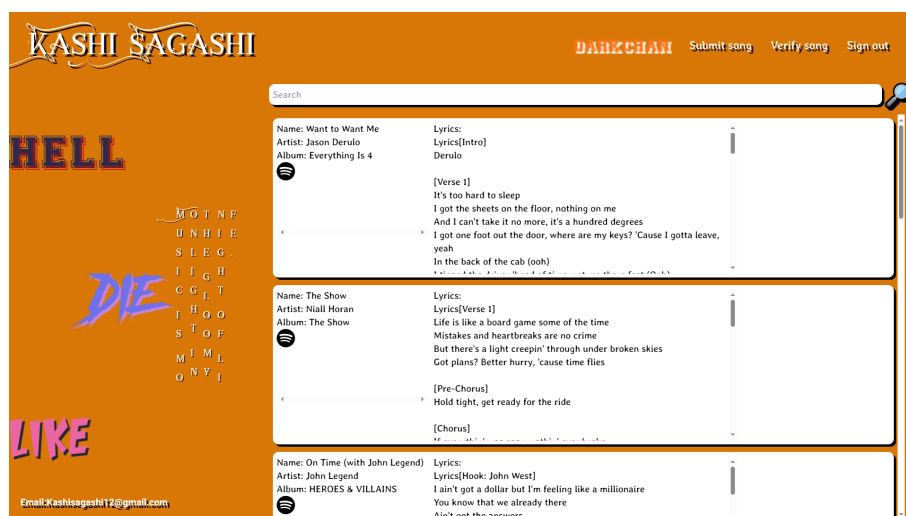
Password reset

Go to sign in page

Ngoài ra có 2 loại tài khoản là tài khoản với quyền admin và tài khoản thông thường.

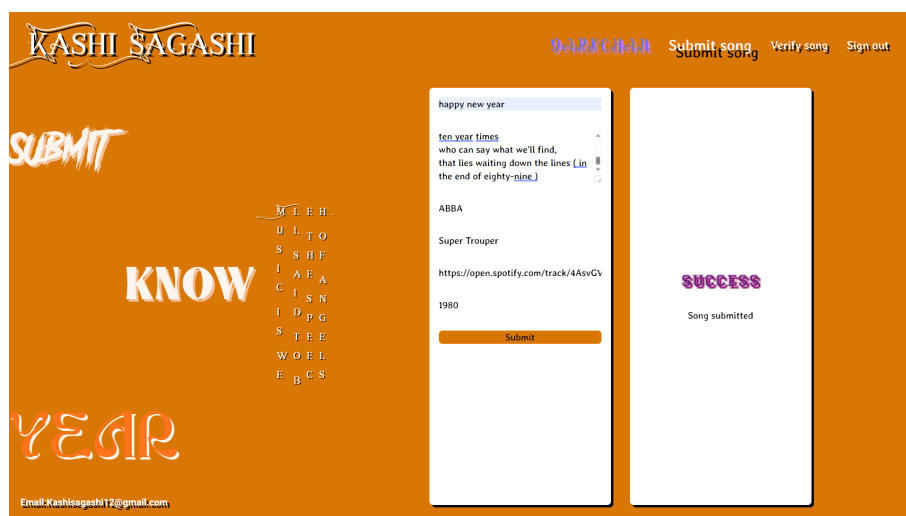
## 6.5 Admin account

Tài khoản với quyền Admin sẽ có các nút sau:



