



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk mengubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Kedudukan dalam pelaksanaan kerja magang ini berada di bawah pengawasan dari Bapak Julio Christian Young selaku Kepala Proyek Penelitian. Kemudian ditetapkan sebagai *programmer* untuk membuat website platform pengumpulan *speech* dataset dengan menggunakan php. Pelaksanaan kerja magang ini juga dilakukan bersama rekan kerja yang lainnya. Pekerjaan yang dilakukan selama kerja magang ini adalah membuat website *platform* pengumpulan *speech* dataset untuk bahasa jepang dengan menggunakan php.

3.2 Tugas Yang Dilakukan

Tugas yang dikerjakan selama kegiatan kerja magang adalah membuat website *platform* pengumpulan *speech* dataset dengan menggunakan bahasa PHP. Front end website menggunakan *package* yang sudah disediakan oleh *framework* laravel dan tambahan beberapa *package* untuk membuat *user interface* yang mudah dan nyaman dilihat oleh pengguna. Back end menggunakan *framework* laravel untuk memudahkan dalam pembuatan website. Berikut tugas utama yang dikerjakan selama magang kerja:

1. Mendiskusikan konsep website dengan pembimbing lapangan kerja magang
2. Mempelajari bahasa pemrograman php menggunakan laravel.
3. Melakukan perancangan alur kerja website, database, beserta relasinya dalam menyimpan, mengubah, dan manajemen data pada website.
4. Melakukan perancangan tampilan website untuk memudahkan pengguna melakukan donasi atau validasi. Perancangan juga dibuat untuk admin, sehingga memudahkan admin dalam melakukan pemantauan dan manajemen website.
5. Membuat website menggunakan bahasa pemrograman php dengan *framework* laravel dan berkolaborasi dengan rekan kerja untuk membuat website yang baik dan benar.

3.3 Uraian Pelaksanaan Kerja Magang

Pelaksanaan kerja magang ini terdiri dari kegiatan yang dilakukan selama kerja magang berlangsung. Penjabaran tentang hal di atas akan dijelaskan pada bagian ini.

3.3.1 Proses Pelaksanaan

Sebelum melakukan proses pelaksanaan, pembimbing magang menjelaskan konsep serta penjabaran detail mengenai website platform pengumpulan *speech* dataset. Kemudian dijabarkan mengenai tabel dan database yang akan digunakan untuk menyimpan data serta *timeline* kerja magang. Kerja magang ini dilaksanakan selama 360 jam. Tabel realisasi kerja magang dapat dilihat pada Tabel 3.1.

Tabel 3.1 Realisasi Kerja Magang

Minggu	Kegiatan
1	<ul style="list-style-type: none">- Membahas detail, konsep, deskripsi pekerjaan magang, serta materi pembelajaran laravel- Mempelajari materi php dengan menggunakan laravel
2	<ul style="list-style-type: none">- Mempelajari materi laravel
3	<ul style="list-style-type: none">- Diskusi sitemap dan database projek- Membuat Register dan login website
4	<ul style="list-style-type: none">- Membuat tabel User- Membuat tampilan admin, contribute, update profile- Membuat Profile Management
5	<ul style="list-style-type: none">- Revisi tampilan contribute- Pembuatan tabel terms- Tampilan form add user untuk user management
6	<ul style="list-style-type: none">- Diskusi perkembangan website- Menambahkan datatable untuk Syllable dan User Management- Revisi Profile Management- Pembuatan Syllable Management
7	<ul style="list-style-type: none">- Menambahkan fitur upload dan update data suara pada Syllable Management- Menambahkan fitur putar suara pada tampilan contribute- Pembuatan fitur updated_by, deleted_by, created_by dengan observer untuk semua tabel- Mengambil random data dan membuat tampilan validasi
8	<ul style="list-style-type: none">- Membuat relation antar tabel dan membuat fitur deleted_by pada tabel donation- Revisi observer, hanya untuk tabel terms- Pembuatan middleware untuk pengecekan pengguna login- Revisi tampilan utama admin dan store donasi ke database
9	<ul style="list-style-type: none">- Diskusi perkembangan projek dan tambahan beberapa fitur- Revisi middleware untuk pengecekan admin- Pembuatan tabel contents dan Content Management

10	<ul style="list-style-type: none"> - Membuat Content Management dengan menggunakan helper - Melanjutkan pekerjaan untuk Content Management - Pembuatan isi konten dan penambahan package icon
11	<ul style="list-style-type: none"> - Penambahan attribute bahasa pada tabel terms - Menambahkan isi konten ke database serta menambahkan fitur pengecekan pengguna - Membuat toast dengan menggunakan package toastr - Diskusi perkembangan projek dan tambahan fitur lain
12	<ul style="list-style-type: none"> - Revisi Content Management, Syllable Management - Pembuatan fitur Download Voice Management - Revisi isi konten website dan pembuatan konsep user journey - Testing, debugging, dan merapikan kode

3.3.2 Alur Kerja

Pada pertemuan pertama kegiatan kerja magang, pembimbing lapangan memberikan penjelasan tentang konsep dan deskripsi pekerjaan magang, serta memberikan penjelasan singkat tentang bahasa pemrograman yang akan digunakan dalam pembuatan website. Selanjutnya pembimbing magang memberikan referensi materi pembelajaran laravel untuk dipelajari selama minggu pertama kerja magang.

Sepanjang minggu kedua magang, lebih difokuskan untuk mempelajari materi php dengan menggunakan laravel. Pada minggu ketiga, pembimbing magang memberikan sitemap dan schema database yang akan digunakan dalam pembuatan website, kemudian dilakukan pembuatan login dan register untuk website. Pembuatan login menggunakan username dan password yang telah ada di database website. Sedangkan proses register berisi data diri pengguna berupa nama, password, secret question, dan secret answer. Secret question dan secret answer digunakan untuk proses validasi forgot password pengguna.

Pada minggu keempat, dilakukan pembuatan tabel user serta beberapa tampilan website baik dari sisi pengguna maupun admin. Kemudian dilanjutkan dengan pembuatan fitur update profile. Dimana pengguna dapat mengubah username. Pada minggu kelima kerja magang, terdapat beberapa revisi pada tampilan website dan dilanjutkan dengan pembuatan tabel terms yang berguna untuk menampung data bahasa jepang yang nantinya akan digunakan pengguna sebagai acuan kata untuk mendonasikan suaranya. Proses kerja magang dilanjutkan dengan membuat syllable management. Syllable management ini ada pada admin, dimana admin dapat membuat, menghapus, dan mengubah data pada tabel terms.

Pada minggu keenam kerja magang, pembimbing magang mendiskusikan perkembangan website. Pada minggu ini juga dibuat data tabel untuk memudahkan admin melihat data terms dan user. Pembuatan add, edit, dan soft delete pada website, data yang ditambahkan dan diubah disimpan di database sedangkan data yang dihapus tetap ada di database dan tidak ditampilkan pada website. Pada minggu ini juga dilakukan revisi pada profile management, dimana pengguna dapat mengubah gambar profil, username, dan kata sandi pengguna.

Pekerjaan kerja magang pada minggu ketujuh dilanjutkan dengan mengubah form untuk mengunggah, memutar, dan mengubah data file suara dengan format mp3 untuk syllable management. Data suara tersebut akan ditampilkan secara acak pada tampilan contribute, sehingga pengguna dapat mendonasikan suaranya pada data yang berbeda – beda. Pembuatan fitur updated, deleted, dan created_by pada semua tabel menggunakan observer sehingga data created, updated, dan deleted_by akan secara otomatis terisi tanpa perlu di program oleh programmer. Kemudian dilanjutkan dengan membuat relation antar tabel serta pengaplikasian fitur deleted_by pada tabel donation. Pembuatan middleware untuk pengecekan login pengguna sehingga pengguna yang belum login tidak dapat mengakses halaman admin. Revisi pada tampilan utama admin. Pada minggu ini observer diubah, yang sebelumnya untuk semua tabel menjadi hanya untuk tabel terms.

