

IF-ITB-IF3250-PPL-2023

DOKUMEN PEMBANGUNAN PERANGKAT LUNAK

Technical Document

Radio Curated App

Versi: 28 April 2023

untuk:

Bharabas

Dipersiapkan oleh:

Ahmad Romy Zahran / 13520009

Aji Andhika Falah / 13520012


Primanda Adyatma Hafiz / 13520022

Farhan Hafiz / 13520027

Hansel Valentino Tanoto / 13520046

Program Studi Teknik Informatika - Institut Teknologi Bandung

Jl. Ganesha 10, Bandung 40132

	Nomor Dokumen		Halaman
	<i>IF-ITB-IF3250-PPL-2023</i>		1/39
	Revisi	-	Tgl: 28 Apr 2023

Daftar Isi

1 Pendahuluan	3
1.1 Tujuan Penulisan Dokumen	4
1.2 Pemilik Perangkat Lunak	4
1.3 Manfaat Perangkat Lunak	4
1.4 Aturan Penomoran	4
1.5 Referensi	4
1.6 Deskripsi Umum Dokumen (Ikhtisar)	4
2 Proses Kerja / Proses Bisnis	5
2.1 Unit Organisasi	6
2.2 Gambaran Proses Bisnis	6
2.3 Deskripsi Kebutuhan Perangkat Lunak	7
3 Kebutuhan Perangkat Lunak	7
3.1 Deskripsi Umum Sistem	7
3.2 Stakeholder	8
3.3 Identifikasi Kebutuhan Perangkat Lunak	8
3.3.1 Product Backlog	8
3.3.2 Sprint Backlog	9
3.3.3 Model Kebutuhan	10
4 Deskripsi Perancangan Perangkat Lunak	12
4.1 Deskripsi Arsitektur	12
4.2 Dekomposisi Fungsional	13
4.3 Architectural View	13
4.3.1 Structural View	13
4.3.2 Behavioural View	14
4.3.2.1 Sequence Diagram UC-1: Access Audio List	15
4.3.2.2 Sequence Diagram UC-2: Play an Audio Content	15
4.3.2.3 Sequence Diagram UC-6: Manage Audio Contents	16
4.3.2.4 Sequence Diagram UC-7: Login	17
4.3.3 Deployment View	17
5 Implementasi Perangkat Lunak	18
6 Pengujian Perangkat Lunak	23
6.1 Automated Unit Testing	23
6.2 Manual Unit Testing	23
6.2.1 Scraping Konten Audio	23
6.2.2 Mendengarkan Konten Audio	24
6.2.3 Filter Konten Audio	25
6.2.4 Pencarian Konten Audio	25
6.2.5 Mengelola Konten Audio	25
6.3 Quality Code Testing	27

7 Petunjuk Instalasi Perangkat Lunak	28
8 Panduan Penggunaan Perangkat Lunak	31
8.1 Halaman Utama	31
8.2 Halaman Daftar Konten Audio	32
8.2.1 Pemilihan Berdasarkan Kategori	32
8.2.2 Pencarian Berdasarkan Input Pengguna	33
8.2.3 Pemutaran Konten Audio	33
8.3 Halaman Detail Audio	34
8.3.1 Detail Audio	34
8.3.2 Platform Sumber	34
8.4 Halaman Admin	35
8.4.1 Login	35
8.4.2 Dashboard	36
8.4.3 Menambah Kategori	37
8.4.4 Menambah Konten Audio	37
8.4.5 Meng-edit Konten Audio	38
8.4.6 Melakukan Scraping Konten Audio	38
8.4.7 Mengelola Job Scraping Konten Audio	39

1 Pendahuluan

1.1 Tujuan Penulisan Dokumen

Dokumen Teknis ini digunakan oleh tim pembuat perangkat lunak untuk kebutuhan dokumentasi dan komunikasi dalam proyek pengembangan perangkat lunak *Radio Curated App* yang diajukan oleh Bharabas. Dokumen ini akan berisi gambaran umum dan penjelasan secara detail mengenai perangkat lunak yang dikembangkan.

1.2 Pemilik Perangkat Lunak

Pemilik dari perangkat lunak ini adalah Bapak Wiku Bhaskoro sebagai perwakilan dari pihak Bharabas (Bhaktera Bahana Suara) Radio 97.5 FM Pekanbaru. Bharabas Radio adalah stasiun radio yang mengudara pada frekuensi 97.5 FM dan sudah berdiri sejak 8 Maret 1978. Saat ini, Bharabas Radio berbasis di Jalan Kali Putih No. 3, Pekanbaru, Indonesia dan memiliki *tagline* “*The Best News and Information*” karena programnya yang berfokus pada berita dan informasi.

1.3 Manfaat Perangkat Lunak

Perangkat lunak Bharabas *Radio Curated App* ini bermanfaat bagi penggunanya untuk mendengarkan konten audio non-musik dari berbagai platform yang dikumpulkan dalam sebuah *website*. Selain itu, perangkat lunak ini juga menyediakan penyaringan melalui kategori dan pencarian konten audio untuk memudahkan penggunanya.

1.4 Aturan Penomoran

Berikut adalah aturan penomoran yang digunakan dalam dokumen ini.

No	Aspek	Aturan Penomoran
1	Bab dan Subbab	Penomoran bab pada dokumen ini menggunakan angka arab yang dimulai dari 1. Sedangkan, penomoran subbab dilakukan dengan menuliskan nomor bab yang diikuti dengan titik dan nomor subbab.
2	Gambar	Penomoran gambar pada dokumen ini menggunakan format “Gambar X.Y Nama” dengan X merupakan penomoran subbab tempat gambar tersebut berada dan Y adalah nomor urutan dari gambar pada sub bab tersebut. Kemudian, Nama adalah nama dari gambar tersebut.
3	Tabel	Penomoran tabel pada dokumen ini menggunakan format “Tabel X.Y Nama” dengan X merupakan penomoran subbab tempat tabel tersebut berada dan Y adalah nomor urutan dari tabel pada subbab tersebut. Kemudian, Nama adalah nama dari tabel tersebut.
4	<i>Product Backlog</i>	Penomoran <i>product backlog</i> pada dokumen ini menggunakan format “PB-XX” dengan PB adalah singkatan dari <i>Product Backlog</i> dan XX adalah nomor (indeks) dari sebuah <i>product backlog</i> .
5	<i>Sprint Backlog</i>	Penomoran <i>sprint backlog</i> pada dokumen ini menggunakan format “SB-X-YY” dengan SB adalah singkatan dari <i>Sprint Backlog</i> , X adalah nomor dari <i>sprint</i> , dan YY adalah nomor dari <i>sprint backlog</i> pada <i>sprint</i> tersebut.

1.5 Referensi

Referensi yang kami gunakan dalam pembuatan dokumen ini adalah:

1. Modul Mata Kuliah IF2250 Rekayasa Perangkat Lunak
2. Modul Mata Kuliah IF3150 Manajemen Proyek Perangkat Lunak

1.6 Deskripsi Umum Dokumen (Ikhtisar)

Dokumen ini terdiri dari 8 bab, yaitu Bab 1 Pendahuluan yang berisi pengantar berupa tujuan penulisan, pemilik perangkat lunak, manfaat perangkat lunak, aturan penomoran, referensi yang digunakan, dan ikhtisar dokumen.

Program Studi Teknik Informatika ITB	IF-ITB-IF3250-PPL-2023	Halaman 4/ dari 39 halaman
Template dokumen ini dan informasi yang dimilikinya adalah milik Program Studi Teknik Informatika-ITB, serta bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Program Studi Teknik Informatika ITB.		

Lalu, Bab 2 Proses Kerja/Proses Bisnis yang berisi penjelasan unit organisasi, gambaran umum proses bisnis dan deskripsi kebutuhan perangkat lunak. Selanjutnya, Bab 3 Kebutuhan Perangkat Lunak, berisi deskripsi umum sistem, daftar *stakeholder*, dan identifikasi kebutuhan perangkat lunak berupa *product backlog*, *sprint backlog*, dan model kebutuhan. Bab 4 Deskripsi Perancangan Perangkat Lunak berisi deskripsi arsitektur, dekomposisi fungsional, dan *architectural view* berupa *structural*, *behavioural*, dan *deployment view*. Kemudian, dilanjutkan Bab 5 Implementasi Perangkat Lunak, Bab 6 Pengujian Perangkat Lunak, Bab 7 Petunjuk Instalasi Perangkat Lunak, dan Bab 8 Panduan Penggunaan Perangkat Lunak.

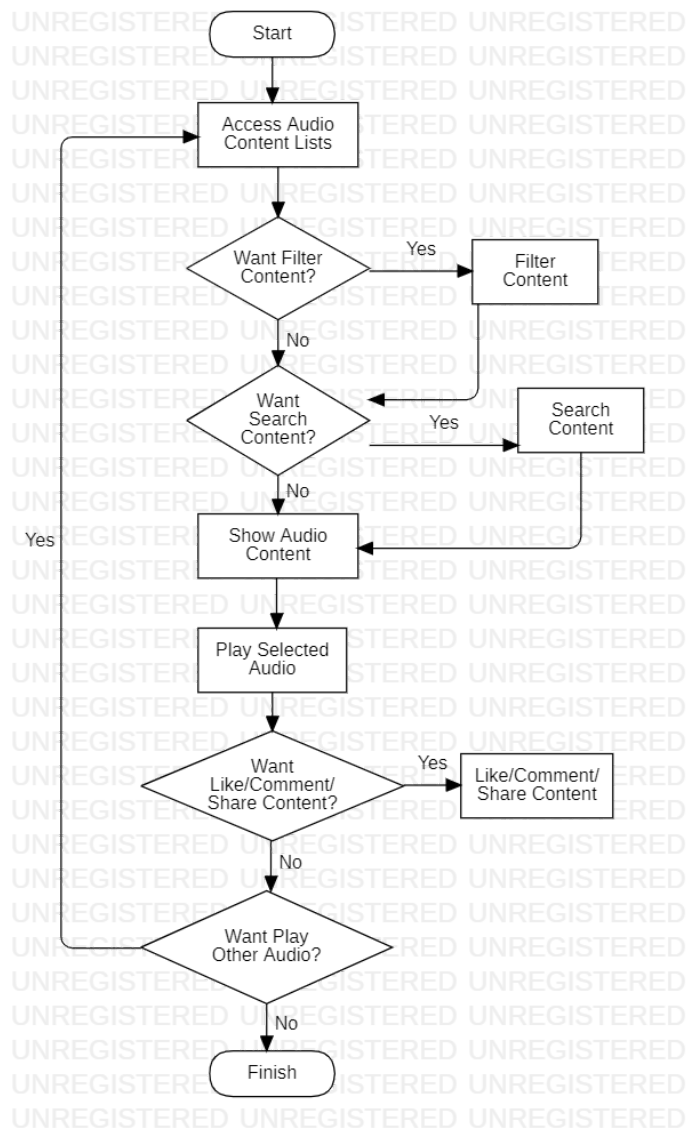
2 Proses Kerja / Proses Bisnis

2.1 Unit Organisasi

Perangkat lunak yang dikembangkan ini akan digunakan dalam sebuah proyek dari Bharabas Radio. Proyek Bharabas ini bertujuan untuk membuat sebuah *website* yang mengumpulkan berbagai konten audio dari berbagai platform terutama konten-konten audio yang jarang didengar oleh pengguna. Harapannya, pengguna mendapatkan berita atau informasi yang bermanfaat sesuai dengan *tagline* Bharabas yaitu “*The Best News and Information*”.

2.2 Gambaran Proses Bisnis

Proses bisnis yang dilakukan dalam pembuatan Bharabas *Radio Curated website* ini adalah penyediaan berita dan informasi melalui konten audio non musik. Proses ini dimulai dari pengguna yang ingin mencari atau melihat suatu konten audio hingga mendengarkan konten tersebut.



Pada *flowchart diagram* di atas, proses dimulai dari *website* yang menampilkan halaman *home* berupa *list* dari konten audio yang tersedia. Kemudian, pengguna dapat memilih apakah ingin melakukan *filter* dan pencarian konten audio sesuai keinginan pengguna. Pengguna akan melihat daftar konten audio sesuai keinginan pengguna sebelumnya dan dapat memainkan konten audio yang dipilih. Setelah itu, pengguna dapat melakukan aksi *like*,

comment, ataupun *share* konten audio yang sedang dipilih. Apabila pengguna tidak ingin mendengarkan konten audio lainnya, maka proses selesai.

2.3 Deskripsi Kebutuhan Perangkat Lunak

Untuk mewujudkan perangkat lunak yang telah digambarkan sebelumnya, perangkat lunak memiliki beberapa kebutuhan antara lain :

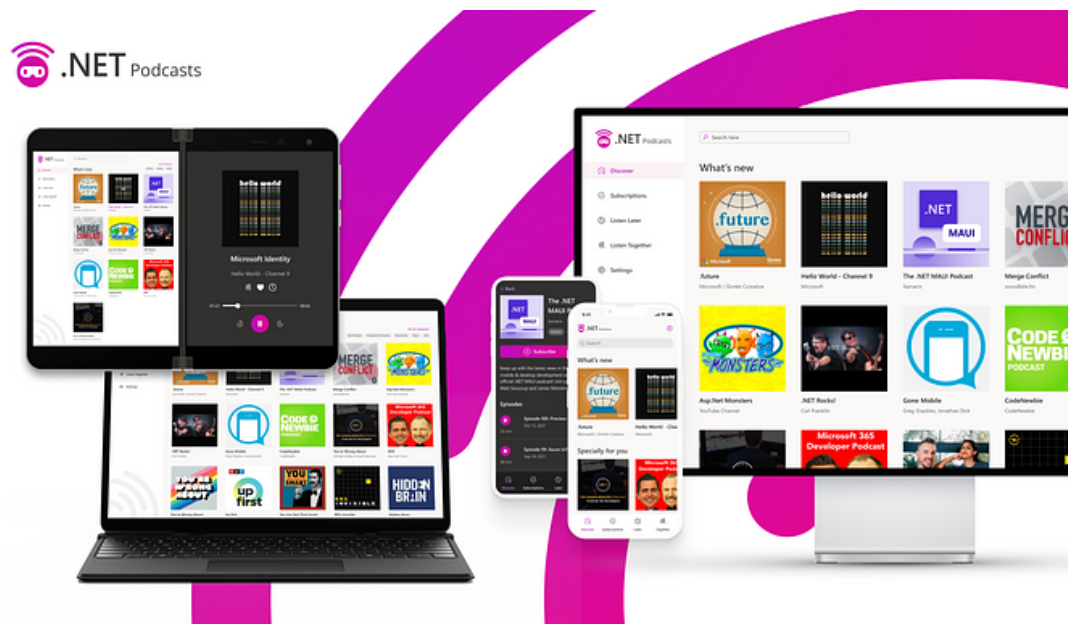
1. Perangkat lunak dapat menampilkan konten audio
2. Perangkat lunak memiliki fitur filter konten audio dengan kategori tertentu
3. Perangkat lunak memiliki fitur pencarian konten audio melalui *input* pengguna
4. Perangkat lunak memiliki fitur memainkan konten audio yang dipilih pengguna
5. Perangkat lunak memiliki fitur *feedback* berupa *like* dan *comment* pada konten audio yang dipilih
6. Perangkat lunak memiliki fitur *share* melalui aplikasi lain pada konten audio yang dipilih

3 Kebutuhan Perangkat Lunak

3.1 Deskripsi Umum Sistem

Bharabas adalah *website* kurasi konten audio non musik dari berbagai platform. Sebuah *website* yang berisi berbagai konten audio non musik yang dikumpulkan dari semua platform yang menyediakan konten audio, seperti Spotify, Google Podcast, Noice, YouTube, Apple Podcast, Soundcloud, Audible, Anchor, serta radio-radio lokal dan internasional.

Konten audio non musik ini nantinya akan disegmentasi berdasarkan kategori atau topik tertentu, antara lain Edukasi, Hiburan, Gaya hidup, Finansial, Teknologi, *Programming*, Bisnis, Manajemen, Fiksi Horor, Fiksi Drama, Psikologi, *Parenting*, *Mental Health*, dan lain-lain. Berikut adalah contoh ilustrasi *website* ini.



Gambar 3.1.1 Ilustrasi tampilan *website*

Sesuai ilustrasi di atas, *website* ini nantinya akan dapat diakses melalui perangkat komputer maupun *mobile* seperti *smartphone* dan tablet sehingga *website* yang dibuat harus bersifat responsif.

Secara umum fitur-fitur yang akan terdapat pada *website* ini, yaitu:

- Mengumpulkan konten audio non musik dari semua platform yang ada di internet (*web scraping*). Data hasil *scraping* ini akan diperbarui minimal setiap 3 hari sekali.
- Mendengarkan langsung konten audio non musik. Konten musik yang diputar pada *web* ini akan bersifat *embedded* sehingga perhitungan *view* akan masuk ke sumber konten aslinya.

- Ada segmentasi konten audio berdasarkan kategori tertentu
- Fitur *search* (pencarian) konten audio
- *Like, share* dan *comment* konten audio tertentu. Fitur ini dapat dilakukan secara anonim sehingga tidak memerlukan fitur *login*.

Website ini akan diimplementasikan menggunakan *framework* ASP.NET Core dan penyimpanan basis datanya akan menggunakan PostgreSQL.