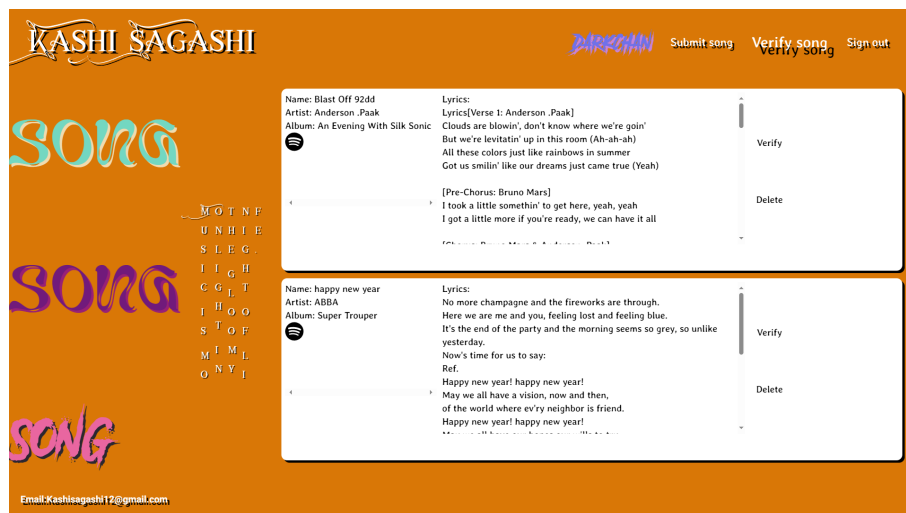
## 6.6 Submit song page

Trong giao diện này, người dùng có thể cung cấp thông tin về bài hát mình sẽ thêm vào ví dụ như tên, lời bài hát, ca sĩ hát bài này,... Trong đây, nhóm em sẽ lấy ví dụ về bài hát Happy New Year của nhóm ABBA.

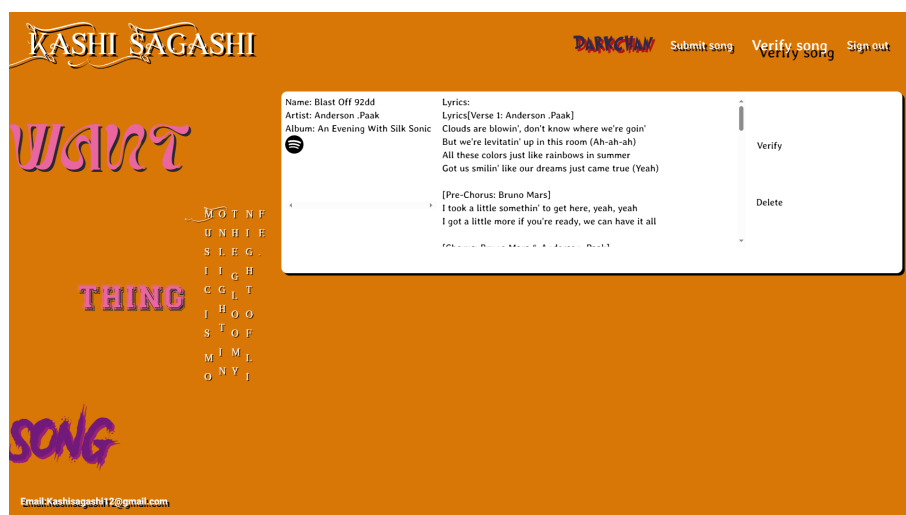


Sau khi nhập đầy đủ thông tin cho bài hát thì khi nhấn phím Submit, trang web sẽ thông báo là submit bài hát thành công.

## 6.7 Verify song page



Đây là tính năng mà chỉ người dùng với quyền Admin mới có, nó cho phép chúng ta duyệt các bài hát mà những người dùng khác muốn thêm vào. Khi bấm vào, sẽ hiển thị 1 danh sách các bài hát cần được phê duyệt. Bấm vào bài hát sẽ hiển thị đầy đủ thông tin về bài hát này, ta cần kiểm duyệt thông tin và nếu đúng thì cho phép đăng lên. Cũng trong giao diện này, người dùng với quyền Admin có thể xóa đi các bài hát với thông tin không chính xác mà đang chờ phê duyệt.