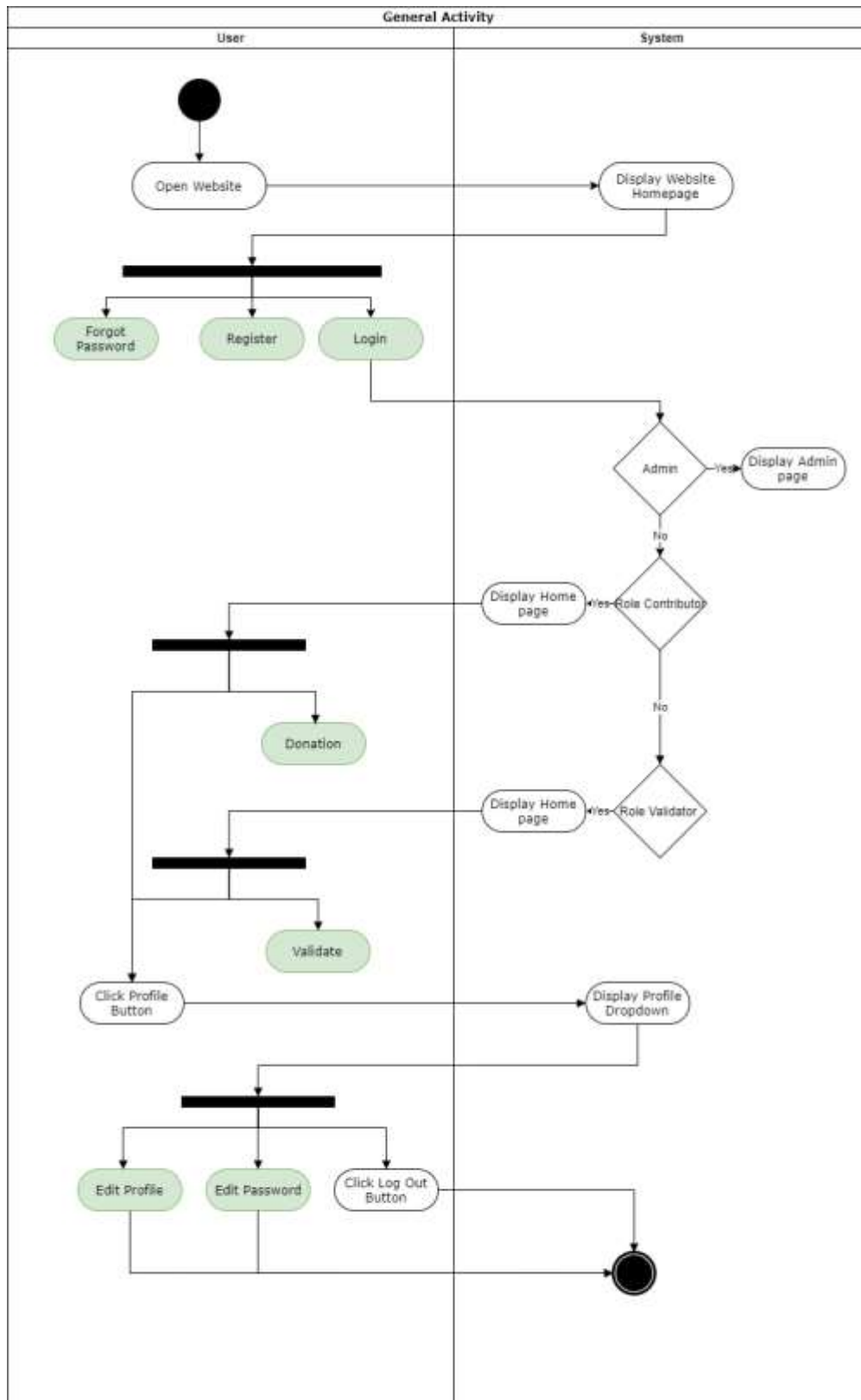
Pada minggu selanjutnya pembimbing magang melakukan diskusi perkembangan projek dan tambahan beberapa fitur pada website. Pada minggu ini dilakukan penambahan middleware untuk pengecekan role pengguna, pengguna yang bukan termasuk admin tidak dapat mengakses halaman admin. Pembuatan tabel contents dan content management berupa add, edit, dan delete konten. Content management berisi data konten yang terdapat pada website, sehingga memudahkan admin untuk mengubah isi konten pada website. Pembuatan content management dibantu dengan helper, serta pembuatan isi konten website.

Pekerjaan magang dilanjutkan dengan menambahkan attribute bahasa pada tabel terms, attribute ini berisi kategori bahasa untuk data terms. Terdapat kategori bahasa jepang, korea, dan bahasa indonesia. Isi konten yang telah dibuat pada minggu sebelumnya dimasukkan ke database dengan menggunakan seeder. Untuk

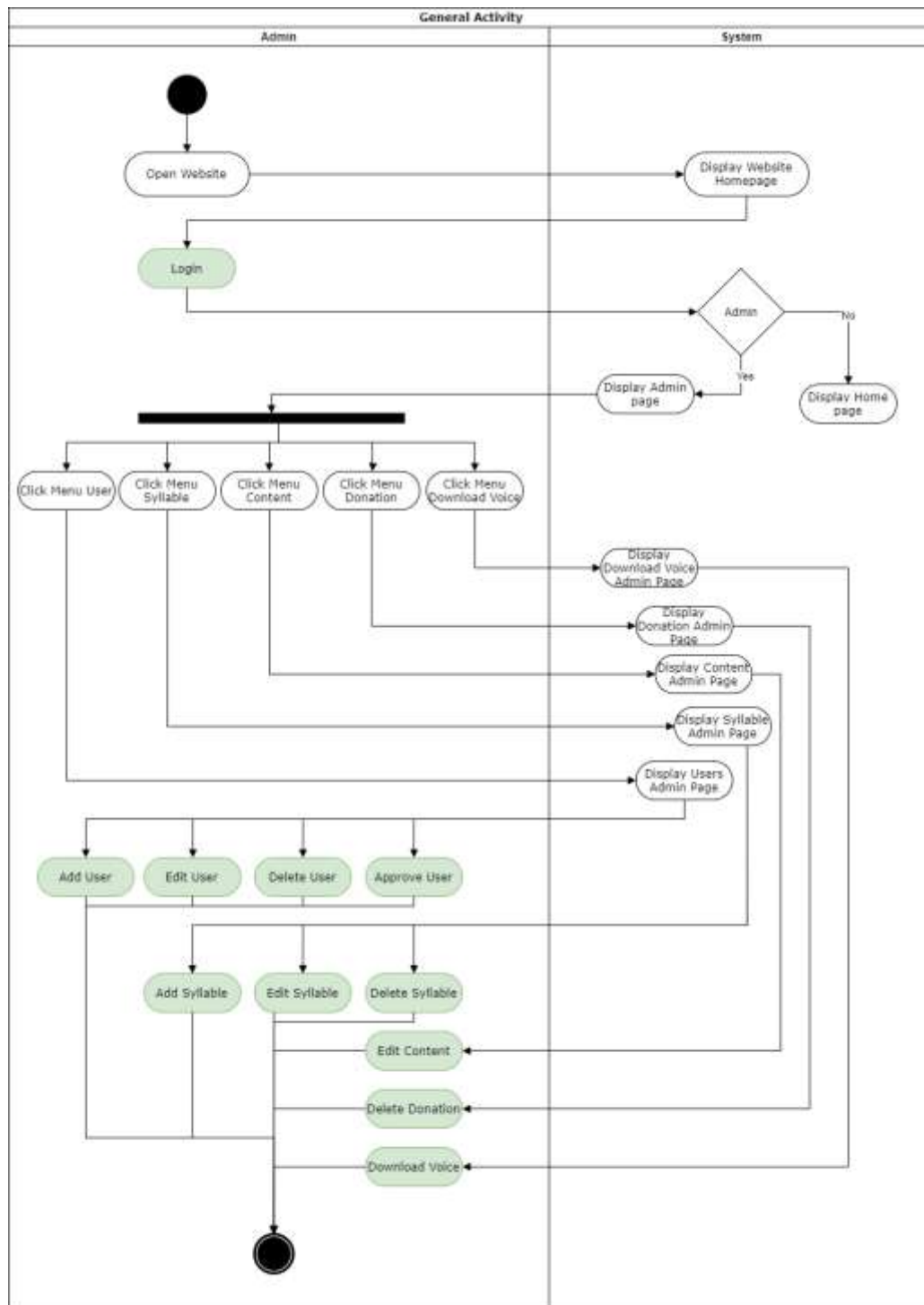
feedback agar tampilan menjadi menarik, website menggunakan toast yang diambil dari toastr. Pada minggu ini juga dilakukan diskusi perkembangan projek dan tambahan beberapa fitur.