3.2 Stakeholder

Berikut adalah daftar *stakeholder* yang terlibat dalam proyek ini:

No	Stakeholder	Deskripsi
1	Project Owner (PO)	<i>Stakeholder</i> ini merupakan pemilik proyek perangkat lunak yang akan dikembangkan ini. PO berperan dalam memberikan kebutuhan perangkat lunak, memberi saran dalam pengembangan perangkat lunak, dan menerima hasil akhir perangkat lunak.
2	Scrum Master	<i>Stakeholder</i> ini berperan dalam memimpin dan mengelola keberjalanan pengembangan proyek perangkat lunak pada tiap <i>sprint</i> -nya.
3	Developer	<i>Stakeholder</i> ini berperan dalam mengembangkan desain dan implementasi dari perangkat lunak.
4	Reviewer	<i>Stakeholder</i> ini berperan dalam meninjau <i>source code</i> apakah sudah sesuai dengan kaidah yang ditentukan.
5	Tester	<i>Stakeholder</i> ini berperan dalam menguji perangkat lunak apakah sudah berjalan dengan baik dan sesuai dengan spesifikasi kebutuhan.

3.3 Identifikasi Kebutuhan Perangkat Lunak

3.3.1 Product Backlog

Berikut ini adalah *product backlog* untuk proyek ini:

ID	Product Backlog
PB-01	<i>Website</i> dapat mengumpulkan konten audio non musik dari berbagai macam platform di internet, seperti Spotify, YouTube, Google Podcast, ListenNotes, dll.
PB-02	Sebagai pengguna, saya dapat melakukan filter audio berdasarkan kategori atau topik tertentu
PB-03	Sebagai pengguna, saya dapat mendengarkan langsung konten audio non musik tanpa membuka <i>website</i> lain.
PB-04	Sebagai pengguna, saya dapat melakukan pencarian konten audio yang sudah dikumpulkan dari berbagai platform.
PB-05	Sebagai pengguna, saya dapat memberikan <i>feedback</i> berupa <i>like</i> dan <i>comment</i> pada konten audio yang dipilih
PB-06	Sebagai pengguna, saya dapat melakukan <i>share</i> melalui aplikasi lain pada konten audio yang dipilih
PB-07	Sebagai pengguna, saya dapat melihat gambar dan deskripsi yang jelas dari audio yang berkaitan
PB-08	Sebagai pengguna, saya dapat melihat dokumentasi sebagai pedoman manual dalam menggunakan produk
PB-09	Sebagai admin, saya dapat mengelola konten yang ada pada <i>website</i>

Keterangan:

- PB-05 dan PB-06 dibatalkan sesuai kesepakatan bersama *product owner*. Dari sisi implementasi sendiri, *database* dan *front-end* untuk fitur *comment*, *like*, dan *share* sudah selesai dibuat. Lalu, khusus untuk fitur *comment* juga sudah dapat digunakan pada *website*, tetapi untuk sementara ketiga fitur tersebut akan disembunyikan.

3.3.2 Sprint Backlog

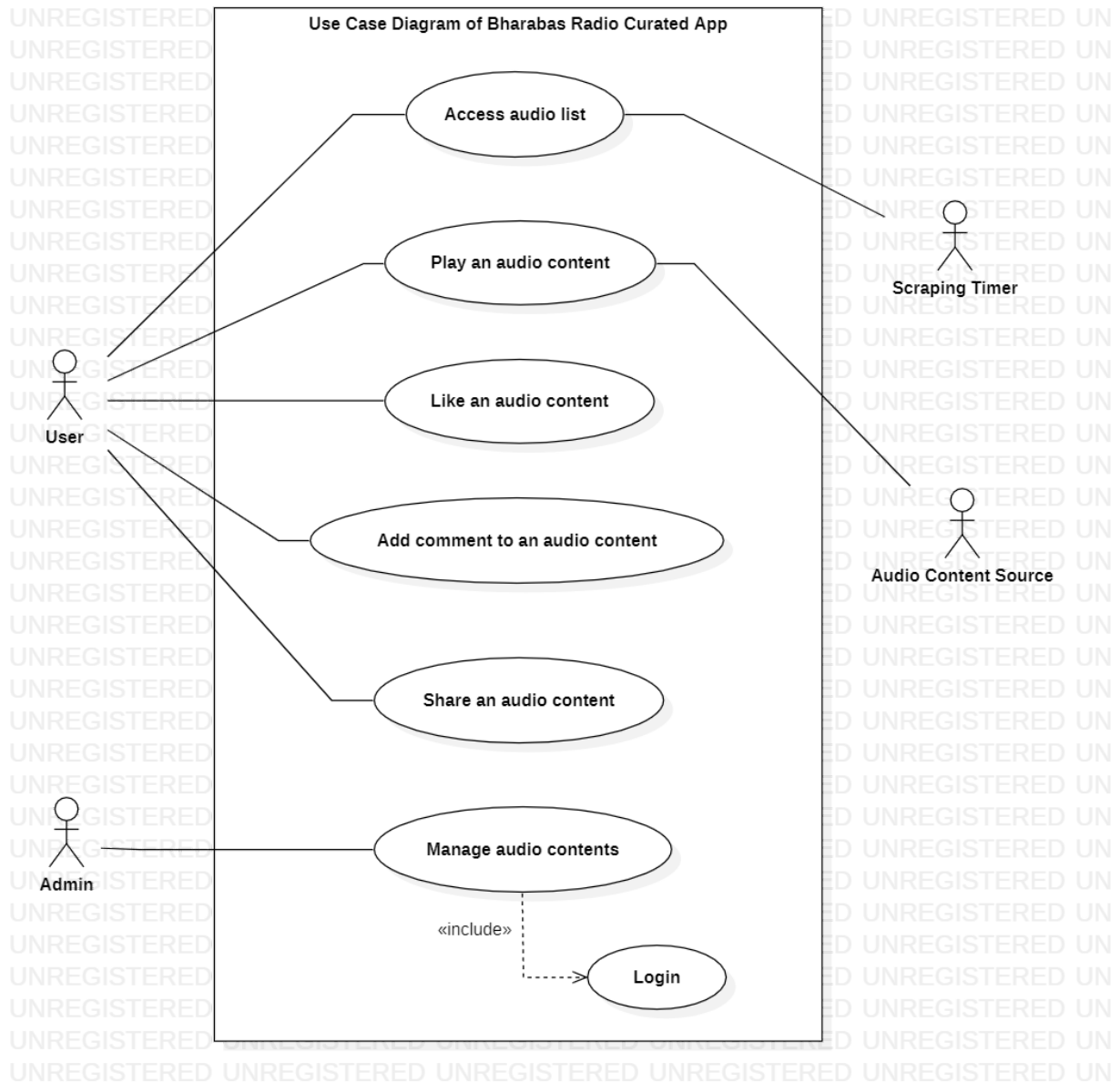
Berikut ini adalah daftar keseluruhan *sprint backlog* dari *sprint* 1 hingga 5 untuk proyek ini:

<i>Sprint#</i>	ID	Task	ID Product Backlog Terkait
1	SB-1-1	<i>Setup dan Initial Project</i>	-
	SB-1-2	<i>Setup dan Initial Database</i>	-
	SB-1-3	<i>Coding: Scraping data konten audio dari YouTube</i>	PB-01
	SB-1-4	<i>Coding: Scraping data konten audio dari ListenNotes</i>	PB-01
	SB-1-5	<i>Coding: Scraping data konten audio dari Google Podcast</i>	PB-01
	SB-1-6	<i>Coding: Front-end tampilan daftar konten audio</i>	PB-03
	SB-1-7	<i>Coding: Back-end tampilan daftar konten audio</i>	PB-03
	SB-1-8	<i>Coding: Back-end audio dapat didengar oleh pengguna</i>	PB-03
2	SB-1-3	<i>Coding: Scraping data konten audio dari YouTube menggunakan YouTube API</i>	PB-01
	SB-2-1	<i>Coding: Front-end halaman utama (beranda) website (responsif dan menarik)</i>	PB-02 PB-03 PB-04 PB-07
	SB-2-2	<i>Coding: Front-end mendengarkan konten audio (responsif dan menarik)</i>	PB-03 PB-07
	SB-2-3	<i>Coding: Membuat audio player</i>	PB-03
	SB-2-4	<i>Coding: Back-end filter (kategori) konten audio</i>	PB-02
	SB-2-5	<i>Coding: Back-end search konten audio</i>	PB-04
	SB-2-6	<i>Coding: Menambah link konten audio yang diambil dari ListenNotes ke platform asalnya</i>	PB-03 PB-07
	SB-2-7	<i>Coding: Menambah link konten audio yang diambil dari Google Podcast ke platform asalnya</i>	PB-03 PB-07
	SB-2-8	<i>Coding: Menambah link konten audio yang diambil dari YouTube ke platform asalnya</i>	PB-03 PB-07
	SB-2-9	<i>Deployment database</i>	PB-01
	SB-2-10	<i>Deployment website</i>	PB-01
3	SB-2-1	<i>Coding: Front-end halaman utama (beranda) website (responsif dan menarik)</i>	PB-02 PB-03 PB-04 PB-07
	SB-2-4	<i>Coding: Back-end filter (kategori) konten audio</i>	PB-02
	SB-3-1	<i>Coding: Scraping data konten audio dari Anchor</i>	PB-01
	SB-3-2	<i>Coding: Mengubah front-end halaman utama (responsif dan menarik)</i>	PB-02 PB-03 PB-04 PB-07
	SB-3-3	<i>Coding: Mengubah front-end halaman mendengarkan konten audio (responsif dan menarik)</i>	PB-03 PB-07
	SB-3-4	<i>Coding: Memfilter konten yang layak untuk di-scraping</i>	PB-01
	SB-3-5	<i>Coding: Mendengarkan konten dari hasil scraping YouTube</i>	PB-03
	SB-3-6	<i>Coding: Back-end tambah komentar pada setiap konten audio</i>	PB-05
4	SB-3-7	<i>Coding: Front-end tambah komentar pada setiap konten audio</i>	PB-05
	SB-3-1	<i>Coding: Scraping data konten audio dari Spotify</i>	PB-01
	SB-3-5	<i>Coding: Mendengarkan konten dari hasil scraping YouTube</i>	PB-03
	SB-4-1	<i>Coding: Front-end fitur login untuk dashboard Bharabas</i>	PB-09
	SB-4-2	<i>Coding: Back-end fitur login untuk dashboard Bharabas</i>	PB-09
	SB-4-3	<i>Coding: Front-end fitur menambah dan mengurangi konten untuk dashboard Bharabas</i>	PB-09
	SB-4-4	<i>Coding: Back-end fitur menambah dan mengurangi konten untuk dashboard Bharabas</i>	PB-09

	SB-4-5	<i>Coding: Front-end</i> fitur membuat kategori untuk <i>dashboard</i> Bharabas	PB-09
	SB-4-6	<i>Coding: Back-end</i> fitur membuat kategori untuk <i>dashboard</i> Bharabas	PB-09
	SB-4-7	<i>Coding: Front-end</i> fitur <i>edit</i> detail konten untuk <i>dashboard</i> Bharabas	PB-09
	SB-4-8	<i>Coding: Back-end</i> fitur <i>edit</i> detail konten untuk <i>dashboard</i> Bharabas	PB-09
	SB-4-9	<i>Coding: Front-end</i> filter konten menggunakan <i>sidebar</i>	PB-02 PB-09
5	SB 5-1	<i>Coding: Memperbaiki deployment</i> pada Azure	PB-01
	SB 5-2	<i>Coding: Membuat CI/CD</i>	PB-01
	SB 5-3	<i>Coding: Memperbaiki front-end</i> untuk fitur <i>user</i> umum	PB-02 PB-03 PB-04 PB-05
	SB 5-4	<i>Coding: Memperbaiki front-end</i> untuk fitur <i>user</i> admin	PB-09
	SB 5-5	<i>Coding: Front-end dashboard scraping</i>	PB-09
	SB 5-6	<i>Coding: Back-end dashboard scraping</i>	PB-09
	SB 5-7	<i>Coding: Timer scraping</i>	PB-01
	SB 5-8	<i>Coding: Automated unit testing</i>	PB-01 PB-02 PB-03 PB-04 PB-05 PB-06 PB-07 PB-09
	SB 5-9	<i>Coding: Manual unit testing</i> untuk fitur <i>user</i> umum	PB-01 PB-02 PB-03 PB-04 PB-05 PB-06 PB-07
	SB 5-10	<i>Coding: Manual unit testing</i> untuk fitur <i>user</i> admin	PB-01 PB-09
	SB 5-11	<i>Coding: Quality code testing</i>	PB-01 PB-02 PB-03 PB-04 PB-05 PB-06 PB-07 PB-09
	SB 5-12	<i>Document: Sequence diagram</i> dan <i>deployment view diagram</i>	PB-08
	SB 5-13	<i>Document: Petunjuk instalasi perangkat lunak</i>	PB-08
	SB 5-14	<i>Document: Pedoman penggunaan perangkat lunak</i>	PB-08
	SB 5-15	<i>Document: Melengkapi dokumen bagian lainnya</i>	PB-08

3.3.3 Model Kebutuhan

Berikut adalah diagram *use case* untuk proyek ini:



Gambar 3.3.1 Use case diagram

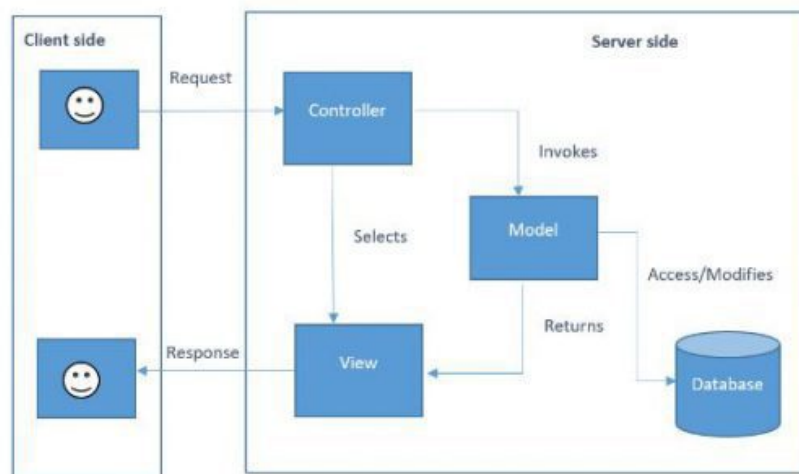
Website Bharabas ini memiliki 2 orang aktor fisik, yaitu *User* dan *Admin*, serta 2 aktor lainnya yang berbentuk entitas/program eksternal, yaitu *Scraping Timer* dan *Audio Content Source*. *User* pada website ini bersifat umum (*anonymous*) karena website ini tidak menyimpan data/kredensial dari *user* karena fitur *login* hanya terbatas untuk *admin* yang dibutuhkan sebagai autentikasi agar dapat mengelola konten audio. Berikut merupakan deskripsi dari setiap *use case*.

ID	Use Case	Deskripsi
UC-1	<i>Access audio list</i>	<ul style="list-style-type: none"> - <i>User</i> dapat melihat daftar konten audio yang tersedia serta melakukan <i>search</i> maupun <i>filter</i> konten audio - <i>Scraping Timer</i> adalah aktor yang akan menjadi penyuplai konten audio untuk disimpan ke database secara periodik (minimal 3 hari sekali)
UC-2	<i>Play an audio content</i>	<ul style="list-style-type: none"> - <i>User</i> dapat mendengarkan konten audio melalui <i>audio player</i> yang tersedia pada website

		<ul style="list-style-type: none"> - <i>Audio Content Source</i> merupakan aktor yang akan menjadi sumber dari konten audio yang diputar, misalnya YouTube, Google Podcast, ListenNotes, dan lain-lain.
UC-3	<i>Likes an audio content</i>	<ul style="list-style-type: none"> - <i>User</i> dapat memberikan <i>like</i> pada sebuah konten audio
UC-4	<i>Add comment to an audio content</i>	<ul style="list-style-type: none"> - <i>User</i> dapat menambahkan komentar pada sebuah konten audio
UC-5	<i>Share an audio content</i>	<ul style="list-style-type: none"> - <i>User</i> dapat membagikan <i>link</i> dari sebuah konten audio ke platform lain
UC-6	<i>Manage audio contents</i>	<ul style="list-style-type: none"> - Admin dapat mengelola konten audio yang terdapat dalam <i>database</i> melalui sebuah halaman <i>dashboard</i>. Hal-hal yang dapat dilakukan, yaitu menambahkan, menghapus, dan meng-<i>edit</i> konten audio atau kategori
UC-7	<i>Login</i>	<ul style="list-style-type: none"> - Admin dapat melakukan <i>login</i> untuk bisa mengakses fitur mengelola konten audio

4 Deskripsi Perancangan Perangkat Lunak

4.1 Deskripsi Arsitektur



Gambar 4.1.1 Arsitektur perangkat lunak

Arsitektur yang digunakan pada *website Bharabas Radio Curated App* ini adalah arsitektur *client-server* yang keduanya diimplementasikan menggunakan *framework* ASP.NET Core. Arsitektur ini diimplementasikan dengan model MVC (*Model-View-Controller*) dengan bagian model berupa pendefinisian kelas yang merepresentasikan struktur basis data yang menggunakan PostgreSQL. Lalu, *View* merupakan bagian visual yang akan ditampilkan ke pengguna. Terakhir, *Controller* merupakan bagian yang bertanggung jawab memanipulasi model. Alasan pemilihan arsitektur dengan model ini adalah untuk memudahkan proses *retrieval* dan modifikasi data dari basis data, serta memudahkan kebutuhan *scaling* di masa mendatang. *Website* dan basis data ini sendiri akan di-*deploy* pada *Azure Web Service*.