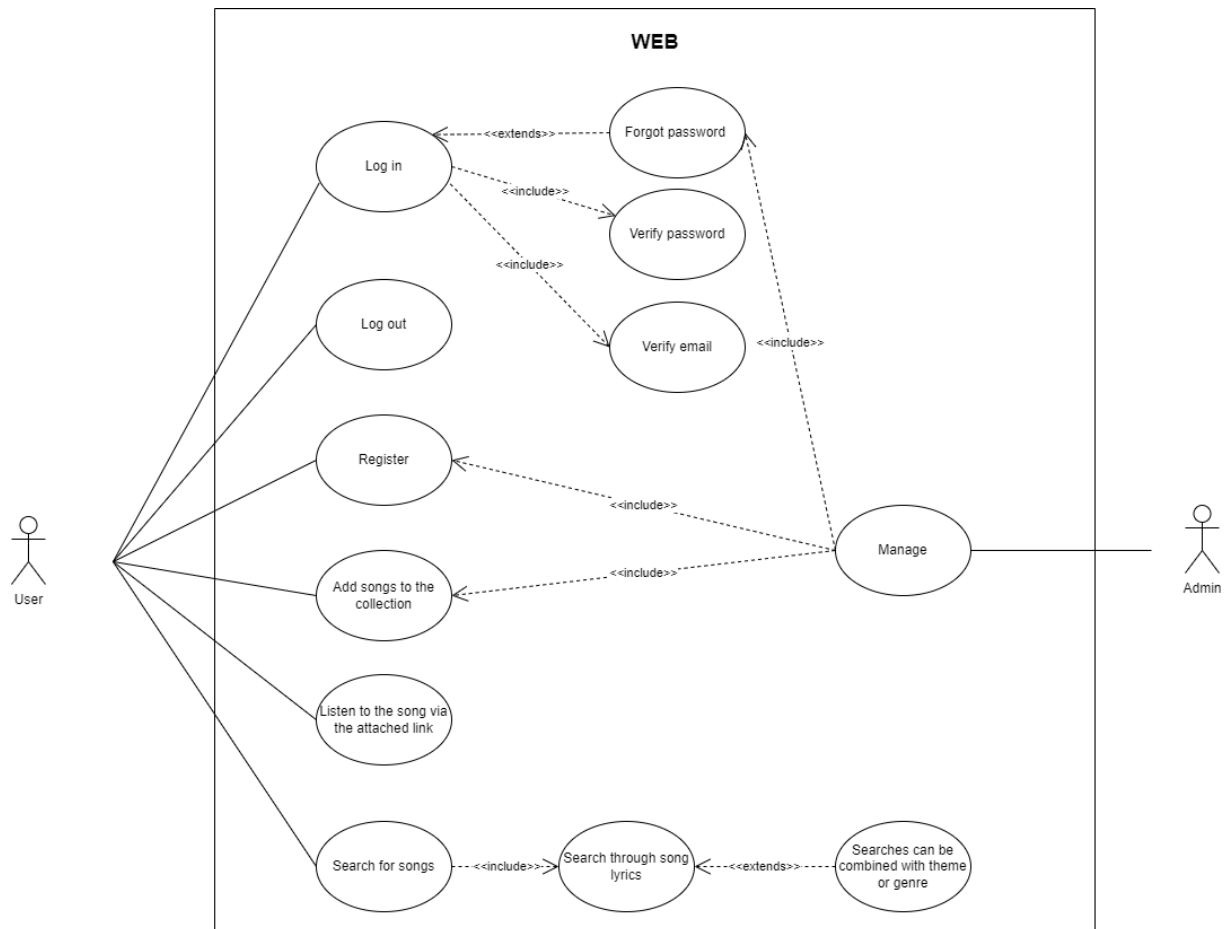


## 6.8 Normal account

Đối với các tài khoản thông thường thì chỉ có thêm tính năng Submit song chứ không có tính năng Verify song. Giao diện Submit song khi này cũng sẽ giống với tài khoản với quyền Admin.

## 7 Screen flow

## 8 Use-case diagram



## 9 Use-case scenarios

1. **Mở trình duyệt web:** người dùng mở trình duyệt web ưa thích trên máy tính hoặc thiết bị di động và kết nối với trang web.
2. **Bắt đầu tìm kiếm:** người dùng đăng nhập vào web sau đó tại thanh tìm kiếm người dùng gõ một số lời bài hát hoặc từ khóa của bài hát mà mình nhớ được.
3. **Truy vấn công cụ tìm kiếm:** người dùng nhấn nút tìm kiếm hoặc nhấn “Enter” trang web trả về danh sách kết quả tìm kiếm liên quan đến lời bài hát.