Pada minggu selanjutnya, pekerjaan magang dilanjutkan dengan melakukan revisi pada Syllable Management dan Content Management. Dimana perubahan pada Content Management yaitu menghapus fitur add dan delete content. Sedangkan pada Syllable Management dilakukan perubahan pada tampilan serta pengambilan data dari database, sehingga kode pemrograman lebih singkat. Pada minggu ini juga dilakukan revisi isi konten serta pembuatan konsep user journey. Dilakukan testing dan debugging pada website serta merapikan kode.

Secara umum diagram alur kerja website dapat dilihat pada Gambar 3.1. Alur kerja website pada sisi pengguna dan 3.2. Alur kerja website pada sisi admin.



Gambar 3.1 Alur kerja website pada sisi pengguna.



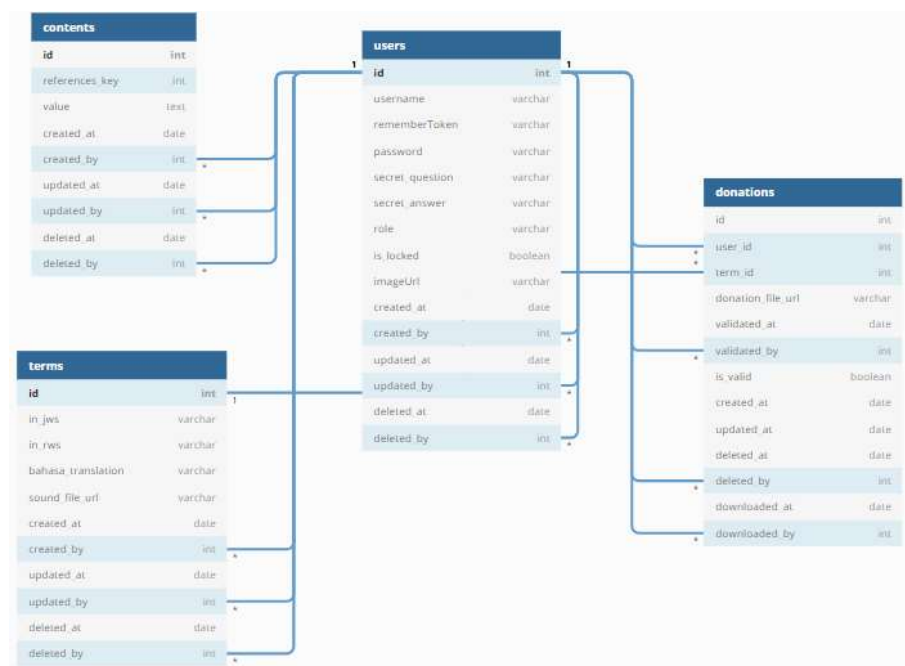
Gambar 3.2 Alur kerja website pada sisi admin

Terdapat *sub activity* pada Gambar 3.1 dan Gambar 3.2. Penjelasan detail mengenai *sub activity* ini akan dijelaskan pada bagian sub-activity diagram.

3.3.3 Struktur Database dan Relasi Tabel

Website yang dibuat merupakan website untuk mengumpulkan dataset suara dalam bahasa jepang, dimana pengguna dapat memberikan donasi berupa rekaman suara dalam bahasa jepang. Pengguna dapat memberikan donasi dalam hanya dengan membaca kata bahasa jepang yang tertera di website, kata jepang tersebut akan ditampilkan secara acak sehingga pengguna dapat menyumbangkan suara mereka dengan intonasi dan pelafalan yang berbeda. Suara tersebut akan disimpan ke dalam database dan akan divalidasi oleh pengguna lain yang berperan sebagai validator. Data suara dapat diubah, dihapus, maupun ditambahkan. Berikut struktur database dan relasi antar tabel yang digunakan pada pembuatan website ini dapat dilihat pada Gambar 3.3.

1. Relasi Antar Tabel



Gambar 3.3 Relasi Tabel

Pada Gambar 3.3, terdapat empat tabel. Pada tabel donations terdapat *many to one relation* terhadap tabel users, fungsi hubungan ini adalah mencatat id pengguna yang sudah memvalidasi data, mencatat id terms dan id pengguna yang menyumbangkan suara, dan mencatat id pengguna (admin) yang menghapus serta mengunduh data donasi. Kemudian tabel users juga memiliki *many to one relation* terhadap tabel

terms dan contents. Fungsi hubungan ini adalah mencatat id pengguna (admin) yang melakukan create, update, dan delete terhadap data syllable serta konten website. Tabel donations memiliki *many to one relation* terhadap tabel terms, dimana hubungan ini berfungsi untuk menyimpan id terms. Id tersebut digunakan untuk mengambil kata jepang yang akan ditampilkan bersama donasi suara untuk di validasi oleh pengguna lain. Tabel users juga memiliki *many to one relation* terhadap tabel user sendiri (recursive relation), hubungan ini berguna untuk pengguna yang memiliki role admin untuk memantau pengguna pada website tersebut.

2. Struktur Tabel

a. Tabel Users

Fungsi dari tabel ini adalah untuk menyimpan semua data pengguna beserta rolenya (kontributor atau validator) dan tanggal kapan data pengguna di update, dibuat, atau dihapus.

Tabel 3.2 Tabel Users

Nama Kolom	Tipe Data	Keterangan
ID	Integer	Id pengguna
Username	Varchar(255)	Nama pengguna dengan constraint unique
rememberToken	Varchar(100)	Untuk menyimpan token pengguna
Password	varchar(60)	Kata sandi pengguna, minimal 8 karakter
Secret_question	Varchar(1024)	Pertanyaan ketika pengguna lupa kata sandi
Secret_answer	Varchar(255)	Jawaban ketika pengguna lupa kata sandi
Role	Varchar(15)	Role pengguna (validator atau kontributor)
Is_locked	Boolean	Status pengguna
imageUrl	Varchar(255)	File gambar profil pengguna

Created_at	Date	Tanggal pembuatan akun pengguna
Created_by	Integer	Id pembuat akun pengguna
Updated_at	Date	Tanggal akun pengguna diubah
Updated_by	Integer	Id pengubah akun pengguna
Deleted_at	Date	Tanggal akun pengguna dihapus
Deleted_by	Integer	Id yang menghapus akun pengguna

b. Tabel Terms

Fungsi dari tabel ini adalah untuk menyimpan data berisi kata jepang, kata untuk membaca kata jepang tersebut, dan arti dari kata jepang. Kata tersebut digunakan untuk membantu pengguna membaca dan mereka suara mereka dalam bahasa jepang. Pada tabel ini juga berisi file audio yang bisa digunakan pengguna sebagai contoh sebelum mendonasikan suara mereka.