4.2 Dekomposisi Fungsional

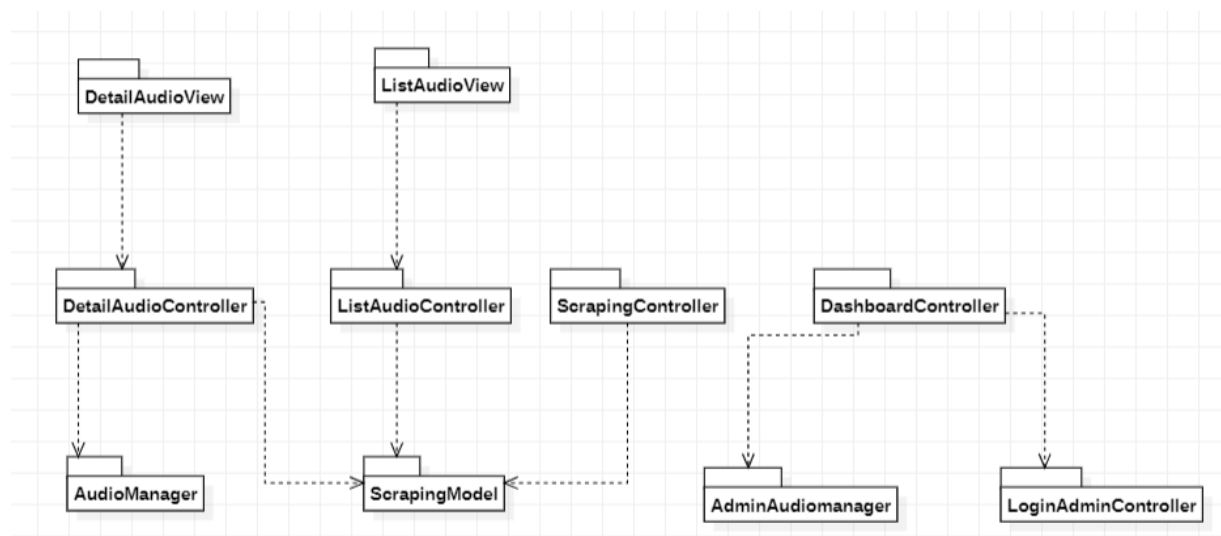
Berikut adalah dekomposisi fungsional dari *website* Bharabas ini.

ID	Nama subsistem/komponen/kelas	Tanggung Jawab	Role-Stereotype
K-1	<i>Scraping</i> Konten Audio	Mengumpulkan data konten audio dari internet dan menyimpannya ke dalam <i>database</i>	<i>Information Holder</i>
K-2	Mendengarkan Konten Audio	Memutar (<i>play</i>) konten audio yang dipilih <i>user</i>	<i>Service Provider</i>
K-3	Filter Konten Audio	Menunjukkan daftar konten audio sesuai kriteria pada filter yang digunakan	<i>Structurer</i>
K-4	Pencarian Konten Audio	Menunjukkan daftar konten audio berdasarkan <i>keyword</i> yang diketikkan	<i>Structurer</i>
K-5	<i>Like</i> Konten Audio	Menyukai salah satu konten audio	<i>Service Provider</i>
K-6	<i>Comment</i> Konten Audio	Menambahkan komentar ke salah satu konten audio	<i>Service Provider</i>
K-7	<i>Share</i> Konten Audio	Menyebarkan konten audio ke aplikasi eksternal	<i>Service Provider</i>
K-8	Mengelola Konten Audio	Melakukan operasi CRUD (<i>Create, Read, Update, Delete</i>) pada konten audio dan kategorinya	<i>Controller</i>

4.3 Architectural View

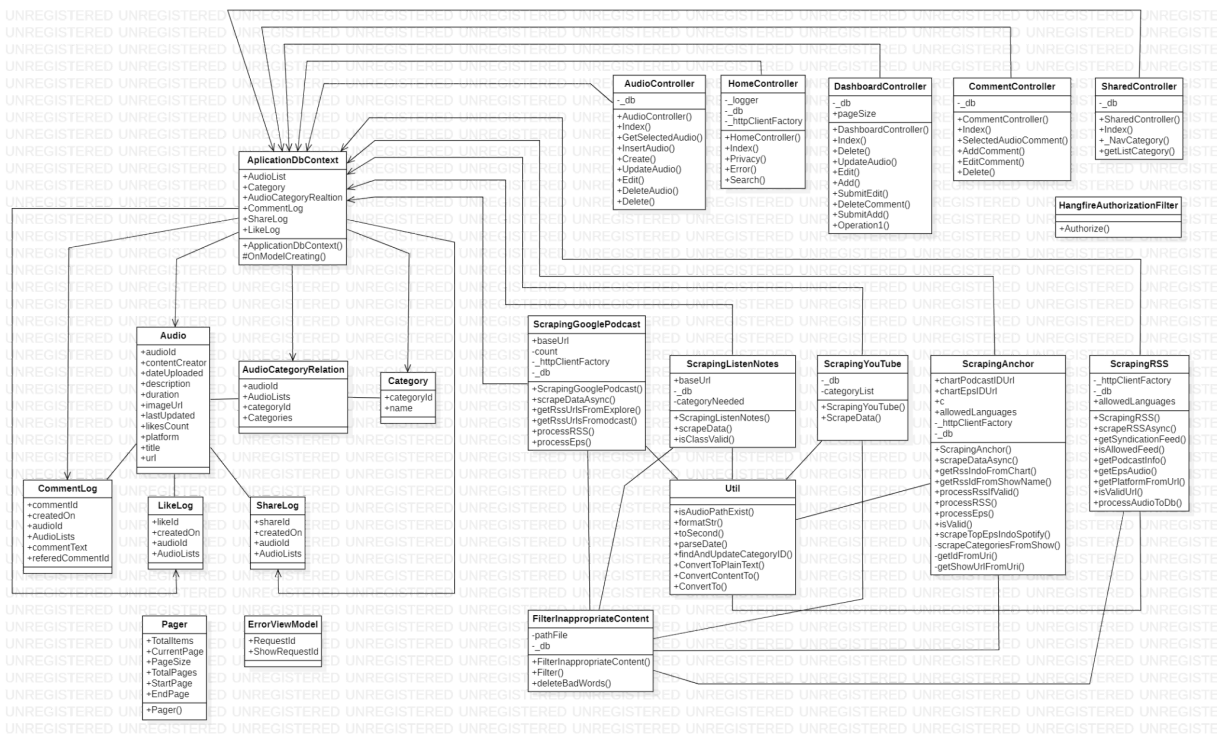
4.3.1 Structural View

Berikut adalah *package diagram* untuk proyek ini:



Gambar 4.3.1 *Package diagram*

Package diagram terdiri atas 2 komponen utama yaitu *package* yang berkaitan dengan *user* umum dan *package* yang berkaitan dengan *user* admin. Pada *user* umum, *package* meliputi komponen untuk menampilkan *list* audio dan detail konten audio yang dilengkapi dengan fitur-fitur tambahan untuk melakukan pencarian ataupun pemfilteran. Selain itu, *package* yang digunakan oleh *user* umum juga meliputi komponen untuk melakukan *scraping* konten audio. Kemudian untuk *package* dari *user* admin, komponennya meliputi *dashboard* untuk manajemen konten audio yang sekaligus mencakup fungsionalitas *login* khusus untuk admin. Berikut adalah detail dari *package* yang terdapat di proyek ini dalam bentuk *class diagram*:



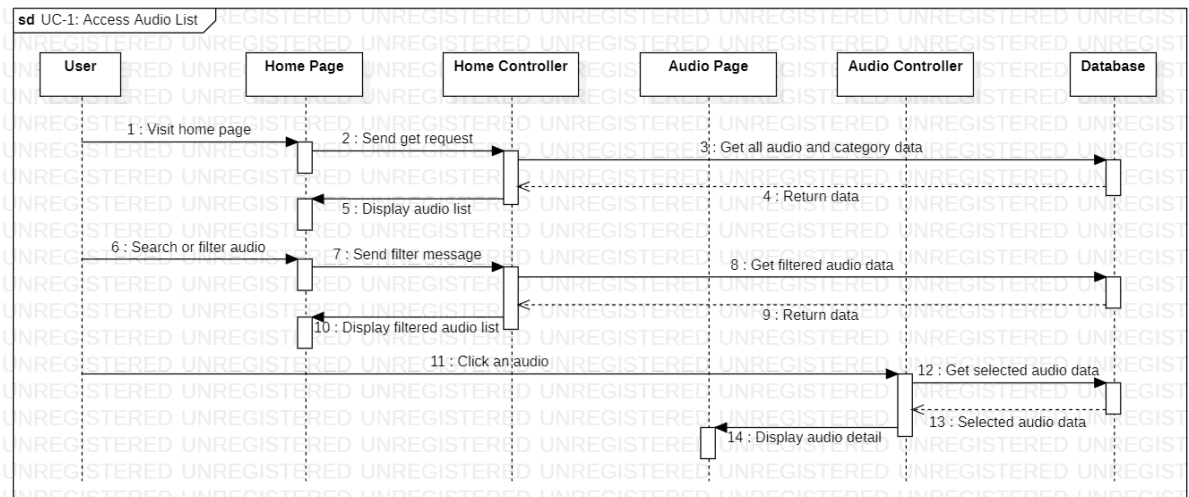
Gambar 4.3.2 Class diagram

Sedangkan untuk *class diagram*, pada dasarnya merupakan detail implementasi dari *package diagram*. *Class diagram* diimplementasikan terbagi atas 3 kelompok besar yaitu kelompok kelas model, *controller*, dan *scraping*. Kelas model sendiri berisi definisi dari setiap atribut tabel pada *database* yang digunakan oleh *software* sebagai data yang disimpan secara persisten. Kelas model meliputi kelas *Audio*, *AudioCategoryRelation*, *Category*, *CommentLog*, *LikeLog*, *ShareLog*, dan *ErrorViewModel*. Adapun definisi dari keseluruhan model tersebut disimpan dalam kelas *ApplicationDbContext*. Kemudian untuk kelas *controller* mendefinisikan *query* yang dilakukan pada model untuk nantinya ditampilkan pada layar *browser*. Controller meliputi kelas *AudioController*, *HomeController*, *CommentController*, dan *DashboardController*. Selanjutnya terdapat pula kelas *scraping* yang tugasnya yaitu untuk melakukan pengambilan data secara otomatis pada suatu platform spesifik. Kelas *scraping* dibagi per platform yang di-*scraping* dan meliputi kelas *ScrapingGooglePodcast*, *ScrapingListenNotes*, *ScrapingYoutube*, dan *ScrapingAnchor*. Sementara itu, kelas *Util* berisi beberapa fungsi pembantu untuk kelas-kelas *scraping* yang disebutkan sebelumnya, sedangkan kelas *FilterInappropriateContent* digunakan untuk melakukan penyaringan (*filter*) pada saat proses *scraping* agar konten yang tidak pantas tidak ikut dimasukkan ke dalam basis data.

4.3.2 Behavioural View

Berikut adalah *sequence diagram* untuk 4 dari 6 *use case*, yaitu UC-1: *Access audio list*, UC-2: *Play an audio content*, UC-4: *Add comment to an audio content*, dan UC-6: *Manage audio contents*.

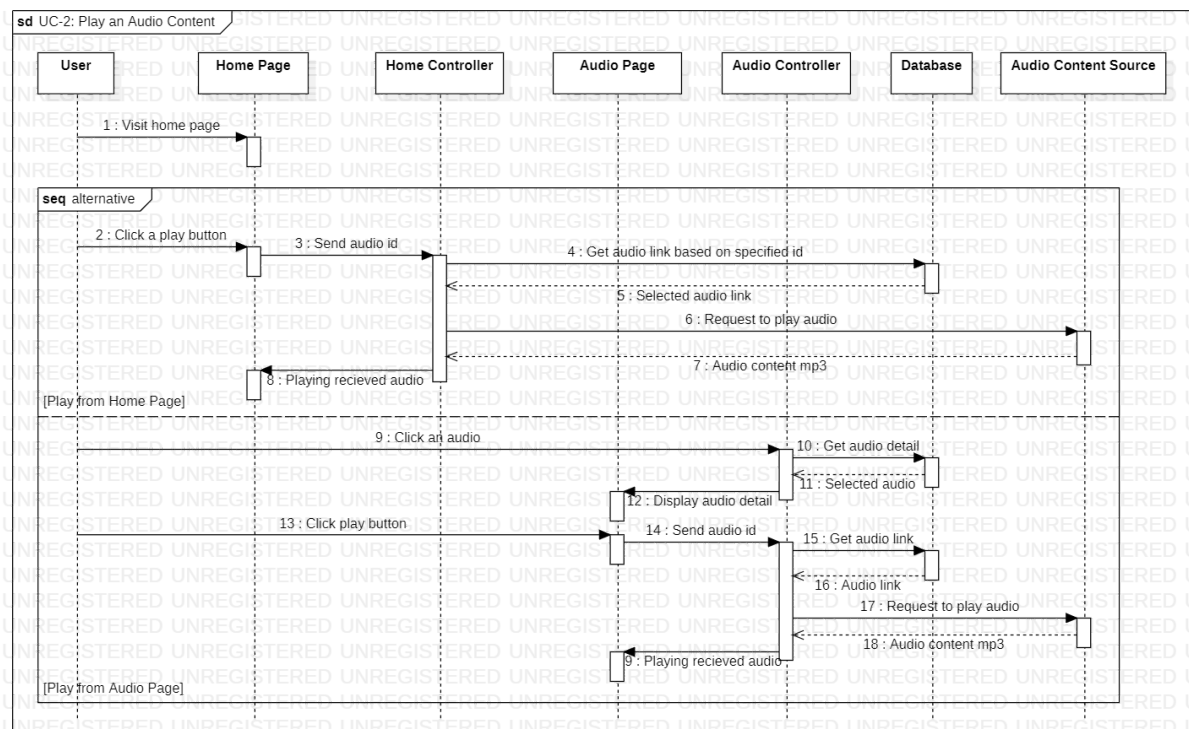
4.3.2.1 Sequence Diagram UC-1: Access Audio List



Gambar 4.3.3 Sequence diagram UC-1: Access Audio List

Berdasarkan *sequence diagram* di atas, alur *user* dalam mengakses konten audio adalah dengan mengunjungi laman utama (*home page*) terlebih dahulu. Kemudian, *user* dapat melakukan *filter* atau *searching* dan mengklik salah satu konten untuk melihat detailnya lebih lanjut.

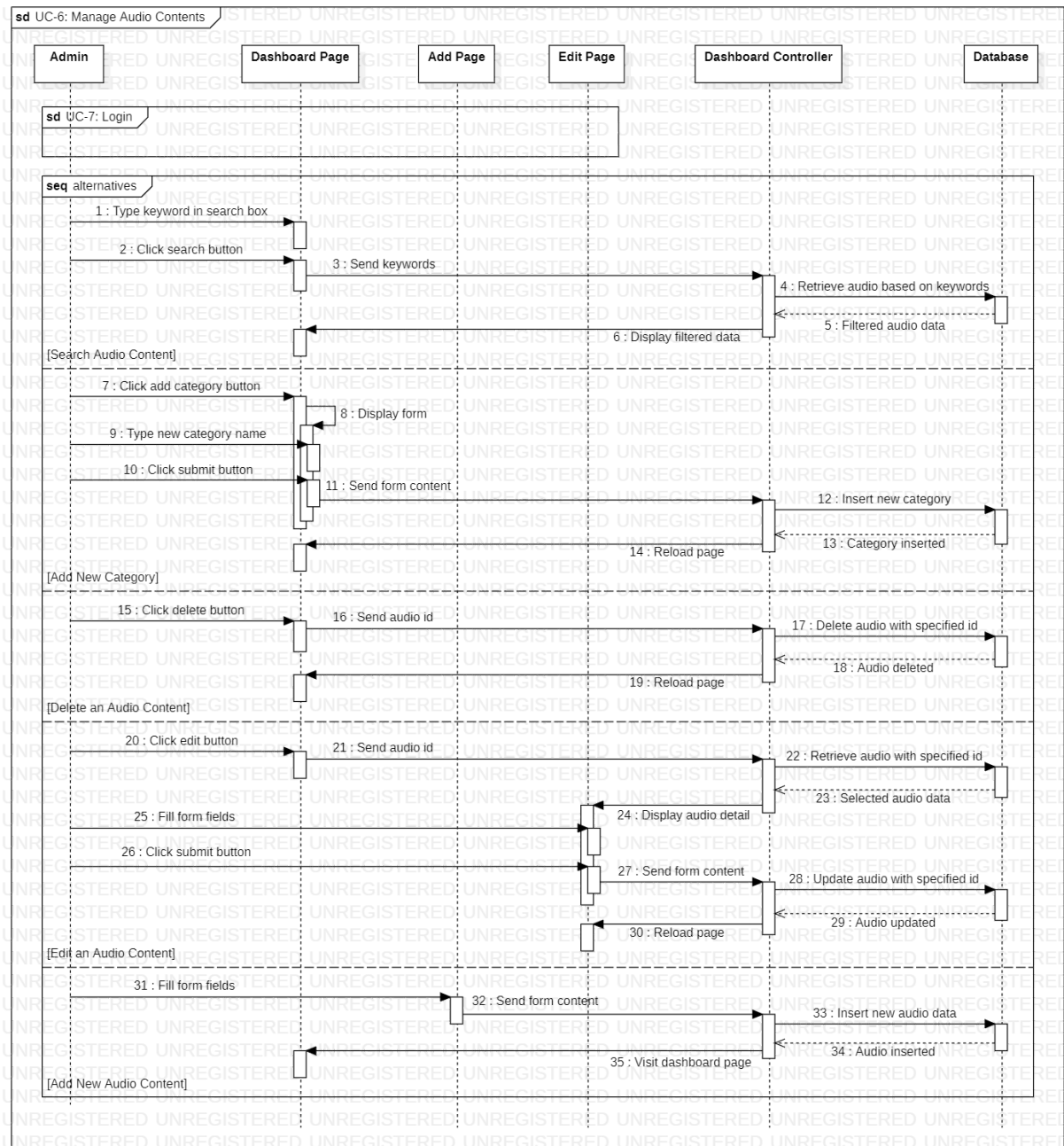
4.3.2.2 Sequence Diagram UC-2: Play an Audio Content



Gambar 4.3.4 Sequence diagram UC-2: Play an Audio Content

Berdasarkan *sequence diagram* tersebut, alur *user* untuk mendengarkan konten audio, dapat dibagi 2, yaitu secara langsung menekan tombol *play* di *home page* atau masuk ke laman detail audio (*audio page*) terlebih dahulu. Kemudian, baik *home controller*, maupun *audio controller*, akan membuat *request* untuk meng-embed *source* mp3 dari sumber aslinya ke *audio player* pada *website*.