4. **Quét kết quả tìm kiếm:** người dùng quét qua kết quả tìm kiếm, tìm kiếm bất kỳ thông tin nào có thể phù hợp với bài hát họ đang tìm. Điều này có thể bao gồm lời bài hát, tên bài hát, tên nghệ sĩ hoặc các trang web liên quan đến âm nhạc.
5. **Thu thập thông tin:** người dùng thu thập thông tin về bài hát, chẳng hạn như tiêu đề, nghệ sĩ và bất kỳ lời bài hát bổ sung nào họ tìm thấy. Họ cũng có thể tìm thấy các liên kết đến các nền tảng phát trực tuyến như Spotify hoặc YouTube nơi họ có thể nghe bài hát.
6. **Nghe bài hát:** người dùng nhấp vào liên kết đến nền tảng phát nhạc và nếu bài hát có sẵn, họ có thể nghe bài hát đó để xác nhận xem đó có phải là bài hát họ đang tìm kiếm hay không.
7. **Additional research:** nếu họ không thể tìm thấy bài hát trong lần tìm kiếm đầu tiên, người dùng có thể tinh chỉnh truy vấn tìm kiếm của mình hoặc tìm kiếm trợ giúp.
8. **Thưởng thức âm nhạc:** giờ đây, người dùng có thể thưởng thức bài hát họ đang tìm kiếm, bằng cách phát trực tuyến, mua bài hát đó hoặc thêm bài hát đó vào bộ sưu tập nhạc của họ.

## 10 Key Technical decisions

### 10.1 BM25

Trong tìm kiếm thông tin, BM25 là hàm tính thứ hạng được các công cụ tìm kiếm sử dụng để xếp hạng các văn bản theo độ phù hợp với truy vấn nhất định. Hàm xếp hạng này dựa trên mô hình xác suất, được phát minh ra vào những năm 1970 – 1980. Thực chất, BM25 dựa trên nền tảng của TF/IDF, và cải tiến dựa trên lý thuyết probabilistic information retrieval. Vì vậy để hiểu được gốc rễ của nó trước tiên chúng ta cần phải biết TF/IDF là gì.

#### 10.1.1 Giới thiệu TF-IDF

Xử lý ngôn ngữ là một kĩ thuật quan trọng nhằm giúp máy tính hiểu được ngôn ngữ của con người, qua đó hướng dẫn máy tính thực hiện và giúp đỡ con người trong những công việc có liên quan đến ngôn ngữ như: tìm kiếm thông tin, dịch thuật, phân tích dữ liệu văn bản,... Nhóm chúng em sẽ ứng dụng kỹ thuật xử lý ngôn ngữ tự nhiên vào trang web tìm kiếm âm nhạc của mình. Trong đó, phương pháp chính là phương pháp TF-IDF (viết tắt của term frequency-inverse document frequency). Nhìn chung, phương pháp này cho phép ta tính tầm quan trọng của một từ trong một đoạn văn bản.

#### 10.1.2 Term frequency

Trong document  $d$ , frequency (tần số) biểu diễn số lần xuất hiện của từ  $t$ . Biểu diễn qua công thức:

$$tf(t, d) = \frac{\text{count of } t \text{ in } d}{\text{number of words in } d}$$

Nói cách khác  $tf(t, d)$  chính là tỉ số giữa số lần xuất hiện của từ  $t$  trong văn bản  $d$  với độ dài của văn bản đó.

### 10.1.3 Inverse document frequency

Tính đặc hiệu của thuật ngữ, được gọi là tần suất tài liệu nghịch đảo (IDF), là thước đo thiết yếu cho tầm quan trọng của một từ. Khi tính  $tf$  mức độ quan trọng của các từ coi là như nhau. Tuy nhiên trong văn bản thường xuất hiện nhiều từ không quan trọng xuất hiện với tần suất cao:

- Từ nối: và, hoặc,... (đối với tiếng Việt)
- Giới từ: trong, của, để...
- Từ dùng để chỉ: đó,...