Tabel 3.3 Tabel Terms

Nama Kolom	Tipe Data	Keterangan
ID	Integer	Id kata jepang
In_jws	Varchar(255)	Kata jepang
In_rws	Varchar(255)	Cara membaca kata jepang tersebut
Bahasa_translation	Varchar(255)	Arti dari bahasa jepang
bahasa	Varchar(255)	Kategori bahasa
Sound_file_url	Varchar(2083)	Link file audio
Created_at	Date	Tanggal pembuatan akun pengguna
Created_by	Integer	Id pembuat akun pengguna
Updated_at	Date	Tanggal akun pengguna diubah
Updated_by	Integer	Id pengubah akun pengguna

Deleted_at	Date	Tanggal akun pengguna dihapus
Deleted_by	Integer	Id yang menghapus akun pengguna

c. Tabel Donations

Fungsi tabel ini adalah untuk menyimpan semua data suara yang sudah didonasikan oleh pengguna. Pada tabel ini juga disimpan tanggal validasi dan oleh siapa suara tersebut di validasi.

Tabel 3.4 Tabel Donations

Nama Kolom	Tipe Data	Keterangan
ID	Integer	Id donasi
User_id	Integer	Id pengguna yang melakukan donasi
Terms_id	Integer	Id data bahasa jepang
Donation_file_url	Varchar(2083)	Link file audio donasi
Validated_at	Date	Tanggal data ini di validasi
Validated_by	Integer	Id pengguna yang memvalidasi data
Is_valid	Boolean	Status data donasi
Downloaded_at	Date	Tanggal data di unduh
Downloaded_by	Integer	Id pengguna yang mengunduh data
Created_at	Date	Tanggal pembuatan donasi
Updated_at	Date	Tanggal data donasi diubah
Deleted_at	Date	Tanggal akun pengguna dihapus
Deleted_by	Integer	Id yang menghapus akun pengguna

d. Tabel Contents

Fungsi dari tabel ini adalah untuk menyimpan semua konten yang ada di dalam website, sehingga jika ada perubahan dalam website tersebut, admin hanya perlu mengubah data pada tabel ini tanpa perlu memprogram ulang website tersebut.

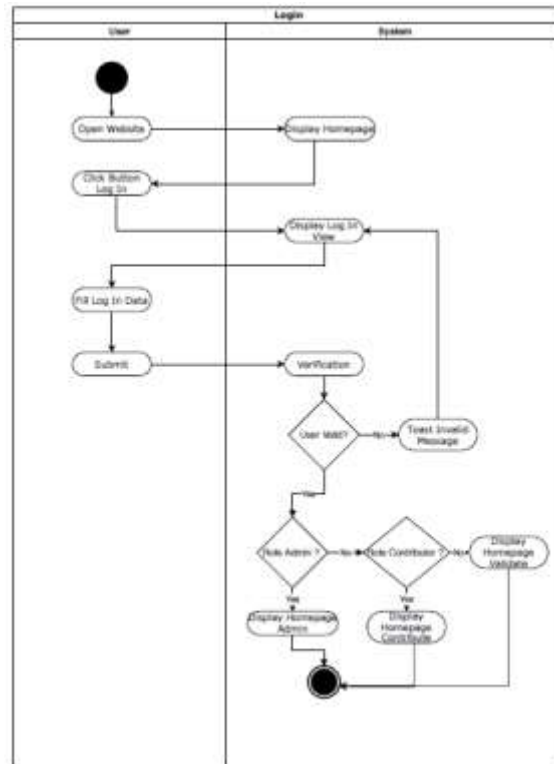
Tabel 3.5 Tabel Contents

Nama Kolom	Tipe Data	Keterangan
ID	Integer	Id konten
Reference_key	Varchar(255)	Flag konten website
Value	Text	Isi dari konten tersebut
Created_at	Date	Tanggal pembuatan konten website
Created_by	Integer	Id pembuat konten website
Updated_at	Date	Tanggal konten diubah
Updated_by	Integer	Id pengubah konten
Deleted_at	Date	Tanggal konten dihapus
Deleted_by	Integer	Id yang menghapus konten

3.3.4 Sub-Activity Diagram

Di bawah ini adalah penjelasan detail *sub activity diagram* website platform pengumpulan *speech* dataset bahasa Jepang.

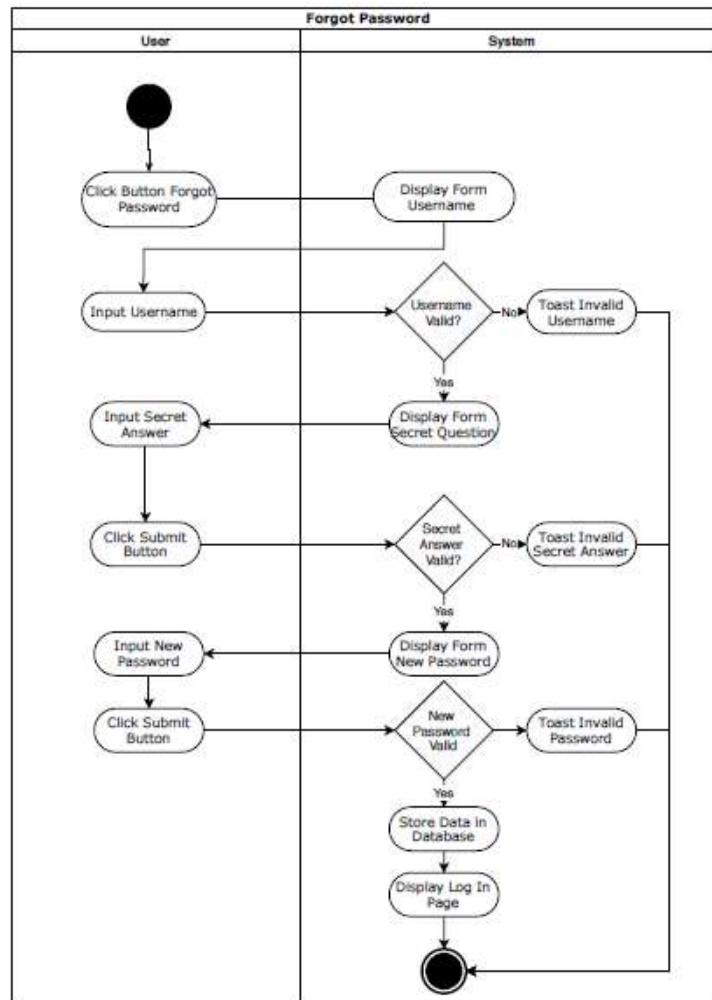
A. Login



Gambar 3.4 Activity Login

Dapat dilihat pada Gambar 3.4, pengguna yang sudah memiliki akun melakukan proses login dengan menekan tombol *login* pada halaman website, kemudian sistem akan menampilkan tampilan *login*. Setelah pengguna memasukkan username dan password, sistem akan memeriksa apakah pengguna tersebut valid, jika iya sistem akan melakukan pengecekan role pengguna tersebut. Pengguna yang memiliki role admin akan diarahkan ke tampilan admin, pengguna yang memiliki role contributor akan diarahkan ke tampilan contribute agar pengguna dapat mendonasikan suara, dan pengguna akan diarahkan ke tampilan validate jika pengguna memiliki role validator.