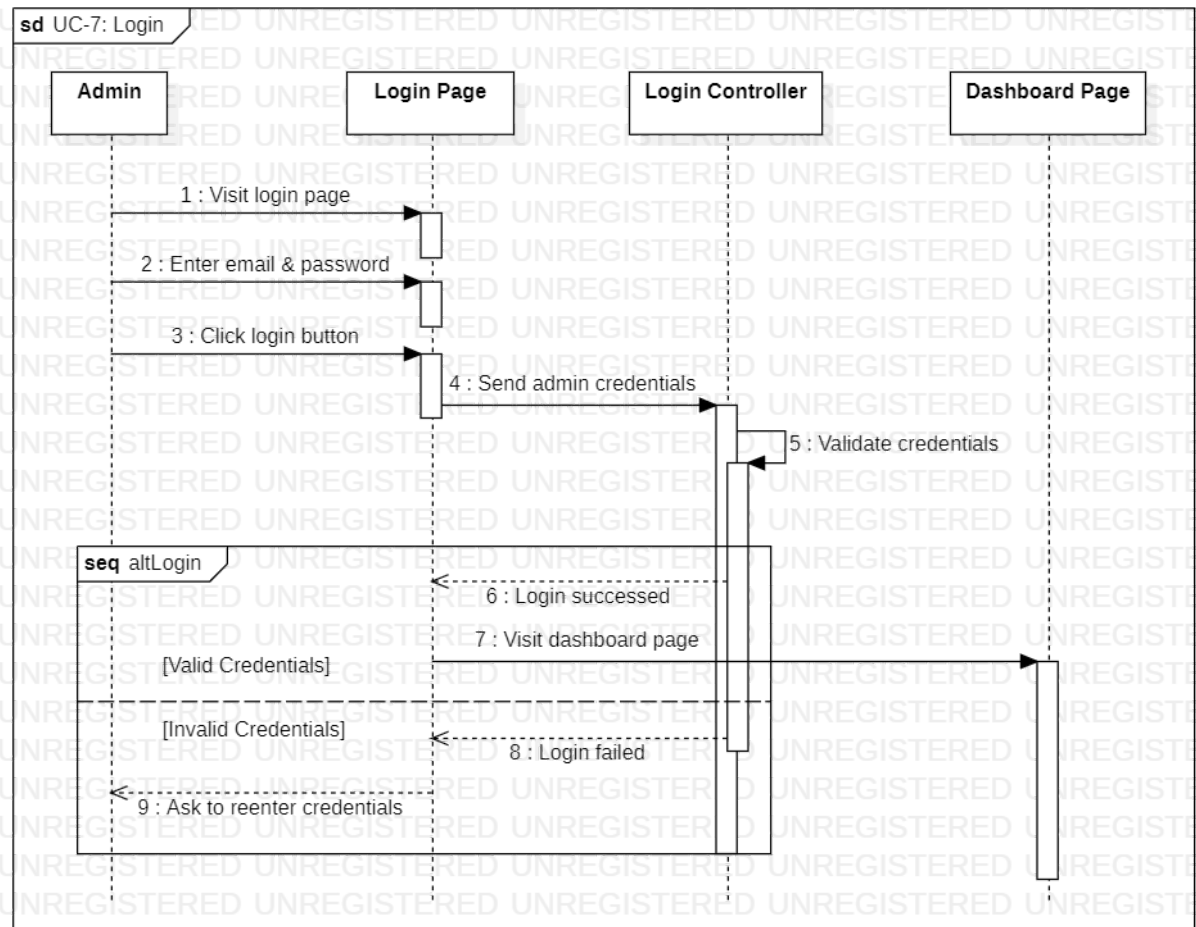
4.3.2.3 Sequence Diagram UC-6: Manage Audio Contents



Gambar 4.3.5 Sequence diagram UC-6: Manage Audio Contents

Berdasarkan *sequence diagram* tersebut, admin harus melakukan *login* terlebih dahulu untuk dapat mengakses fitur CRUD (*Create, Read, Update, Delete*) terhadap data kategori dan konten audio. Setelahnya, admin dapat melakukan operasi *searching* konten audio, menambahkan kategori, dan menghapus kategori melalui laman *dashboard*. Sedangkan untuk *edit* konten audio dapat dilakukan melalui laman *Edit Audio* dan menambahkan konten audio dapat dilakukan melalui laman *Add Audio*.

4.3.2.4 Sequence Diagram UC-7: Login

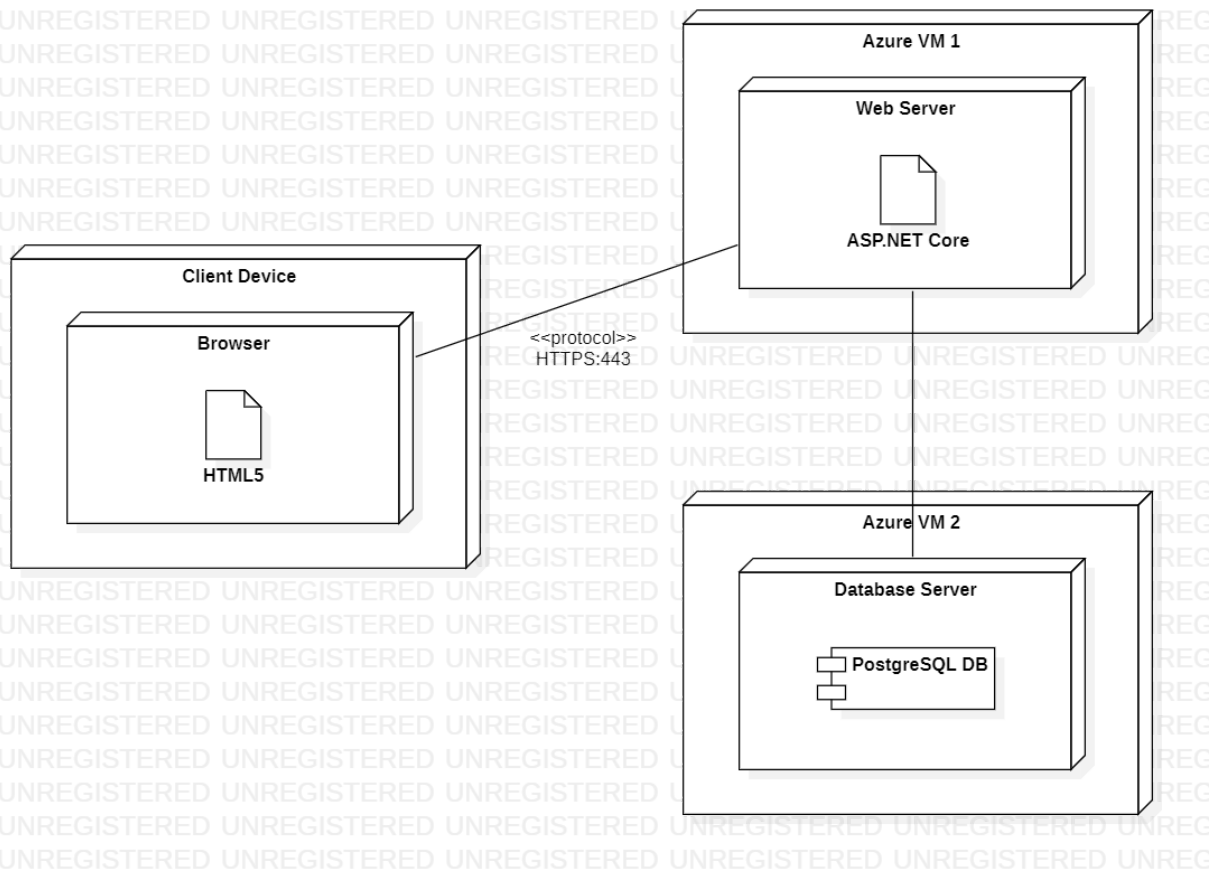


Gambar 4.3.6 Sequence diagram UC-7: Login

Sesuai diagram *sequence* di atas, untuk *login*, admin harus memasukkan data *credentials* berupa *email* dan *password* yang benar sehingga bisa dialihkan ke laman *dashboard* untuk melakukan operasi CRUD.

4.3.3 Deployment View

Website Bharabas Radio Curated App ini akan di-deploy pada 2 buah VM (*Virtual Machine*) terpisah dari Azure, yaitu VM 1 untuk Web Server berisi *source code website* dalam *framework* ASP.NET Core MVC dan VM 2 untuk *database* menggunakan PostgreSQL. *Client (User)*, baik admin maupun umum, akan bisa mengakses *website* ini dari *browser* masing-masing menggunakan protokol HTTPS (*port* 443) dengan url <https://bharabas.azurewebsites.net>. Nantinya *web server* tersebut akan berhubungan dengan *database server* untuk alur pertukaran data (operasi CRUD) yang dibutuhkan oleh *website* menggunakan modul Npgsql. Berikut adalah *deployment diagram* dari aplikasi *website* ini.



Gambar 4.3.7 Deployment diagram

5 Implementasi Perangkat Lunak

Berikut adalah struktur direktori dan daftar *file* hasil implementasi melalui *command* berikut:

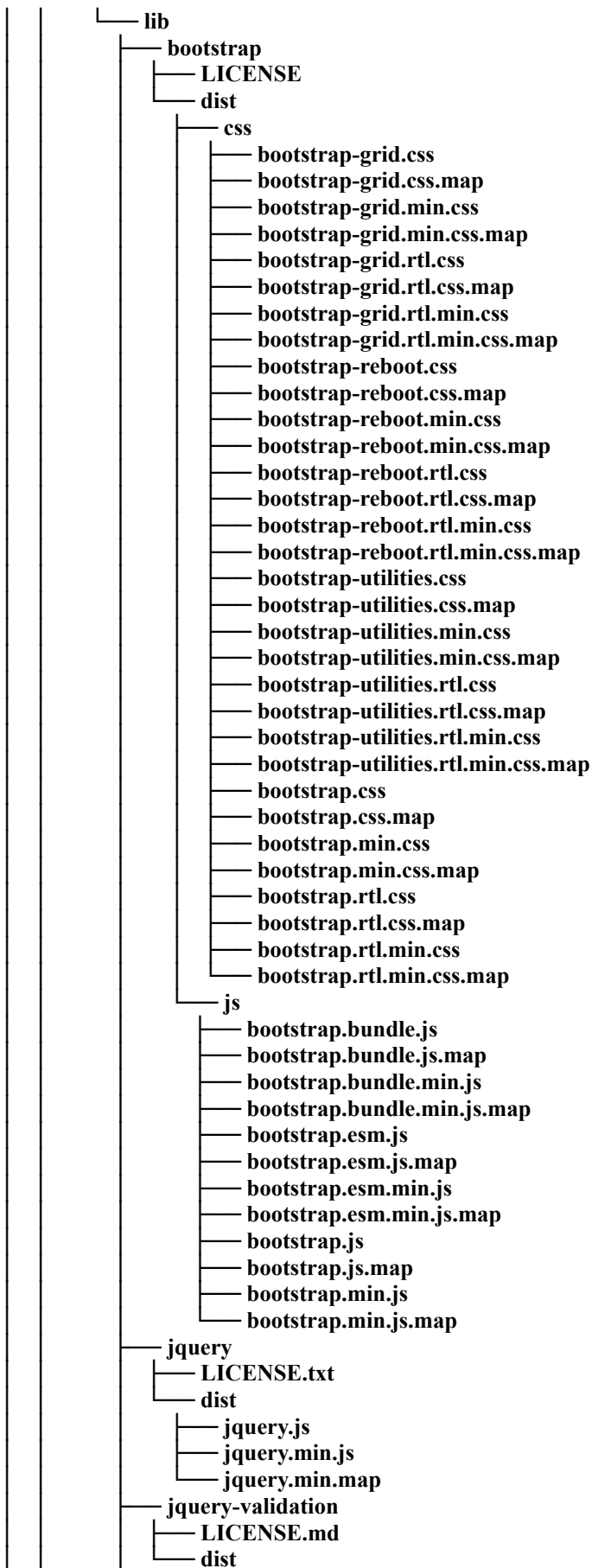
```
git ls-tree --full-tree --name-only -r HEAD | tree --fromfile
```

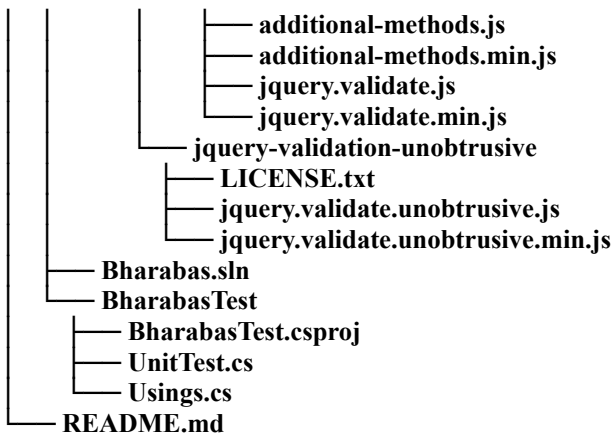
```

.
├── .gitignore
├── .gitlab-ci.yml
├── Bharabas
│   └── Bharabas
│       ├── Areas
│       │   └── Identity
│       │       ├── Data
│       │       │   ├── ApplicationUser.cs
│       │       │   └── AuthDbContext.cs
│       │       └── Pages
│       │           └── Account
│       │               ├── Login.cshtml
│       │               ├── Login.cshtml.cs
│       │               ├── Logout.cshtml
│       │               ├── Logout.cshtml.cs
│       │               ├── Register.cshtml
│       │               ├── Register.cshtml.cs
│       │               ├── RegisterConfirmation.cshtml
│       │               ├── RegisterConfirmation.cshtml.cs
│       │               └── _ViewImports.cshtml
  
```

	<ul style="list-style-type: none"> ValidationScriptsPartial.cshtml ViewImports.cshtml ViewStart.cshtml
	Bharabas.csproj
	Controllers
	<ul style="list-style-type: none"> AudioController.cs CommentController.cs DashboardController.cs HangfireAuthorizationFilter.cs HomeController.cs SharedController.cs
	Data
	<ul style="list-style-type: none"> ApplicationDbContext.cs
	Migrations
	<ul style="list-style-type: none"> 20230208165128_bharabasdbV1.Designer.cs 20230208165128_bharabasdbV1.cs 20230208165647_bharabasdbV2.Designer.cs 20230208165647_bharabasdbV2.cs 20230209091534_dateUploadedForAudio.Designer.cs 20230209091534_dateUploadedForAudio.cs 20230209093046_changeAudioTableName.Designer.cs 20230209093046_changeAudioTableName.cs 20230209093719_changeAudioTableNamev2.Designer.cs 20230209093719_changeAudioTableNamev2.cs 20230209101827_addImageUrlAttribute.Designer.cs 20230209101827_addImageUrlAttribute.cs 20230209110543_nullableSetting.Designer.cs 20230209110543_nullableSetting.cs 20230215063542_UpdateDatabase_150223.Designer.cs 20230215063542_UpdateDatabase_150223.cs 20230216044952_NamaMigrasi.Designer.cs 20230216044952_NamaMigrasi.cs 20230216120009_NamaMigrasi2.Designer.cs 20230216120009_NamaMigrasi2.cs 20230216163716_migrationRepair.Designer.cs 20230216163716_migrationRepair.cs 20230223124815_linkToPlatform.Designer.cs 20230223124815_linkToPlatform.cs ApplicationDbContextModelSnapshot.cs
	AuthDb
	<ul style="list-style-type: none"> 20230402222857_CreateIdentitySchema.Designer.cs 20230402222857_CreateIdentitySchema.cs AuthDbContextModelSnapshot.cs
	Models
	<ul style="list-style-type: none"> Audio.cs AudioCategoryRelation.cs Category.cs CommentLog.cs ErrorViewModel.cs LikeLog.cs Pager.cs ShareLog.cs
	Program.cs
	Properties
	<ul style="list-style-type: none"> ServiceDependencies <ul style="list-style-type: none"> bharabas - Zip Deploy <ul style="list-style-type: none"> profile.arm.json launchSettings.json
	Scraping

- FilterInappropriateContent.cs
- ScrapingAnchor.cs
- ScrapingGooglePodcast.cs
- ScrapingListenNotes.cs
- ScrapingRSS.cs
- ScrapingYoutube.cs
- Util.cs
- rssID.txt
- Views
 - Audio
 - AudioDetail.cshtml
 - Comment
 - SelectedAudioComment.cshtml
 - Dashboard
 - Add.cshtml
 - Edit.cshtml
 - Index.cshtml
 - RemoveComment.cshtml
 - ScrapeAudio.cshtml
 - Home
 - Index.cshtml
 - Privacy.cshtml
 - Search.cshtml
 - Shared
 - Error.cshtml
 - _Layout.cshtml
 - _Layout.cshtml.css
 - _LayoutDashboard.cshtml
 - _LoginPartial.cshtml
 - _NavCategory.cshtml
 - _ValidationScriptsPartial.cshtml
 - _ViewImports.cshtml
 - _ViewStart.cshtml
- appsettings.Development.json
- appsettings.example.json
- wwwroot
 - assets
 - Spotify for Podcasters.png
 - Spotify-logo.png
 - badwords.txt
 - bharabas-logo.png
 - google-podcasts-logo.png
 - image-not-found.png
 - listen-notes-logo.png
 - youtube-logo.png
 - css
 - audio-detail.css
 - dashboard.css
 - player.css
 - site.css
 - theme.css
 - favicon.ico
 - js
 - categoryList.js
 - dashboardAdd.js
 - dashboardEdit.js
 - detail.js
 - home.js
 - site.js