Do đó chúng ta cần giảm mức độ quan trọng của những từ đó bằng IDF. Công thức được tính như sau:

$$idf(t, D) = \log \frac{\text{number of documents in } D}{d \in D: t \in d}$$

với  $d \in D : t \in d$  là số văn bản chứa từ  $t$  trong tập  $D$

### 10.1.4 TF - IDF

Trọng số  $tf-idf$  có xu hướng lọc ra các từ phổ biến và cho điểm cao cho các từ duy nhất. Được tính bằng công thức:

$$tf-idf(t, d, D) = \text{term frequency}(t, d) * \text{inverse document frequency}(t, D)$$

Vấn đề quan trọng nhất mà xử lý ngôn ngữ tự nhiên phải đối mặt là các mô hình học máy có xu hướng chỉ xử lý các giá trị số. Đây là một vấn đề, vì các con số không thể chỉ biểu thị ngôn ngữ tự nhiên, nếu không chúng sẽ mất ý nghĩa. Vì vậy, chúng ta phải vector hóa văn bản để chuyển nó thành số. Ngoài ra, sự đơn giản và dễ sử dụng của  $tf-idf$  là những lợi ích đáng kể nhất của nó. Kết quả là, nó dễ tính toán, chạy không tốn kém và là điểm khởi đầu rõ ràng cho các tính toán tương tự. Tuy nhiên,  $tf-idf$  không thể hỗ trợ mang ý nghĩa ngữ nghĩa. Nó cân nhắc các từ và cân nhắc chúng khi xác định tầm quan trọng của chúng, nhưng không phải lúc nào nó cũng có thể suy ra ngữ cảnh của cụm từ hoặc xác định ý nghĩa của chúng theo cách đó.

## 10.2 Tại sao là BM25?

BM25 giới thiệu độ bão hòa thuật ngữ, có nghĩa là khi TF tăng lên thì tác động của nó giảm đi. Điều này ngăn không cho một thuật ngữ duy nhất chiếm ưu thế trong bảng xếp hạng và cung

cấp cách trình bày nhiều sắc thái hơn về mức độ liên quan của tài liệu. Trong khi cả TF-IDF và BM25 đều sử dụng IDF để đo lường tính thông tin của các thuật ngữ thì thành phần IDF trong BM25 được điều chỉnh với hệ số làm mịn để cải thiện độ tin cậy, đặc biệt đối với các thuật ngữ có tần suất tài liệu thấp. Tóm lại, TF-IDF trong BM25 khác với TF-IDF bằng cách đưa ra một công thức phức tạp hơn bao gồm bão hòa thuật ngữ và chuẩn hóa độ dài tài liệu, cho phép BM25 cung cấp thứ hạng nhiều sắc thái và hiệu quả hơn trong các tác vụ truy xuất thông tin, đặc biệt là khi xử lý các tài liệu có độ dài khác nhau.

## 11 Tại sao lại chọn làm web?

- Khả năng tiếp cận rộng rãi: Trang web có thể truy cập được trên nhiều loại thiết bị, bao gồm máy tính cá nhân, máy tính bảng và điện thoại di động mà không cần phải tạo ứng dụng riêng cho từng hệ điều hành.
- Dễ dàng cập nhật và duy trì: Cập nhật nội dung và tính năng của trang web có thể thực hiện một cách nhanh chóng và dễ dàng, mà không cần phải chờ các bản cập nhật từ cửa hàng ứng dụng (app store).
- Tích hợp dễ dàng: Trang web có thể tích hợp với các dịch vụ và ứng dụng khác một cách dễ dàng qua API (Application Programming Interface).
- Tiện lợi cho việc tìm kiếm: Trang web có thể được tìm thấy dễ dàng trên công cụ tìm kiếm như Google, điều này giúp tăng cơ hội tiếp cận đối tượng mục tiêu của bạn.
- Khả năng chia sẻ nội dung: Trang web có thể dễ dàng chia sẻ qua mạng xã hội và email, giúp tạo sự lan truyền nội dung một cách nhanh chóng.