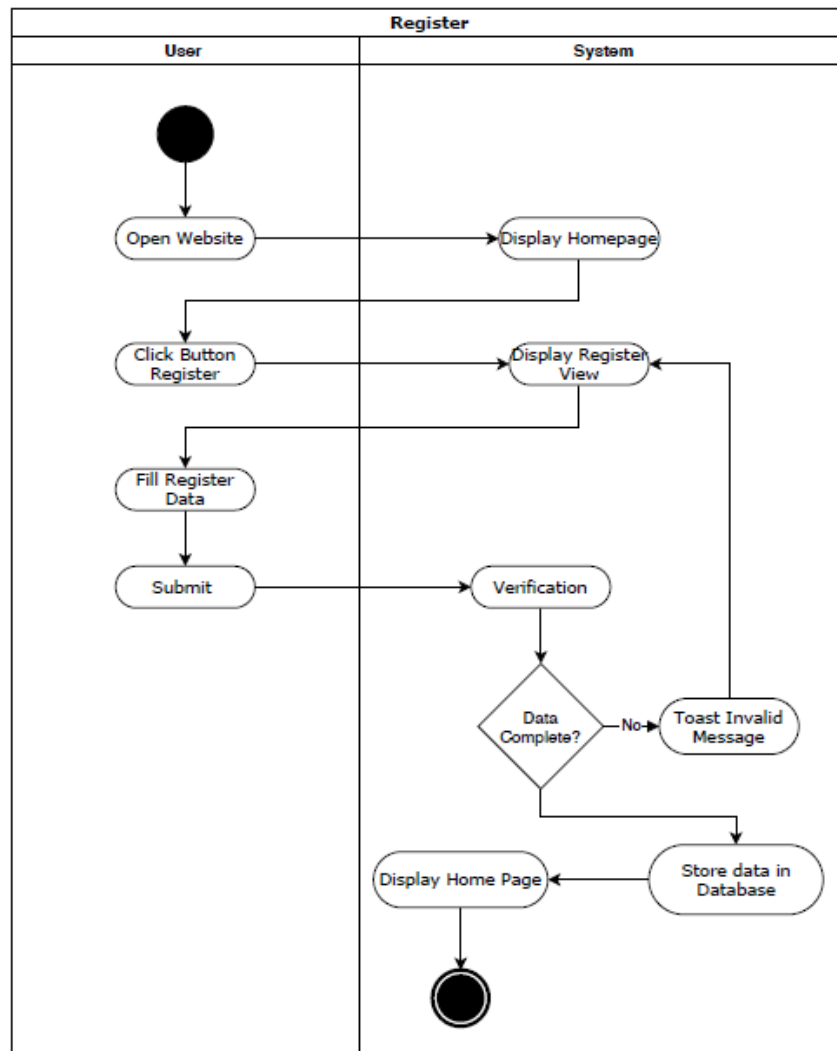
B. Forgot Password



Gambar 3.5 Activity Forgot Password

Dapat dilihat pada Gambar 3.5, apabila pengguna tidak mengingat kata sandi akun mereka maka pengguna dapat menekan tombol forgot password dan akan diarahkan ke tampilan forgot password berupa form dan memasukkan username. Pengguna akan diminta untuk menjawab pertanyaan sesuai dengan pertanyaan yang dipilih oleh pengguna saat registrasi, apabila jawaban pengguna cocok dengan data yang ada di database. Pengguna akan diarahkan ke tampilan untuk membuat kata sandi baru. Apabila kata sandi sudah dimasukan dan kata sandi valid, pengguna akan diarahkan ke tampilan login.