37 directories, 174 files

Berikut adalah tabel *traceability* dari setiap *file* pada struktur direktori di atas:

<i>Product Backlog</i>	Elemen Kelas	Pengujian	Daftar <i>File</i>
PB-01	ScrapingListenNotes, ScrapingGooglePodcast, ScrapingYoutube, ScrapingAnchor, ScrapingRSS, Util, FilterInappropriateContent	Automated Unit Testing, Manual Unit Testing: Scraping Konten Audio	ScrapingListenNotes.cs, ScrapingGooglePodcast.cs, ScrapingYoutube.cs, ScrapingAnchor.cs, ScrapingRSS.cs, Util.cs, FilterInappropriateContent.cs
PB-02	Audio, Category, AudioCategoryRelation, ApplicationDbContext	Manual Unit Testing: Filter Konten Audio	Audio.cs, Category.cs, AudioCategoryRelation.cs, ApplicationDbContext.cs
PB-03	HomeController, AudioController, ApplicationDbContext, Audio	Manual Unit Testing: Mendengarkan Konten Audio	HomeController.cs, AudioController.cs, ApplicationDbContext.cs, Audio.cs
PB-04	HomeController, ApplicationDbContext, Audio	Manual Unit Testing: Pencarian Konten Audio	HomeController.cs, ApplicationDbContext.cs, Audio.cs
PB-05	CommentController, LikeLog, CommentLog, Audio	*	CommentController.cs, LikeLog.cs, CommentLog.cs, Audio.cs
PB-06	ShareLog, Audio	*	ShareLog.cs, Audio.cs
PB-07	HomeController, AudioController, ApplicationDbContext, Audio	Manual Unit Testing: Pencarian Konten Audio	HomeController.cs, AudioController.cs, ApplicationDbContext.cs, Audio.cs
PB-08	-	-	-
PB-09	DashboardController	Manual Unit Testing: Mengelola Konten Audio	DashboardController.cs

Keterangan:

- PB-05 dan PB-06 dibatalkan sesuai kesepakatan bersama *product owner*, sehingga tidak ada pengujian

6 Pengujian Perangkat Lunak

Jenis *testing* yang digunakan pada pembuatan *software* dibagi menjadi dua yaitu *unit testing* yang dilakukan secara otomatis dan *unit testing* yang dilakukan secara manual. Selain itu, kami juga melakukan pengujian kualitas kode secara manual menggunakan kakas *open source*.

6.1 Automated Unit Testing

Unit testing otomatis dilakukan dengan melakukan pengetesan per fitur dari bagian *back-end* dari aplikasi. Pengujian dilakukan dengan menggunakan XUnit *testing* yang disediakan pada .NET *framework*. Adapun fitur yang dilakukan pengetesan secara otomatis meliputi fitur *searching* untuk memastikan diperoleh hasil *search* yang valid, fitur *scraping* untuk memastikan fungsi utilitas yang digunakan telah sesuai, fitur filter berdasarkan kategori untuk memastikan data kategori yang diambil lengkap, dan pagination untuk memastikan fungsi utilitas berjalan dengan baik

6.2 Manual Unit Testing

Unit testing manual dilakukan pada *software* setiap akhir sprint. *Testing* ini dilakukan pada 2 *user* utama yaitu *user* umum dan *user* admin. *Unit testing* manual sendiri dilakukan pada setiap fitur yang dibuat dan dikatakan berhasil bila seluruh kasus *testing* pada seluruh fitur berhasil.

6.2.1 Scraping Konten Audio

No.	Kasus Pengujian	Langkah-Langkah Pengujian	Hasil Pengujian
1	Admin melakukan <i>scraping</i> konten audio pada platform ListenNotes secara manual melalui laman <i>scrape audio</i>	<ul style="list-style-type: none">- Admin membuka laman <i>dashboard</i> dengan mengetikkan https://bharabas.azurewebsites.net/dashboard pada <i>browser</i>- Admin membuka laman <i>scraping</i> dengan menekan menu <i>scrape audio</i> pada <i>navigation bar</i>- Admin menekan tombol <i>Scrape</i> pada <i>section</i> ListenNotes untuk memulai proses <i>scraping</i> data	Aplikasi berhasil melakukan <i>scraping</i> data konten audio dari platform ListenNotes dan menyimpannya dalam <i>database</i>
2	Admin melakukan <i>scraping</i> konten audio pada platform Google Podcast secara manual melalui laman <i>scrape audio</i>	<ul style="list-style-type: none">- Admin membuka laman <i>dashboard</i> dengan mengetikkan https://bharabas.azurewebsites.net/dashboard pada <i>browser</i>- Admin membuka laman <i>scraping</i> dengan menekan menu <i>scrape audio</i> pada <i>navigation bar</i>- Admin menekan tombol <i>Scrape</i> pada <i>section</i> Google Podcast untuk memulai proses <i>scraping</i> data	Aplikasi berhasil melakukan <i>scraping</i> data konten audio dari platform Google Podcast dan menyimpannya dalam <i>database</i>
3	Admin melakukan <i>scraping</i> konten audio pada platform YouTube secara manual melalui laman <i>scrape audio</i>	<ul style="list-style-type: none">- Admin membuka laman <i>dashboard</i> dengan mengetikkan https://bharabas.azurewebsites.net/dashboard pada <i>browser</i>- Admin membuka laman <i>scraping</i>	Aplikasi berhasil melakukan <i>scraping</i> data konten audio dari platform YouTube dan menyimpannya dalam

		<p>dengan menekan menu <i>scrape audio</i> pada <i>navigation bar</i></p> <ul style="list-style-type: none"> - Admin menekan tombol <i>Scrape</i> pada <i>section</i> YouTube untuk memulai proses <i>scraping</i> data 	<i>database</i>
4	Admin melakukan <i>scraping</i> konten audio pada platform Spotify secara manual melalui laman <i>scrape audio</i>	<ul style="list-style-type: none"> - Admin membuka laman <i>dashboard</i> dengan mengetikkan https://bharabas.azurewebsites.net/dashboard pada <i>browser</i> - Admin membuka laman <i>scraping</i> dengan menekan menu <i>scrape audio</i> pada <i>navigation bar</i> - Admin menekan tombol <i>Scrape</i> pada <i>section</i> Spotify untuk memulai proses <i>scraping</i> data 	Aplikasi berhasil melakukan <i>scraping</i> data konten audio dari platform Spotify dan menyimpannya dalam <i>database</i>
5	Admin mengaktifkan fitur untuk melakukan <i>scraping</i> konten audio secara periodik dan otomatis	<ul style="list-style-type: none"> - Admin membuka laman <i>dashboard</i> dengan mengetikkan https://bharabas.azurewebsites.net/dashboard pada <i>browser</i> - Admin membuka laman <i>scraping</i> dengan menekan menu <i>scrape audio</i> pada <i>navigation bar</i> - Admin mengisi <i>field timer</i> untuk menentukan periode aplikasi akan melakukan <i>scraping</i> kembali - Admin mengisi <i>field</i> waktu pertama kali <i>scraping</i> untuk menentukan kapan aplikasi akan memulai <i>scraping</i> untuk pertama kali - Admin menekan tombol <i>Submit</i> untuk mengaktifkan fitur ini 	Aplikasi berhasil melakukan <i>scraping</i> data konten audio secara otomatis sesuai periode yang ditentukan dan menyimpannya dalam <i>database</i>
6	Admin melakukan <i>scraping</i> melalui url RSS tertentu	<ul style="list-style-type: none"> - Admin membuka laman <i>dashboard</i> dengan mengetikkan https://bharabas.azurewebsites.net/dashboard pada <i>browser</i> - Admin membuka laman <i>scraping</i> dengan menekan menu <i>scrape audio</i> pada <i>navigation bar</i> - Admin mengisi <i>field</i> RSS url yang akan digunakan sebagai sumber <i>scraping</i> - Admin menekan tombol <i>Scrape</i> pada <i>section</i> <i>Add Audio from RSS</i> untuk memulai proses <i>scraping</i> 	Aplikasi berhasil melakukan <i>scraping</i> data konten audio dari url RSS yang diberikan dan menyimpannya dalam <i>database</i>

6.2.2 Mendengarkan Konten Audio

No.	Kasus Pengujian	Langkah-Langkah Pengujian	Hasil Pengujian
1	User mendengarkan audio dalam halaman <i>home</i>	<ul style="list-style-type: none"> - User membuka halaman <i>home</i> - User menekan satu kali pada audio yang ingin dimainkan - User dapat mengontrol audio, seperti 	Aplikasi berhasil memainkan audio dan audio tersebut dapat dikontrol

		<i>pause, play, dan mengubah slider audio</i>	
2.	<i>User</i> mendengarkan audio dalam halaman detail audio	<ul style="list-style-type: none"> - <i>User</i> membuka halaman <i>home</i> - <i>User</i> menekan dua kali pada audio yang ingin dimainkan untuk membuka halaman detail lagu - <i>User</i> menekan satu kali pada tombol <i>play</i> - <i>User</i> dapat mengontrol audio, seperti <i>pause, play, dan mengubah slider audio</i> 	Aplikasi berhasil memainkan audio dan audio tersebut dapat dikontrol

6.2.3 Filter Konten Audio

No.	Kasus Pengujian	Langkah-Langkah Pengujian	Hasil Pengujian
1	<i>User</i> memfilter audio berdasarkan <i>category</i>	<ul style="list-style-type: none"> - <i>User</i> membuka halaman <i>home</i> - <i>User</i> memilih kategori yang ingin dilihat pada <i>sidebar</i> 	Aplikasi berhasil menunjukkan audio dengan kategori yang dipilih

6.2.4 Pencarian Konten Audio

No.	Kasus Pengujian	Langkah-Langkah Pengujian	Hasil Pengujian
1	<i>User</i> mencari audio dengan <i>search bar</i>	<ul style="list-style-type: none"> - <i>User</i> membuka halaman <i>home</i> - <i>User</i> memasukkan <i>search key</i> pada <i>search bar</i> - <i>User</i> menekan tombol <i>search</i> 	Aplikasi berhasil menunjukkan audio dengan <i>search key</i> yang sama

6.2.5 Mengelola Konten Audio

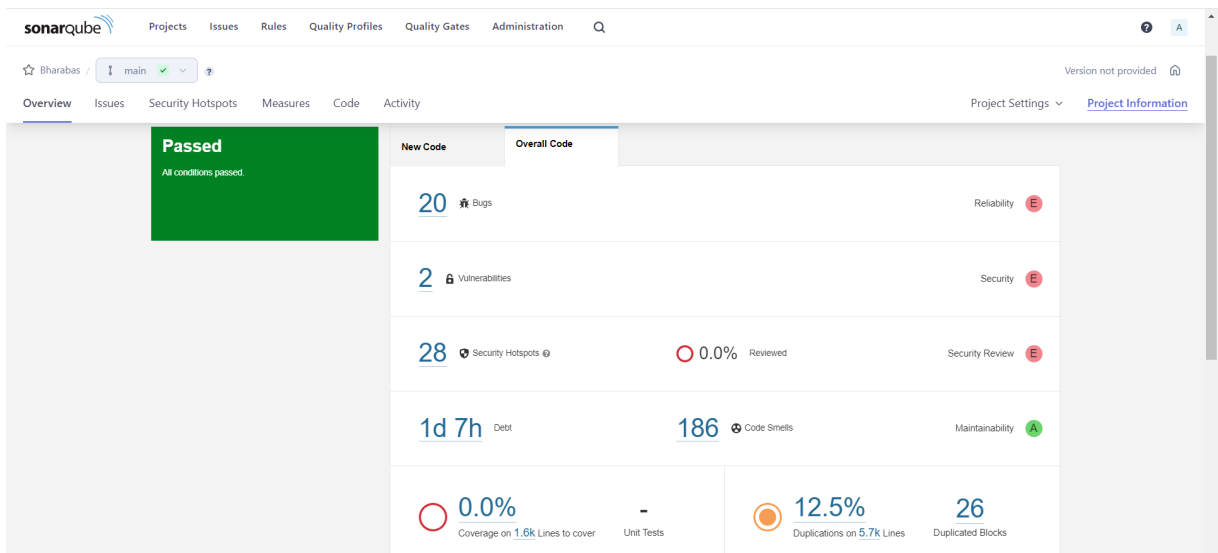
No.	Kasus Pengujian	Langkah-Langkah Pengujian	Hasil Pengujian
1	Admin melakukan <i>login</i> dengan <i>credential</i> (<i>email</i> dan <i>password</i>) yang valid	<ul style="list-style-type: none"> - Admin membuka laman <i>dashboard</i> dengan mengetikkan https://bharabas.azurewebsites.net/dashboard pada <i>browser</i> - Admin akan dialihkan ke laman <i>login</i> untuk proses autentikasi (https://bharabas.azurewebsites.net/Identity/Account/Login?ReturnUrl=%2Fdashboard) - Admin memasukkan <i>email</i> “bharabaspodcast@gmail.com” - Admin memasukkan <i>password</i> “proyekpodcast!” - Admin menekan tombol <i>Login</i> 	Admin berhasil melakukan <i>login</i> dan dialihkan ke laman <i>dashboard</i>

2	Admin melakukan <i>login</i> dengan <i>credential</i> (<i>email</i> dan <i>password</i>) yang tidak valid	<ul style="list-style-type: none"> - Admin membuka laman <i>dashboard</i> dengan mengetikkan https://bharabas.azurewebsites.net/Dashboard pada <i>browser</i> - Admin akan dialihkan ke laman <i>login</i> untuk proses autentikasi (https://bharabas.azurewebsites.net/Identity/Account/Login?ReturnUrl=%2Fdashboard) - Admin memasukkan <i>email</i> dan/atau <i>password</i> yang salah - Admin menekan tombol <i>Login</i> 	Admin gagal melakukan <i>login</i> dan menerima pesan kesalahan “ <i>Invalid login attempt</i> ”
3	Admin melakukan <i>logout</i>	<ul style="list-style-type: none"> - Admin sudah melakukan <i>login</i> - Admin menekan tombol <i>Logout</i> 	Admin berhasil melakukan <i>logout</i> dan dialihkan ke laman utama/ <i>home</i> (https://bharabas.azurewebsites.net/)
4	Admin melakukan pencarian pada audio di laman <i>dashboard</i>	<ul style="list-style-type: none"> - Admin melakukan <i>login</i> - Admin membuka laman <i>dashboard</i> - Admin mengetikkan <i>keyword</i> pencarian pada <i>search box</i> - Admin menekan tombol <i>Search</i> 	Aplikasi berhasil menampilkan daftar konten yang sesuai dengan <i>keyword</i> tersebut
5	Admin menambahkan kategori baru	<ul style="list-style-type: none"> - Admin melakukan <i>login</i> - Admin membuka laman <i>dashboard</i> - Admin menekan tombol <i>Add Category</i> untuk memunculkan <i>dialog box</i> berisi <i>form</i> untuk menambahkan kategori - Admin mengetikkan nama kategori yang ingin ditambahkan pada kolom <i>Category Name</i> - Admin menekan tombol <i>Submit</i> 	Aplikasi berhasil menambahkan kategori ke dalam <i>database</i> dan Admin dialihkan ke laman <i>dashboard</i> kembali
6	Admin meng- <i>edit</i> detail salah satu konten audio	<ul style="list-style-type: none"> - Admin melakukan <i>login</i> - Admin membuka laman <i>dashboard</i> - Admin menekan tombol <i>Edit</i> pada salah satu konten audio untuk menampilkan laman berisi <i>form</i> untuk meng-<i>edit</i> konten - Admin mengisi <i>field</i> pada kolom-kolom yang tersedia - Admin menekan tombol <i>Submit</i> 	Aplikasi berhasil mengubah detail konten audio pada <i>database</i> berdasarkan isi <i>form</i> dan Admin dialihkan kembali (<i>reload</i>) ke laman yang sama.
7	Admin menghapus salah satu konten audio	<ul style="list-style-type: none"> - Admin melakukan <i>login</i> - Admin membuka laman <i>dashboard</i> - Admin menekan tombol <i>Delete</i> pada salah satu konten audio 	Aplikasi berhasil menghapus konten audio yang dihapus pada <i>database</i> dan Admin dialihkan kembali (<i>reload</i>) ke laman <i>dashboard</i>
8	Admin menambahkan konten audio baru	<ul style="list-style-type: none"> - Admin melakukan <i>login</i> - Admin membuka laman <i>add audio</i> - Admin mengisi semua <i>field</i> pada 	Aplikasi berhasil menambahkan konten audio baru pada <i>database</i>

		<i>form</i> - Admin menekan tombol <i>Submit</i>	berdasarkan isi <i>form</i> dan Admin dialihkan kembali (<i>reload</i>) ke laman <i>dashboard</i>
--	--	---	---

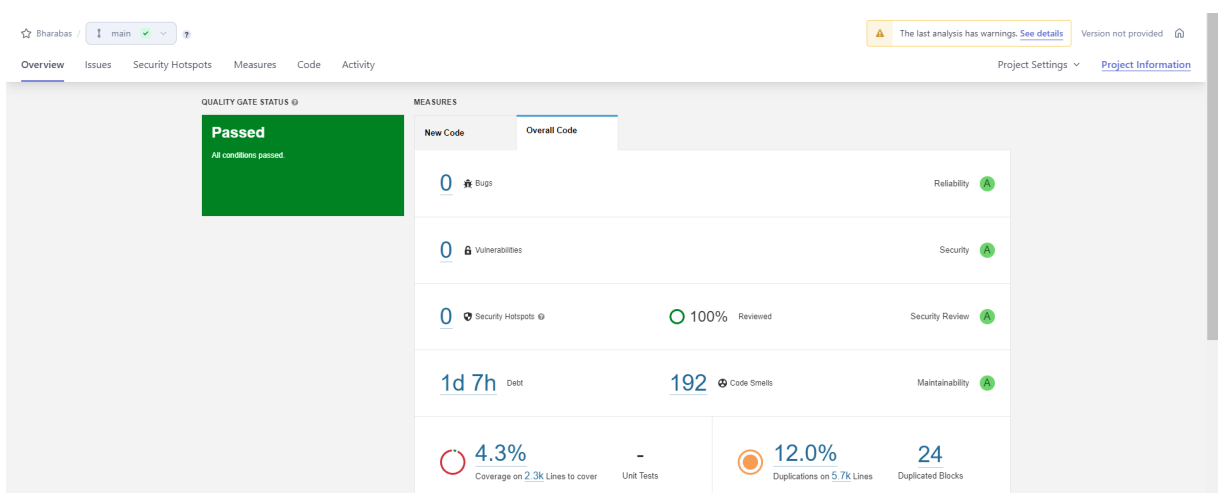
6.3 Quality Code Testing

Pengujian kualitas kode dilakukan dengan kakas SonarQube versi *open-source*. Dengan kakas ini, didapat pengukuran dan daftar isu-isu mengenai *bugs*, *vulnerability*, *security hotspots*, *code smells*, *coverage*, dan *duplicate*.