## 12 Sử dụng Next.js để xây dựng ứng dụng

1. **Định tuyến tệp:** Next.js cung cấp khả năng định tuyến tệp hợp lý, tinh chỉnh tổ chức và điều hướng dự án. Nó cung cấp khả năng ánh xạ trực tiếp các URL tới các tệp trong thư mục dự án, chuyển đổi các tệp thành các tuyến đường một cách hiệu quả, do đó nâng cao hiệu quả điều hướng.
2. **Thay thế mô-đun (HMR) để thuận tiện cho nhà phát triển:** Các nhà phát triển đánh giá cao những nền tảng mang lại sự tiện lợi và khả năng kiểm soát. Next.js kết hợp tính năng Thay thế mô-đun (HMR), cho phép sửa đổi mã theo thời gian thực mà không cần tải lại toàn bộ. Tính năng này tăng tốc đáng kể quá trình phát triển, mang lại hiệu quả cao hơn cho các nhà phát triển.
3. **Kết xuất phía máy chủ (SSR) để tải trang nhanh:** SSR là một kỹ thuật kết xuất phía máy chủ trong phát triển web, nơi các trang web được tạo và hiển thị trên máy chủ trước khi được gửi trực tiếp dưới dạng tài liệu HTML được định dạng đầy đủ đến trình



duyet của khách hàng. Cách tiếp cận này tăng tốc thời gian tải trang và bao gồm quy trình sau:

- Người dùng yêu cầu một trang web mới.
  - Máy chủ xác nhận và xử lý yêu cầu.
  - Nó tìm nạp và hiển thị nội dung HTML nếu cần.
  - Dữ liệu được kết xuất trước sẽ được gửi kịp thời đến trình duyệt của người dùng, hiển thị nội dung trực tiếp.
  - SSR vượt trội hơn so với kết xuất phía máy khách (CSR) truyền thống bằng cách cung cấp các trang tải nhanh hơn, tránh sự chậm trễ và đảm bảo trải nghiệm người dùng mượt mà hơn.
4. **Khả năng SEO với Next.js:** Next.js mang lại những lợi thế đáng kể cho các dự án, đặc biệt là về Tối ưu hóa Công cụ Tìm kiếm (SEO). SEO là yếu tố then chốt để xây dựng thương hiệu thành công và Next.js đóng góp đáng kể. Trình thu thập thông tin tìm kiếm của Google ưu tiên các trang nhanh, an toàn và nhiều thông tin. Các trang được phát triển bằng Next.js sở hữu những phẩm chất này, giúp công cụ tìm kiếm lập chỉ mục tốt hơn. Việc đưa vào sơ đồ trang web giúp cải thiện đáng kể khả năng hiển thị và khả năng khám phá thương hiệu, nâng cao hiệu suất SEO tổng thể.

## 13 Sử dụng Express.js để xây dựng ứng dụng

1. Express JS được thiết kế để giúp việc tạo API và ứng dụng web trở nên đơn giản.
2. Một lý do khác để sử dụng Express là nó được viết bằng JavaScript, một ngôn ngữ đơn giản, ngay cả khi bạn không có kiến thức trước về bất kỳ ngôn ngữ nào. Express JS cho phép một số lượng lớn các nhà phát triển mới tham gia vào lĩnh vực phát triển web.
3. Viết code tối thiểu sạch sẽ, hiệu quả, có thể tái sử dụng và chứa phần nhận xét thích hợp.
4. Tạo và duy trì các thành phần phía máy chủ. Tạo và triển khai các ứng dụng hiệu suất cao.
5. Tích hợp logic phía máy chủ vào các thành phần hướng tới người dùng do nhà phát triển giao diện người dùng tạo. Bao gồm bảo mật dữ liệu, đàm phán nội dung và các biện pháp an ninh mạng.