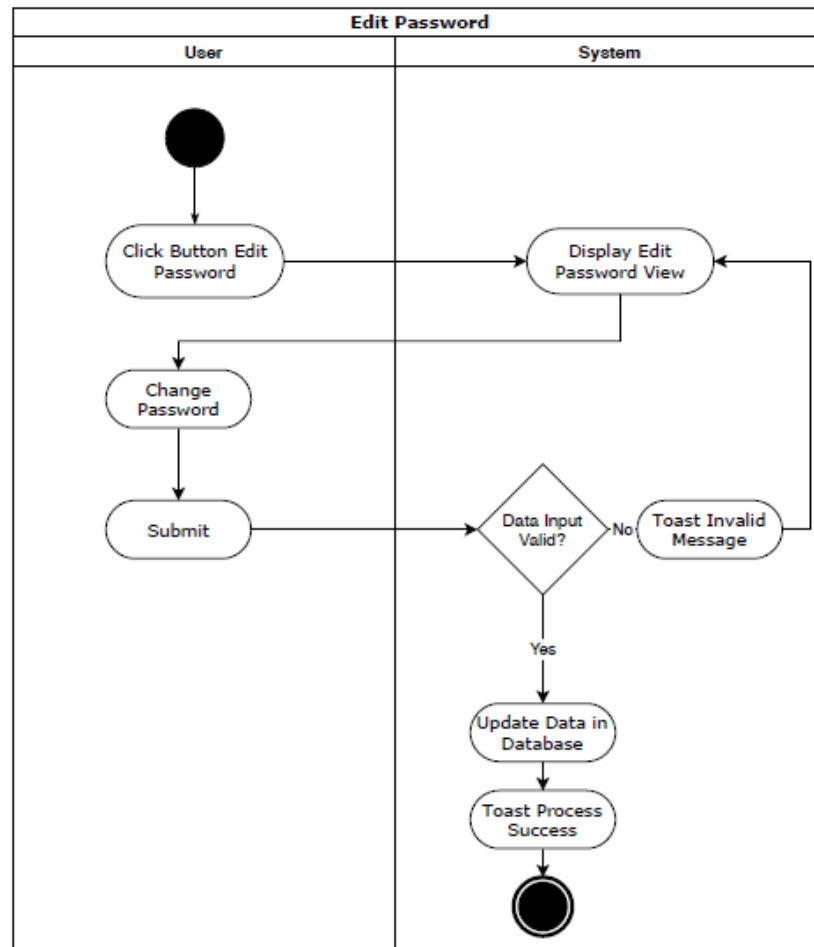
C. Register



Gambar 3.6 Activity Register

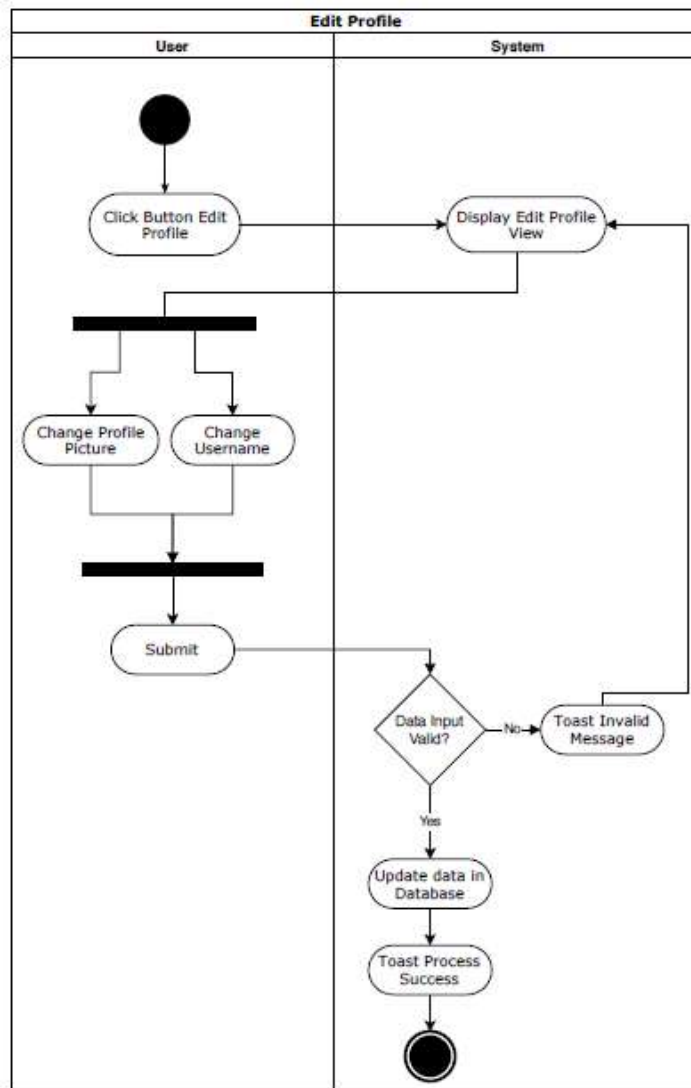
Gambar 3.6 menunjukkan diagram aktivitas dimana pengguna yang belum memiliki akun melakukan proses register untuk dapat melakukan donasi ataupun validasi suara dengan menekan tombol *register* pada halaman website. Sistem akan menampilkan halaman register dan pengguna mengisi data diri yang diperlukan. Ketika pengguna selesai mengisi data, sistem akan melakukan pengecekan terhadap data yang diisi pengguna. Sistem akan mengeluarkan *alert* apabila data pengguna tidak valid, jika data pengguna valid sistem akan menyimpan data pengguna ke dalam database.

D. Edit Profile dan Password



Gambar 3.7 Activity Edit Password

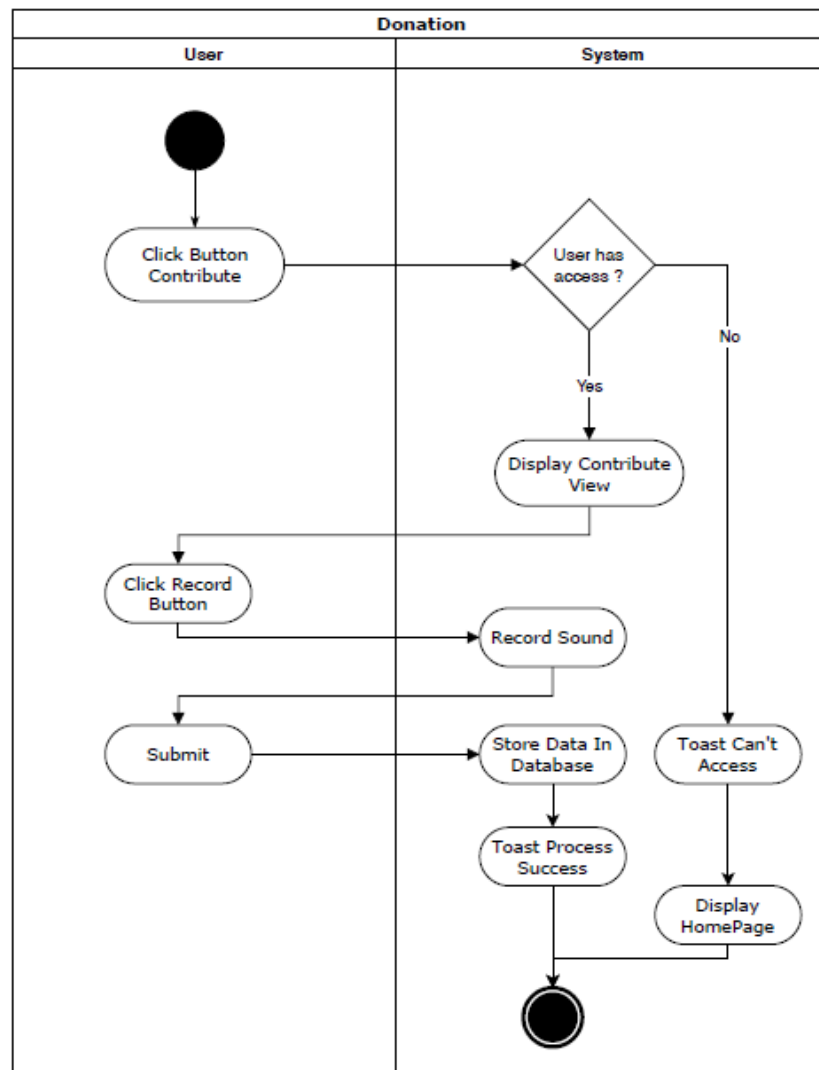
Pengguna dapat mengganti kata sandi dengan cara menekan tombol edit password pada dropdown profile di halaman utama website. Data yang telah diisi oleh pengguna akan dilakukan pengecekan dan disimpan ke dalam database.



Gambar 3.8 Activity Edit Profile

Pengguna dapat mengganti gambar profil atau nama pengguna dengan cara menekan tombol edit profile pada dropdown list profile di halaman website. Gambar profil dan nama pengguna yang sudah diisi akan dilakukan pengecekan dan disimpan dalam database.

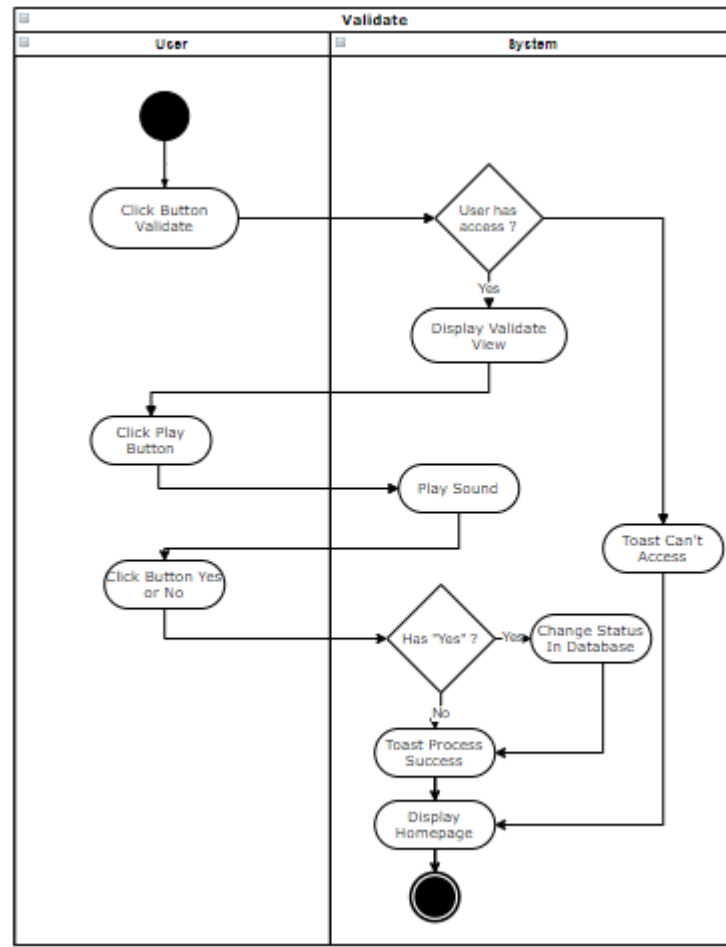
E. Donation



Gambar 3.9 Activity Donation

Pengguna yang memiliki role Contributor dapat melakukan donasi suaranya. Pengguna dapat menekan tombol Contribute pada halaman website dan sistem akan melakukan pengecekan hak akses pengguna tersebut, jika pengguna sudah diberikan hak akses oleh admin maka website akan menampilkan tampilan donasi kepada pengguna. Namun jika belum, website akan menampilkan toast. Pengguna yang ingin melakukan donasi dapat menekan tombol record, sistem akan merekam suara pengguna. Ketika pengguna menekan tombol submit maka donasi suara tersebut akan disimpan di database.