Gambar 6.3.1 Hasil *quality code testing* 1

Kami mendapat *rating* E untuk *reliability*, *security*, dan *security review*. Untuk *reliability*, setelah kami memperbaiki potensi *blocker bug*, menambah deskripsi gambar, serta mengonfirmasi potensi *bug* yang ditemukan, kami mendapat *rating* A. Untuk *security*, kami memindahkan dan membuat ulang beberapa kredensial *hard-code* ke *appsettings.json* agar lebih aman bila *repository* dibuat publik dan mendapat *rating* A. Untuk *security review*, kami me-review semua *security hotspot* dan mendapat *rating* A. Sehingga didapat hasil seperti di bawah.



Gambar 6.3.2 Hasil *quality code testing* 2

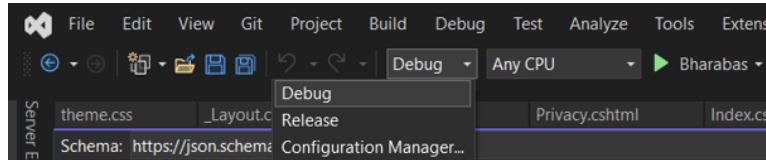
7 Petunjuk Instalasi Perangkat Lunak

Perangkat lunak telah dilakukan *deploy* pada <https://bharabas.azurewebsites.net/>. Untuk *file code* bisa *clone* dari *repo* https://gitlab.informatika.org/if3250_2023_k01_08_bharabas/if3250_2023_k01_08_bharabas_backend.git.

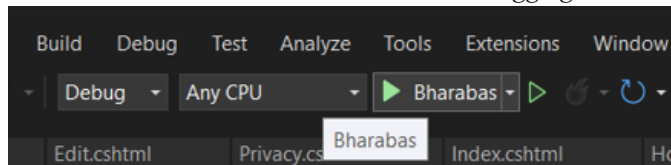
Melakukan Debug

Berikut merupakan cara melakukan *debug* dan *build code project* menggunakan IDE Visual Studio

1. Buka *file project* Bharabas
2. Pada bagian atas, pastikan Visual Studio berjalan pada mode *debug*



3. Klik tombol “Bharabas” untuk melakukan *debugging*

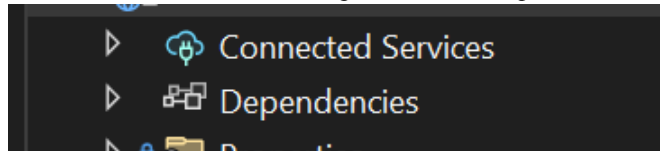


4. Tunggu hingga proses *build* selesai
5. Jika proses *build* telah selesai, perangkat lunak akan terbuka pada *browser* dan Anda dapat melakukan *debug*

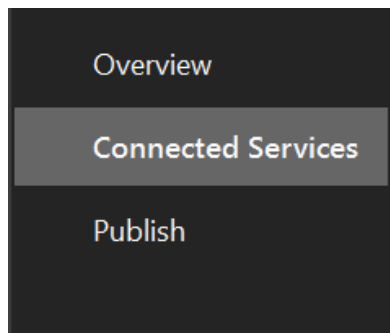
Melakukan Deployment

Berikut merupakan cara melakukan *deployment* pada Azure dengan menggunakan IDE Visual Studio

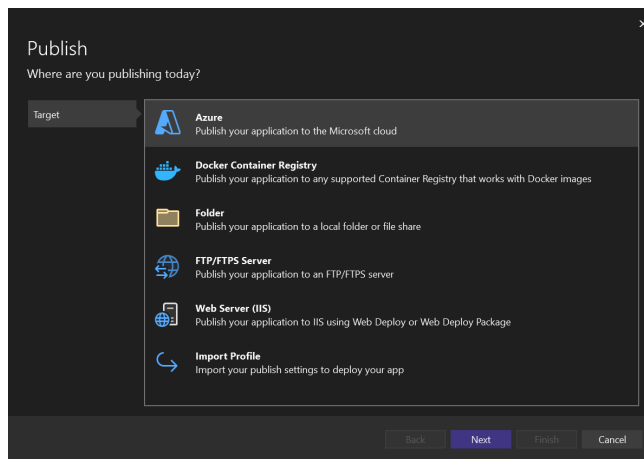
1. Buka *file project* Bharabas
2. Klik menu *Connected Services* pada *Solution Explorer*



3. Pilih menu *Publish*



4. Pilih target Azure kemudian isi data yang diperlukan



5. Membuat App service pada Azure dengan target berupa .NET 7 dan Operating System dengan Linux

Microsoft Azure Search resources, services, and docs (G+)

Home > App Services >

Create Web App

Basics Deployment Networking Monitoring Tags Review + create

App Service Web Apps lets you quickly build, deploy, and scale enterprise-grade web, mobile, and API apps running on any platform. Meet rigorous performance, scalability, security and compliance requirements while using a fully managed platform to perform infrastructure maintenance. [Learn more](#)

Project Details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * Azure for Students

Resource Group * (New) Resource group [Create new](#)

Instance Details

Need a database? [Try the new Web + Database experience.](#)

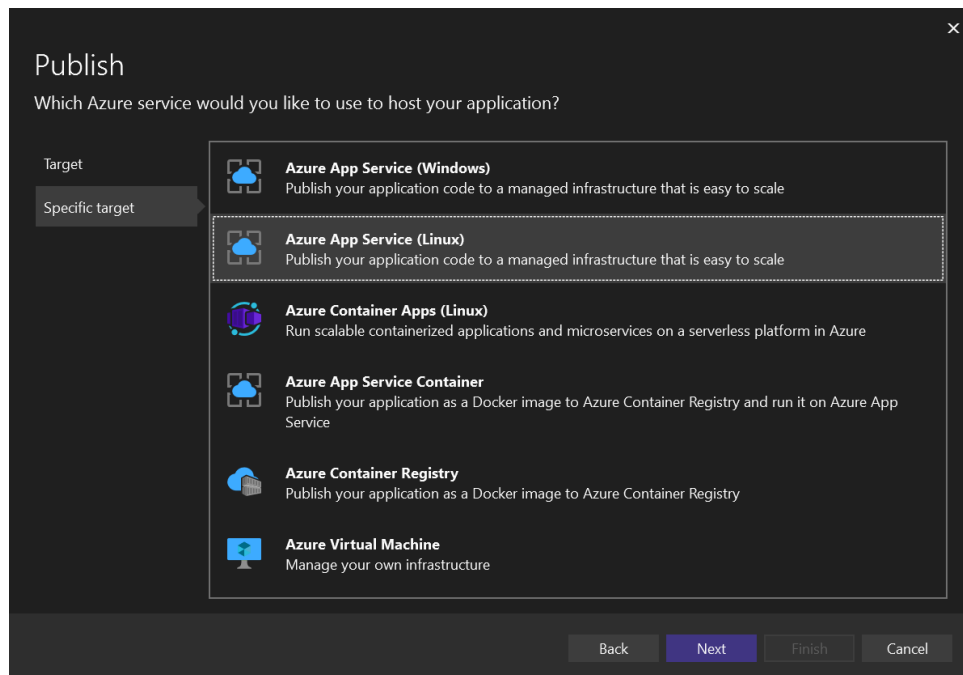
Name * Web App name. .azurewebsites.net

Publish * ☒ Code ☐ Docker Container ☐ Static Web App

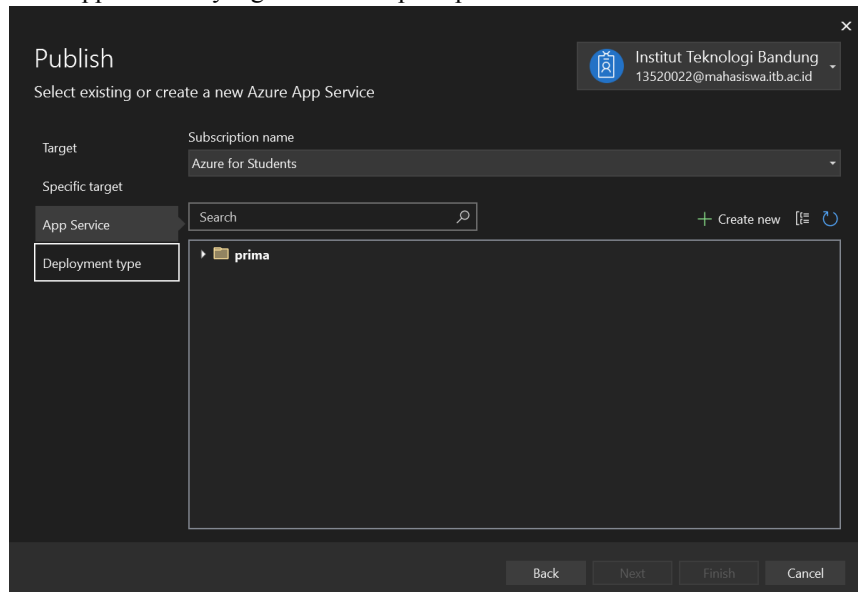
Runtime stack * .NET 7 (STS)

Operating System * ☒ Linux ☐ Windows

6. Pilih *Specific Target* berupa Azure App Service (Linux)



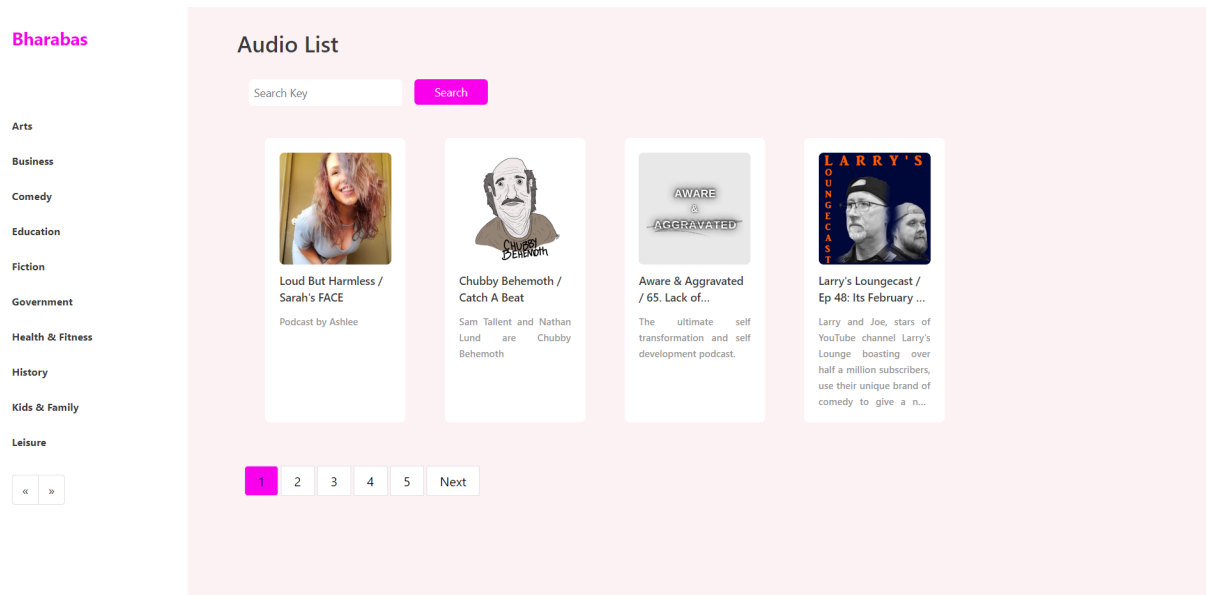
7. Pilih App Services yang baru dibuat pada poin 5 lalu tekan tombol *Finish*



8. Website telah berhasil di-*deploy*

8 Panduan Penggunaan Perangkat Lunak

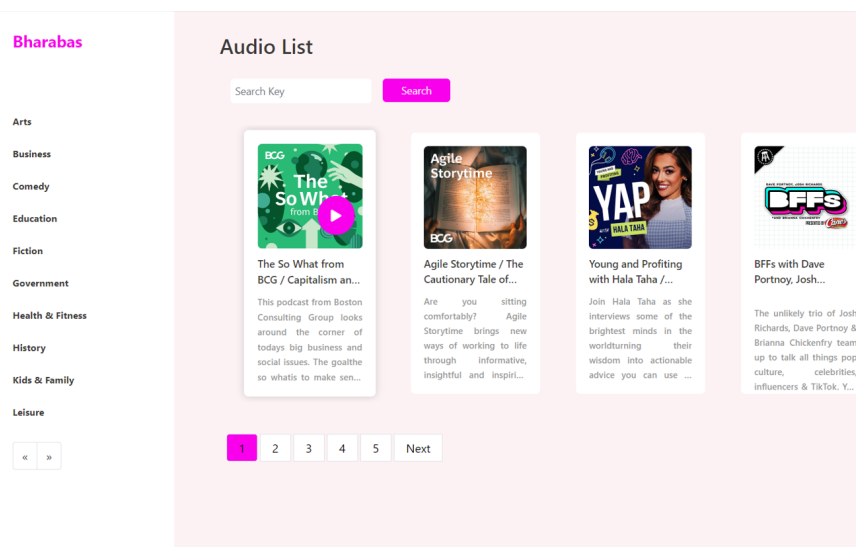
8.1 Halaman Utama



Gambar 8.1.1 Tampilan halaman utama

Pada halaman utama *website* Bharabas, pengguna akan melihat tampilan seperti gambar di atas. Pada bagian kiri, terdapat *navigation bar* yang terdiri atas nama *website* yaitu “Bharabas” dan daftar dari kategori konten audio yang dapat dipilih oleh pengguna. Pada halaman utama juga ditampilkan *search bar* untuk mencari lagu berdasarkan judul, daftar konten audio yang dapat didengar, serta *page list* untuk melihat daftar konten audio yang lainnya.

- Jika pengguna ingin memilih suatu kategori tertentu dapat melakukan klik pada kategori yang diinginkan pada *navbar* dan akan dialihkan ke halaman **Pemilihan Berdasarkan Kategori (8.2.1)**.
- Jika pengguna ingin mencari suatu konten audio dapat mengisi *input* pada bagian “Search Key” kemudian klik “Search”. Pengguna akan dialihkan ke halaman **Pencarian Berdasarkan Input Pengguna (8.2.2)**.
- Jika pengguna ingin memilih konten audio yang diinginkan, dapat melakukan klik pada judul audio yang diinginkan dan akan dialihkan ke halaman **Detail Audio**.

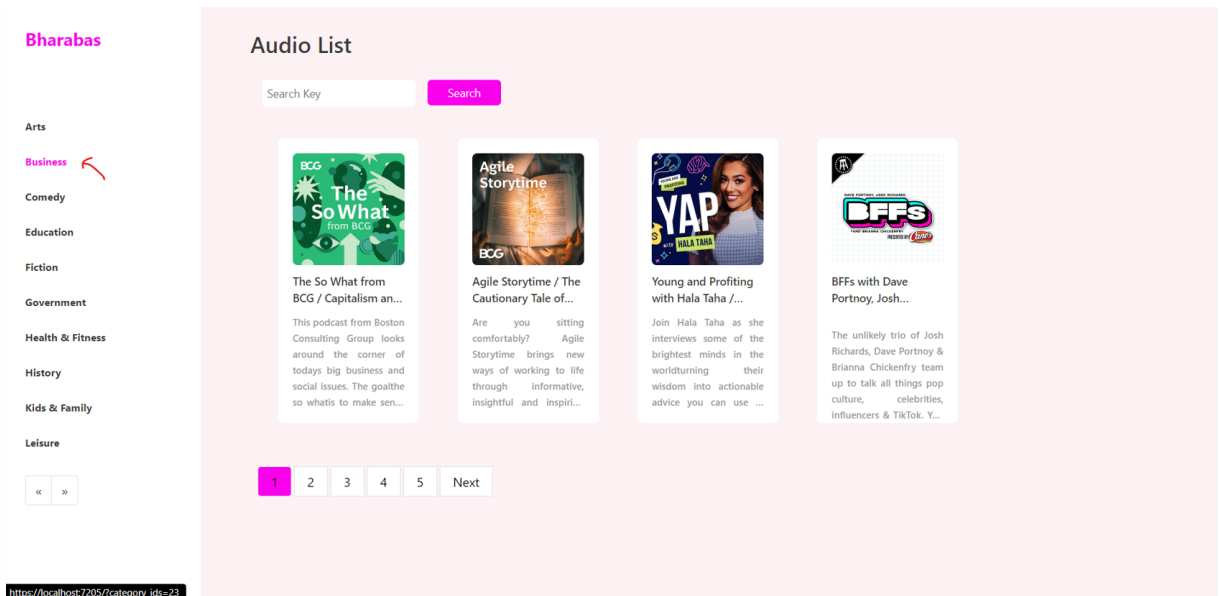


Gambar 8.1.2 Tampilan tombol *play* halaman utama

- Jika pengguna ingin memutar konten audio yang diinginkan, pengguna dapat mengarahkan *cursor* pada audio yang diinginkan, lalu melakukan klik pada logo *play* yang muncul. Pengguna akan dialihkan ke halaman **Pemutaran Konten Audio (8.2.3)**.