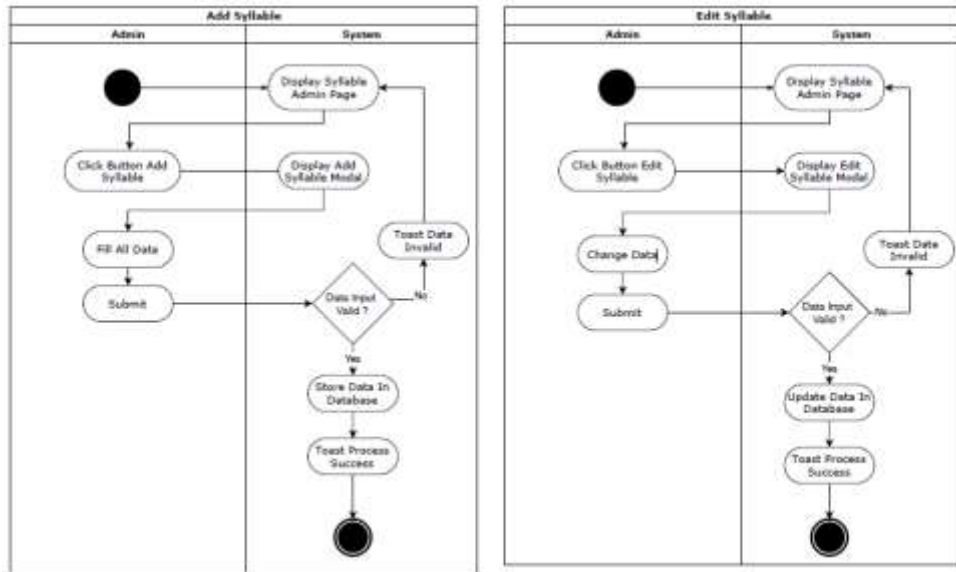
F. Validation



Gambar 3.10 Activity Validate

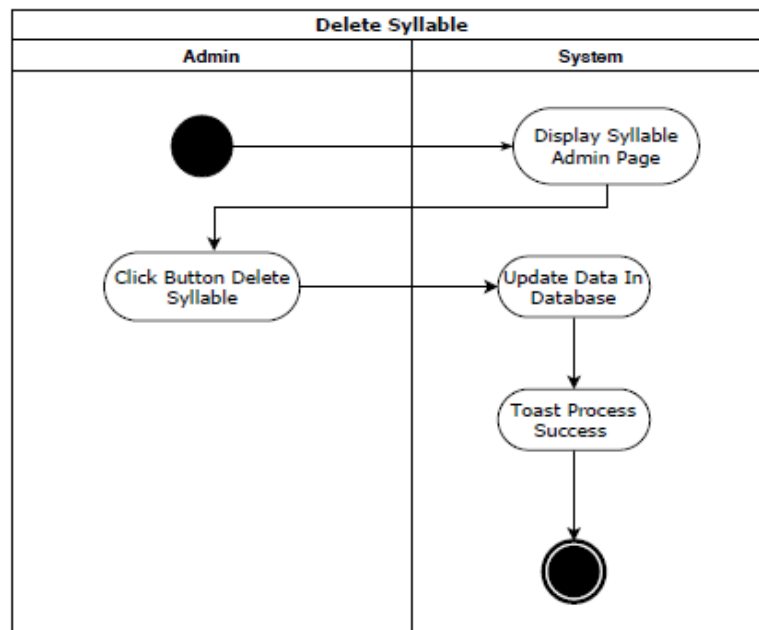
Pengguna yang memiliki role Validator dapat melakukan validasi data suara. Pengguna dapat menekan tombol Validate pada halaman website, sistem akan melakukan pengecekan hak akses. Jika pengguna sudah diberikan hak akses sistem akan menampilkan tampilan validasi, namun jika pengguna tidak memiliki hak akses sistem akan menampilkan toast dan menampilkan tampilan utama. Pengguna dapat melakukan validasi dengan cara menekan tombol yes/no pada website. Apabila tombol yes yang ditekan, maka data donasi di database akan diubah. Namun apabila tombol no yang ditekan data donasi tidak berubah. Setiap kali pengguna mendonasikan atau memvalidasi suara tampilan akan memberikan *feedback* berupa toast.

G. Create, Update, dan Delete Syllable



Gambar 3.11 Activity Add dan Edit Syllable

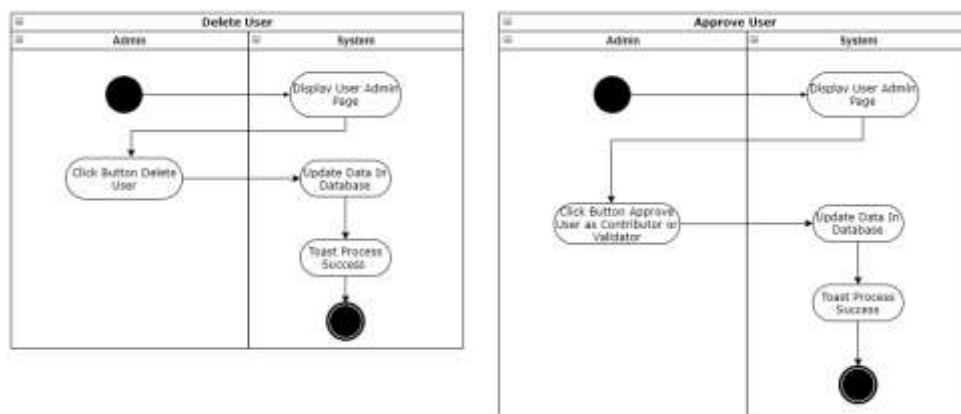
Pada Gambar 3.11, admin dapat membuat dan mengubah data kata donasi dengan cara menekan tombol edit atau add pada halaman Syllable. Ketika admin menekan tombol add syllable, add syllable modal akan muncul. Admin dapat mengisi data pada modal tersebut dan menekan tombol submit. Kemudian data tersebut akan dilakukan pengecekan, apakah sudah sesuai dengan ketentuan. Jika iya data akan disimpan di database, jika tidak admin mendapatkan *feedback* berupa toast bahwa data invalid. Begitu pula dengan proses ubah data, admin menekan tombol edit syllable dan modal syllable akan ditampilkan. Admin dapat mengubah data pada modal tersebut dan menekan tombol edit, kemudian data dilakukan pengecekan serta disimpan di dalam database.



Gambar 3.12 Activity Delete Syllable

Admin dapat menghapus data kata donasi dengan cara menekan tombol delete pada halaman Syllable. Data yang sudah dihapus admin akan diubah dan disimpan ke database. Proses penghapusan data syllable menggunakan soft delete dimana data tersebut tetap ada di dalam database namun tidak ditampilkan di website.