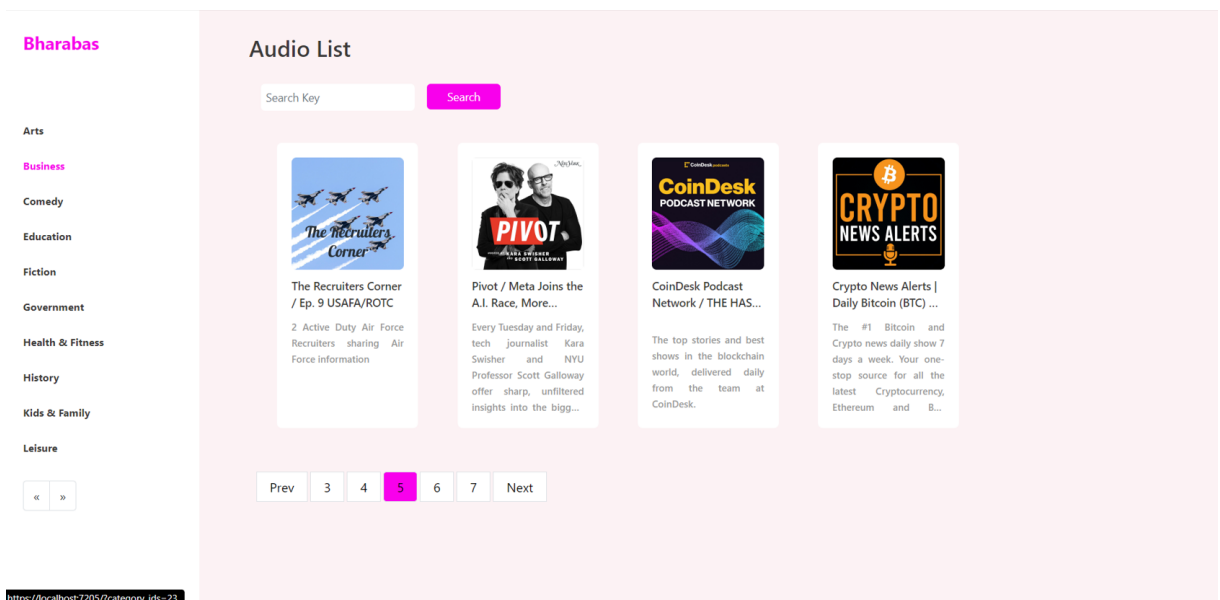
8.2 Halaman Daftar Konten Audio

8.2.1 Pemilihan Berdasarkan Kategori



Gambar 8.2.1 Tampilan halaman pemilihan berdasarkan kategori

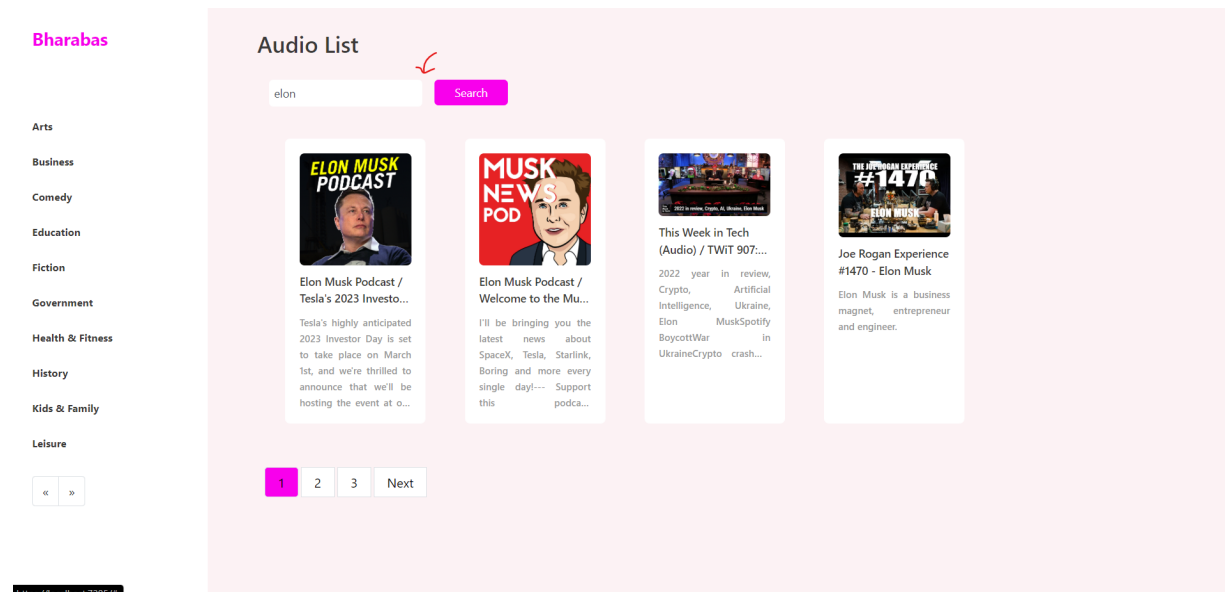
Pengguna dapat melakukan *filtering* konten audio berdasarkan kategori yang ada dengan melakukan klik pada daftar kategori yang tersedia pada *navbar*. Untuk melihat daftar kategori yang lebih banyak, pengguna dapat melakukan klik pada tombol panah yang berada pada bagian bawah daftar kategori tersebut.



Gambar 8.2.2 Tampilan pagination

Jika pengguna ingin melihat daftar konten audio yang lainnya, dapat melakukan ke *page* yang lain dengan memilih nomor *page* yang ada pada bagian bawah konten audio.

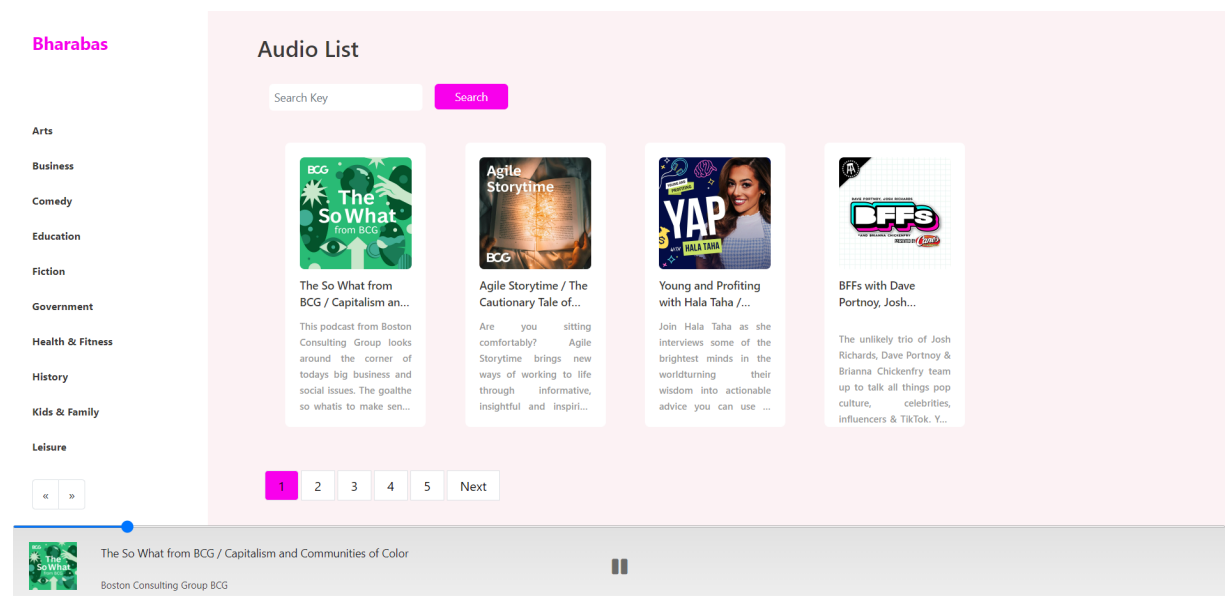
8.2.2 Pencarian Berdasarkan *Input* Pengguna



Gambar 8.2.3 Tampilan halaman pencarian berdasarkan *input* pengguna

Pengguna juga dapat melakukan pencarian berdasarkan judul yang di-*input* dari pengguna. Pengguna dapat mengisi pada bagian “*Search Bar*” lalu klik tombol “*Search*”. Halaman akan menampilkan konten audio yang judulnya mengandung berdasarkan *input* pengguna.

8.2.3 Pemutaran Konten Audio

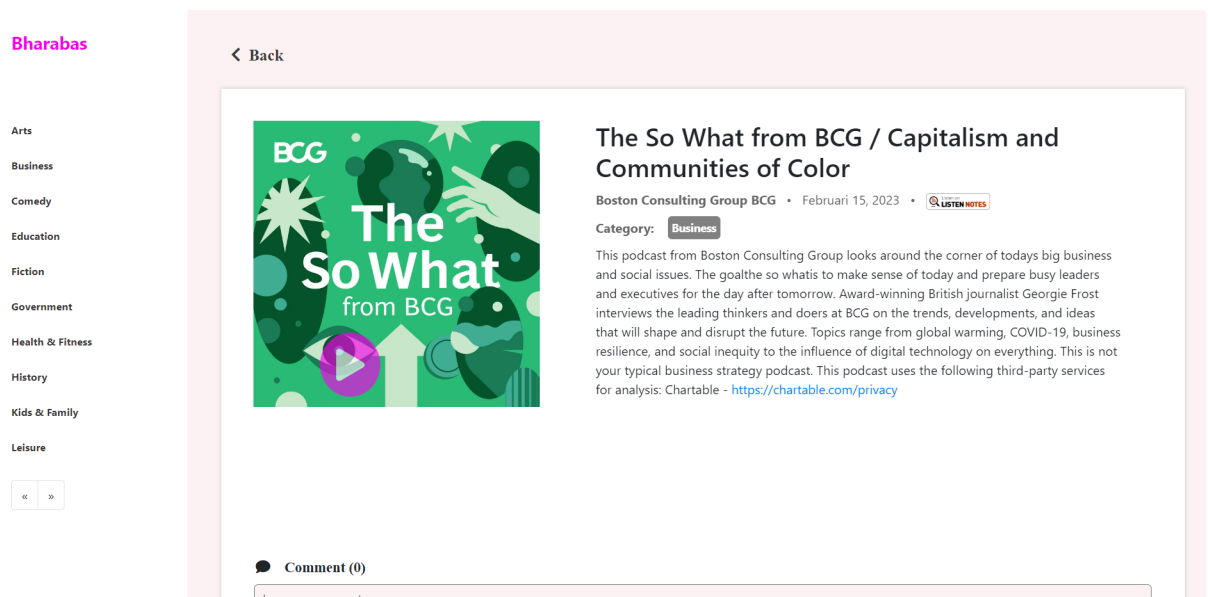


Gambar 8.2.4 Tampilan halaman pemutaran konten audio

Ketika pengguna melakukan klik tombol “*Play*” pada konten audio yang diinginkan, maka *bar player* akan muncul pada bagian bawah yang berisikan judul, *author*, serta tombol *play/pause*. Pengguna juga akan mendengarkan dari konten audio yang diputar.

8.3 Halaman Detail Audio

8.3.1 Detail Audio

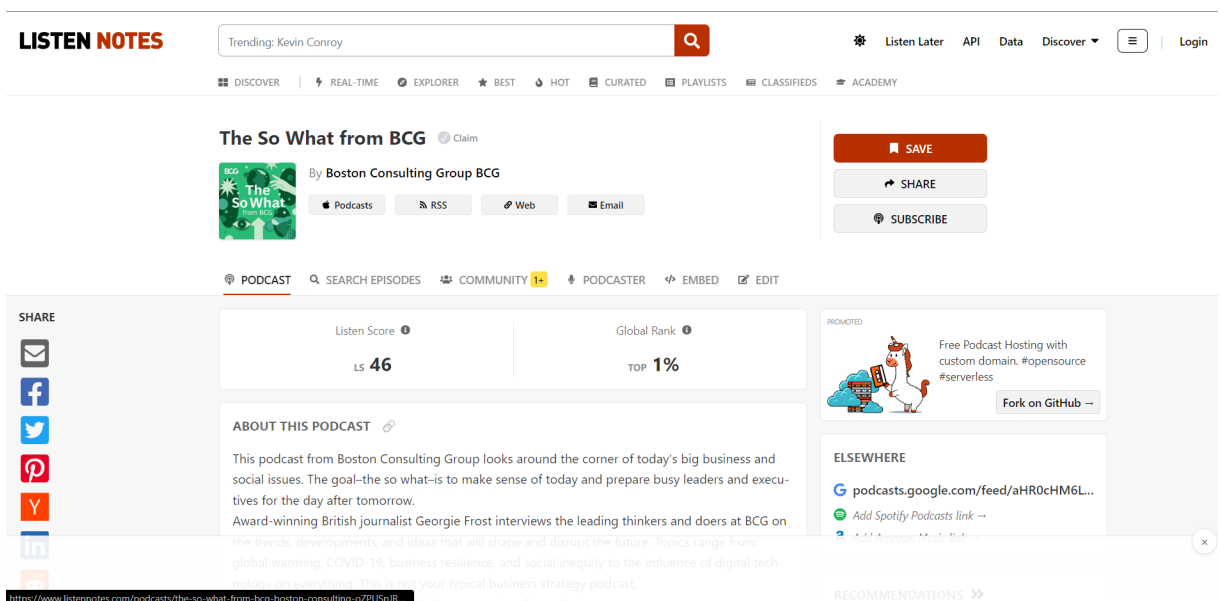


Gambar 8.3.1 Tampilan halaman detail audio

Ketika pengguna melakukan klik pada suatu judul konten audio, maka akan dialihkan ke halaman detail audio tersebut. Detail dari konten audio ini terdiri atas :

- Gambar konten audio
- Judul konten audio
- *Author* konten audio
- Tanggal konten audio dirilis
- Platform asal konten audio, jika pengguna klik tombol platform ini, akan dialihkan ke halaman **Platform sumber (8.3.2)**
- Kategori dan deskripsi konten audio

8.3.2 Platform Sumber



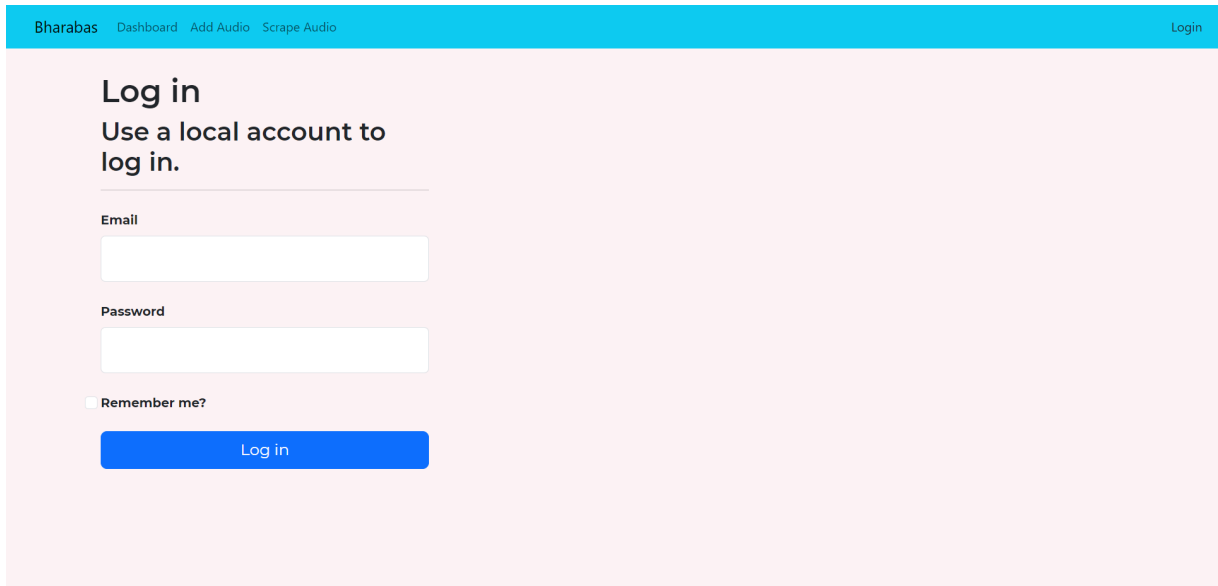
Gambar 8.3.2 Tampilan halaman platform sumber

Jika pengguna melakukan klik pada tombol platform, maka akan *redirect* ke halaman sumber aslinya.

8.4 Halaman Admin

Untuk mengakses ke halaman admin, hanya dapat dilakukan dengan melakukan penambahan `/Dashboard` pada *address website* Bharabas.

8.4.1 Login

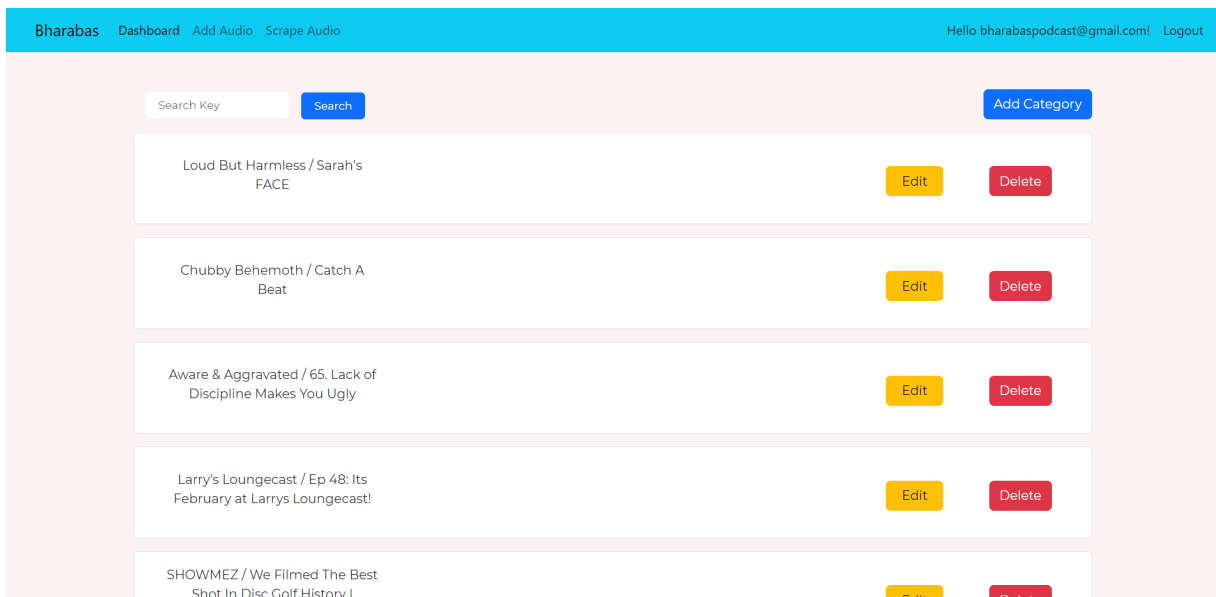


Gambar 8.4.1 Tampilan halaman *login*

Ketika pertama kali membuka halaman admin, akan diminta verifikasi terlebih dahulu dengan melakukan *login* menggunakan akun yang admin yang telah tersimpan. Langkah-langkah yang harus dilakukan adalah :

- Isi *email* menggunakan akun yang valid
- Isi *password* yang benar dari *email* yang dimasukkan
- Jika ingin menyimpan akun, dapat melakukan *checkbox* pada “Remember Me”
- Klik *Log In*
- Admin akan dialihkan ke halaman **Dashboard (8.4.2)**

8.4.2 Dashboard

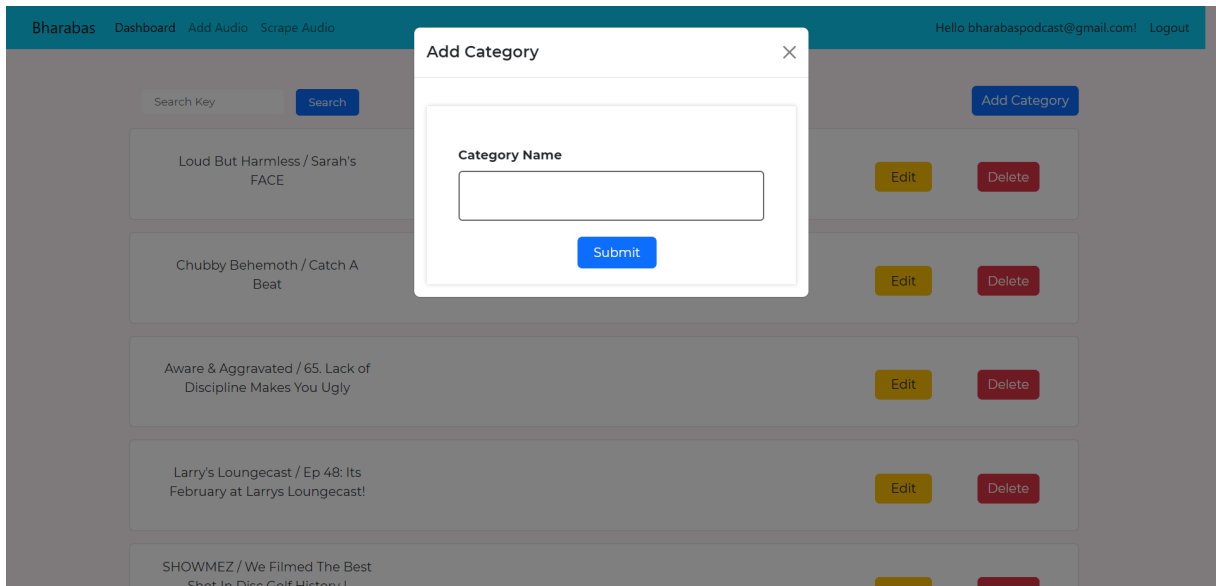


Gambar 8.4.2 Tampilan halaman *dashboard*

Pada halaman *dashboard website* Bharabas, admin akan melihat tampilan seperti gambar di atas. Pada bagian atas, terdapat *navigation bar* yang terdiri atas *Dashboard*, *Add Audio*, dan *Scrape Audio*. Pada halaman utama juga ditampilkan *search bar* untuk mencari lagu berdasarkan judul, tombol *Add Category*, dan semua judul konten audio yang dapat diedit dan dihapus..

- Jika admin ingin menambah kategori, maka dapat melakukan klik pada tombol “*Add Category*” dan akan dialihkan ke bagian **Menambah Kategori (8.4.3)**.
- Jika admin ingin menambah konten audio, maka dapat melakukan klik pada *navbar* “*Add Audio*” dan akan dialihkan ke bagian **Menambah Konten Audio (8.4.4)**.
- Jika admin ingin mengedit konten audio, maka dapat melakukan klik pada tombol “*Edit*” pada konten audio yang diinginkan dan akan dialihkan ke bagian **Meng-edit Konten Audio (8.4.5)**.
- Jika admin ingin menghapus konten audio, maka dapat melakukan klik pada tombol “*Delete*” pada konten audio yang diinginkan dan akan menghapus konten audio tersebut.
- Jika admin ingin melakukan *scraping* konten audio, maka dapat melakukan klik pada *navbar* “*Scrape Audio*” dan akan dialihkan ke bagian **Melakukan Scraping Konten Audio (8.4.5)**.

8.4.3 Menambah Kategori



Gambar 8.4.3 Tampilan *dialog box* menambahkan kategori

Untuk menambah kategori, langkah-langkah yang harus dilakukan adalah :

- Mengisi nama kategori yang diinginkan pada *bar* yang disediakan
- Klik tombol “*Submit*”
- Kategori baru telah tersimpan

8.4.4 Menambah Konten Audio

COVER IMAGE URL	TITLE	
<input type="text"/>	<input type="text"/>	
AUDIO SOURCE URL	CONTENT CREATOR	UPLOADED DATE
<input type="text"/>	<input type="text"/>	<input type="text" value="mm/dd/yyyy"/>
	PLATFORM	
	<input type="text" value="Platform"/>	<input type="text"/>
	CATEGORY	
	<input type="text" value="Nothing selected"/>	
	DESCRIPTION	
	<input type="text"/>	
	<input type="button" value="Submit"/>	

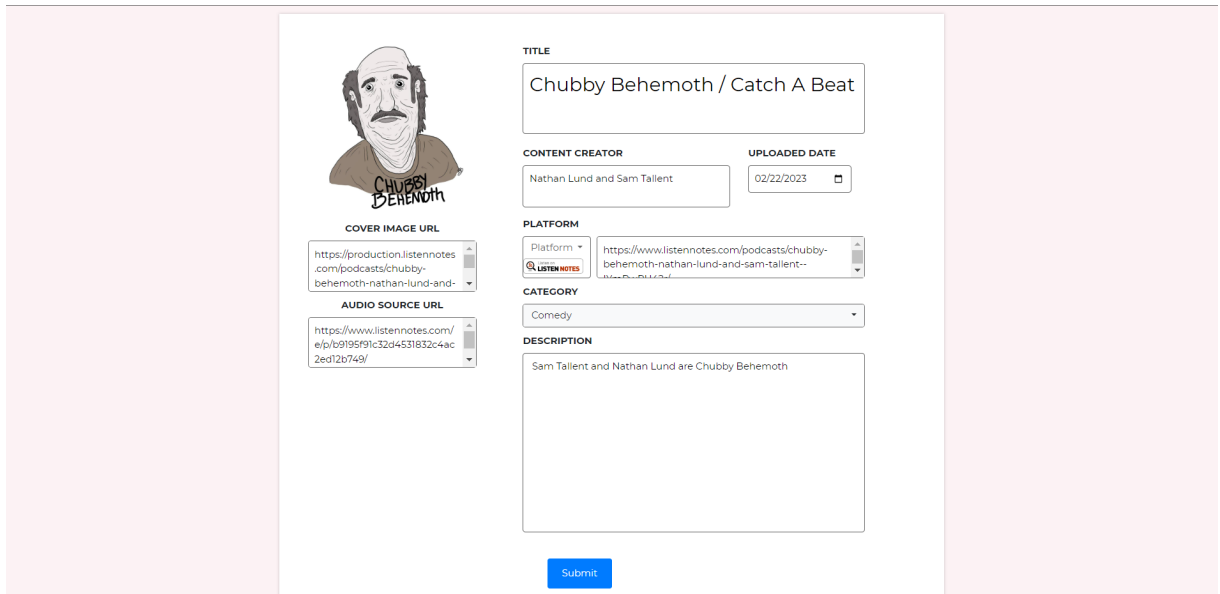
Gambar 8.4.4 Tampilan halaman menambah konten audio

Untuk menambah konten audio, langkah-langkah yang harus dilakukan adalah :

- Mengisi url *image* yang akan ditampilkan pada konten audio yang diinginkan
- Mengisi url *audio* yang akan diputar
- Mengisi judul konten audio
- Mengisi nama konten kreator konten audio
- Memilih tanggal melakukan *upload* konten audio

- Memilih platform sumber beserta *link url*-nya
- Memilih kategori yang sesuai
- Mengisi deskripsi untuk konten audio
- Klik tombol “*Submit*”

8.4.5 Meng-*edit* Konten Audio



The screenshot shows a web form for editing audio content. On the left, there is a cover image of a man with a mustache, labeled 'CHUBBY BEHEMOTH'. Below the image are two text input fields: 'COVER IMAGE URL' and 'AUDIO SOURCE URL'. The main form area on the right contains the following fields:

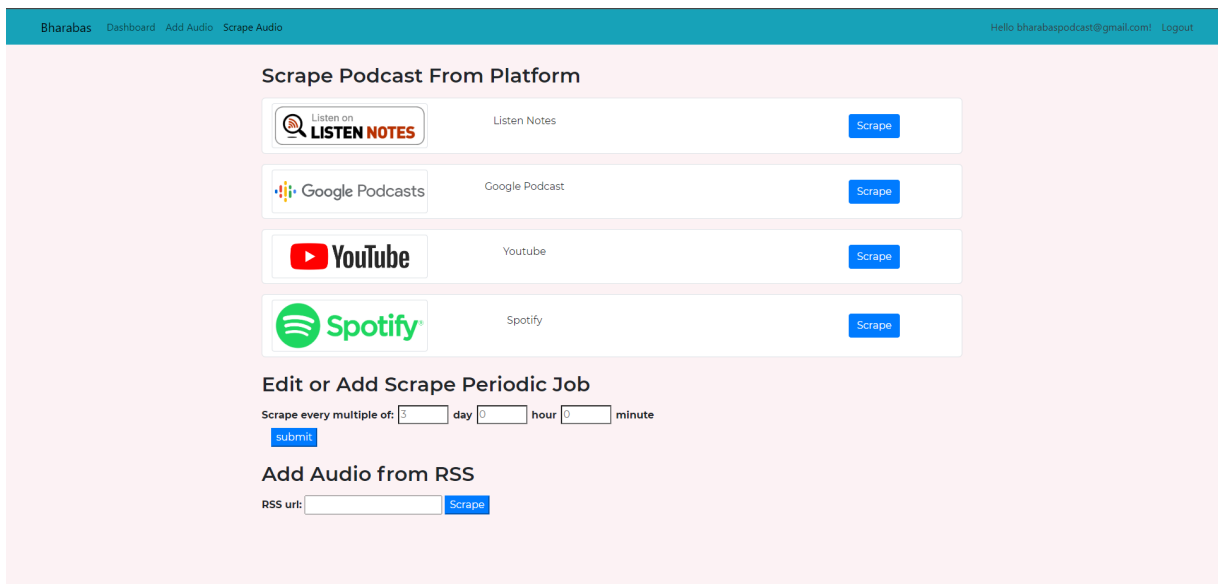
- TITLE:** Chubby Behemoth / Catch A Beat
- CONTENT CREATOR:** Nathan Lund and Sam Tallent
- UPLOADED DATE:** 02/22/2023
- PLATFORM:** Listen Notes
- CATEGORY:** Comedy
- DESCRIPTION:** Sam Tallent and Nathan Lund are Chubby Behemoth

A blue 'Submit' button is located at the bottom center of the form.

Gambar 8.4.5 Tampilan halaman meng-*edit* konten audio

Untuk mengedit konten audio yang dipilih, admin dapat mengubahnya dengan mengganti pada bagian yang diinginkan, lalu klik tombol “*Submit*”.

8.4.6 Melakukan *Scraping* Konten Audio



The screenshot shows a web interface for scraping audio content. The top navigation bar includes 'Bharabas', 'Dashboard', 'Add Audio', and 'Scrape Audio'. The main content area is titled 'Scrape Podcast From Platform' and contains four rows, each with a platform logo, the platform name, and a 'Scrape' button:

- Listen on LISTEN NOTES
- Google Podcasts
- Youtube
- Spotify

Below this section is a section titled 'Edit or Add Scrape Periodic Job' with a form to set the scraping frequency: 'Scrape every multiple of: 5 day 0 hour 0 minute'. A blue 'submit' button is located below the form.

At the bottom is a section titled 'Add Audio from RSS' with an 'RSS url:' input field and a blue 'Scrape' button.

Gambar 8.4.6 Tampilan halaman *scraping* konten audio

Untuk melakukan *scraping* konten audio, langkah-langkah yang harus dilakukan adalah :

- Jika admin ingin melakukan *scrape* secara manual sesuai *platform* yang diinginkan, dapat melakukan klik pada tombol “*Scrape*” pada *platform* Listen Notes/Google Podcast/Youtube/Spotify.
- Jika admin ingin mengatur periode waktu akan melakukan *scrape* secara otomatis, dapat mengisi waktu pada *bar* yang telah disediakan lalu klik tombol “*Submit*”
- Jika ingin menambah audio dari *scrape* RSS secara langsung, dapat mengisi RSS url pada *bar* yang telah disediakan, lalu klik tombol “*Scrape*”

8.4.7 Mengelola *Job Scraping* Konten Audio

Hangfire Dashboard	Jobs 1	Retries 0	Recurring Jobs 4	Servers 1	Back to site
--------------------	---------------	------------------	-------------------------	------------------	------------------------------

Recurring Jobs						
Trigger now Delete		Items per page: 10 20 50 100 500 1,000 5,000				
<input type="checkbox"/> Id	Cron	Time zone	Job	Next execution	Last execution	Created
<input type="checkbox"/> Periodic Scrape Listen Notes	0 0 */3 * *	SE Asia Standard Time	ScrapingListenNotes.scrapeData	In 2 days	20 hours ago	4 days ago
<input type="checkbox"/> Periodic Scrape Google Podcast	0 0 */3 * *	SE Asia Standard Time	ScrapingGooglePodcast.scrapeDataAsync	In 2 days	20 hours ago	4 days ago
<input type="checkbox"/> Periodic Scrape Youtube	0 0 */3 * *	SE Asia Standard Time	ScrapingYouTube.ScrapeData	In 2 days	19 hours ago	4 days ago
<input type="checkbox"/> Periodic Scrape Spotify	0 0 */3 * *	SE Asia Standard Time	ScrapingAnchor.scrapeDataAsync	In 2 days	15 hours ago	4 days ago

Total Items: 4

Gambar 8.4.7.1 Tampilan halaman *periodic job* untuk *scraping*

Setelah melakukan *scraping* atau mengatur periode waktu *scraping*, admin akan diarahkan ke halaman ini yang beralamat di `/dashboard/hangfire/recurring`. Di halaman ini terdapat informasi *schedule* yang digunakan (0 0 */3 * * artinya pada pukul 00.00 pada tanggal 1,4, ..., 31), kapan eksekusi berikutnya, dan kapan eksekusi terakhir. Selain itu, admin dapat mencentang *job* yang ada dan menekan *trigger now* untuk menjalankannya sekarang juga.

Hangfire Dashboard	Jobs 1	Retries 0	Recurring Jobs 4	Servers 1	Back to site
--------------------	---------------	------------------	-------------------------	------------------	------------------------------

Processing Jobs			
Requeue jobs Delete selected		Items per page: 10 20 50 100 500 1,000 5,000	
<input type="checkbox"/> Id	Server	Job	Started
<input type="checkbox"/> #59	LW1SDLWK0000VP:68	ScrapingRSS.scrapeRSSAsync	a few seconds ago

Total Items: 1

Gambar 8.4.7.2 Tampilan halaman daftar *job* untuk *scraping*

Pada halaman ini admin dapat melihat daftar *job* dan statusnya seperti sedang diproses, sudah sukses, dan dihapus. Admin juga dapat membatalkan *job* yang sedang diproses. Setiap *job* yang sudah selesai akan dihapus otomatis setelah 24 jam.