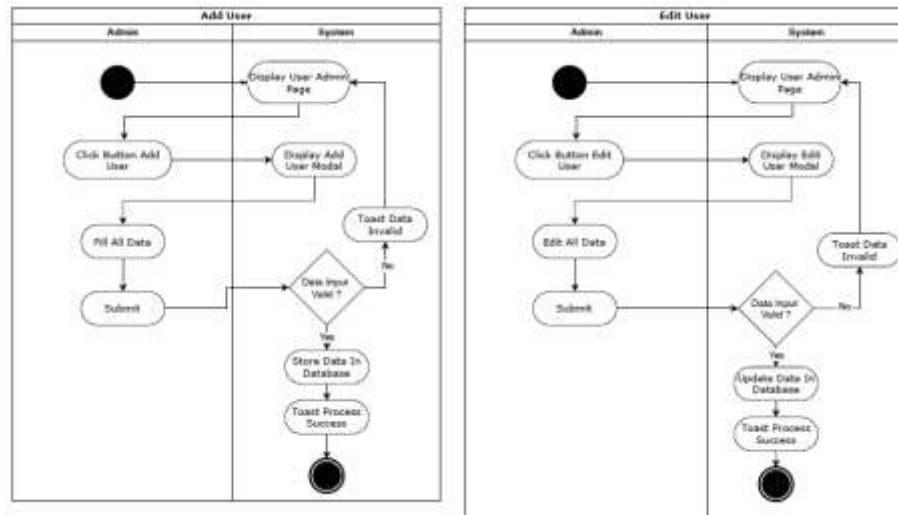
H. Create, Update, Delete, dan Approve User



Gambar 3.13 Activity Delete dan Approve User

Admin dapat menghapus pengguna dengan cara menekan tombol delete pada halaman Users, admin juga dapat memberikan hak akses kepada pengguna serta mengganti role pengguna dengan cara menekan tombol

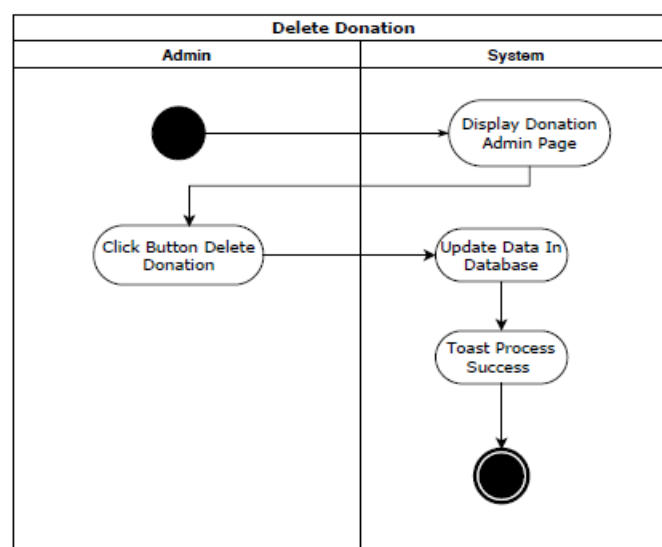
approve. Data pengguna yang telah dihapus dan disetujui akan disimpan di database.



Gambar 3.14 Activity Add dan Edit User

Admin dapat menambahkan atau mengubah data diri pengguna dengan cara menekan tombol add dan edit user. Admin dapat mengisi dan mengubah data pada add modal dan edit modal, data yang telah diisi dan diubah akan dilakukan pengecekan. Apabila data tersebut sesuai dengan ketentuan maka data akan disimpan di database dan pengguna mendapatkan *feedback* berupa toast.

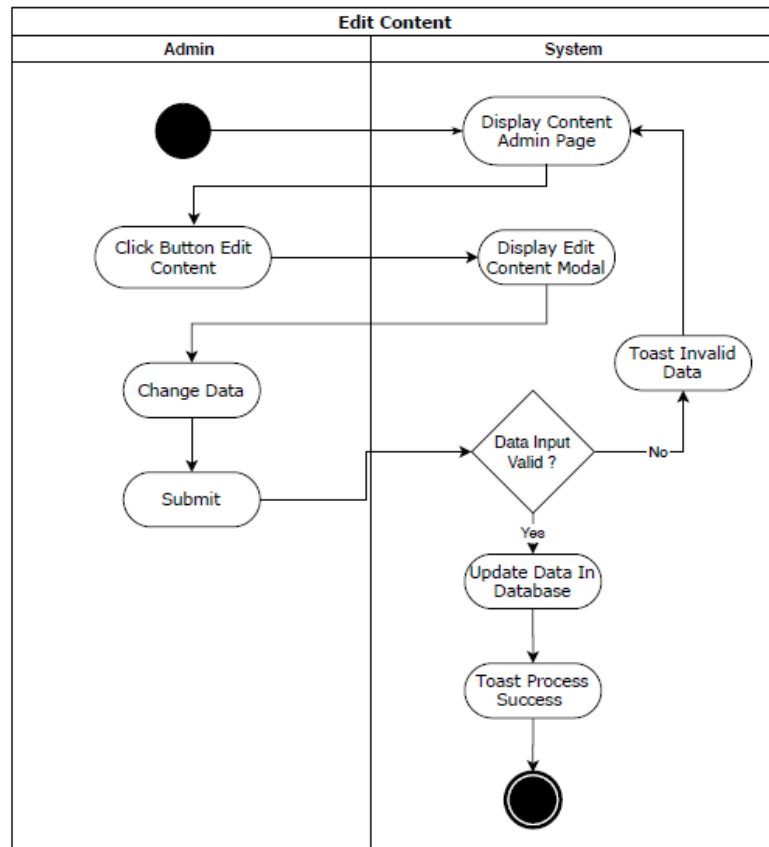
I. Delete Donation



Gambar 3.15 Activity Delete Donation

Gambar 3.15 menunjukkan diagram aktivitas dimana admin dapat menghapus data donasi pengguna dengan cara menekan tombol delete pada halaman Donation. Data donasi pengguna akan tetap ada di database namun tidak ditampilkan di website.

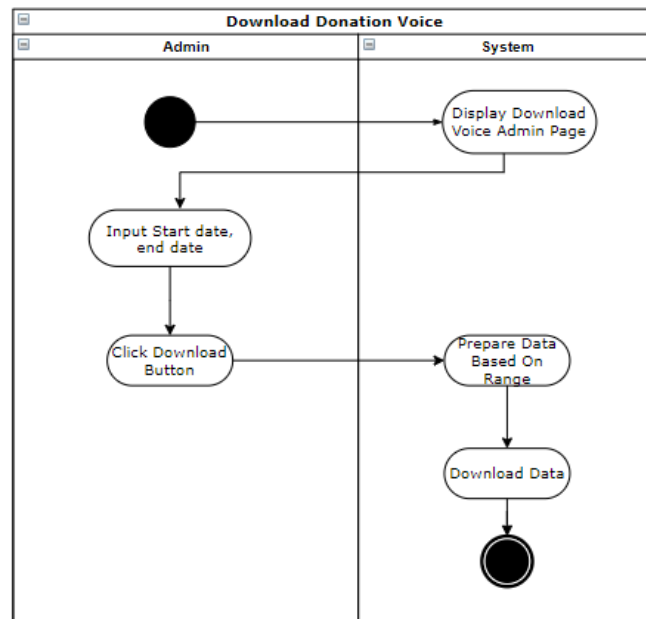
J. Update Content



Gambar 3.16 Activity Edit Content

Admin dapat mengubah konten yang terdapat pada website nihonesia. Admin dapat menekan tombol edit pada halaman Content dan sistem akan menampilkan modal edit. Admin mengisi data tersebut dan proses pengecekan data dilakukan. Apabila data telah sesuai dengan ketentuan, data akan disimpan ke dalam database. Namun apabila data tidak sesuai ketentuan admin akan mendapatkan toast yang menunjukkan bahwa data tidak valid.

K. Download Donation Voice



Gambar 3.17 Activity Download Voice

Gambar 3.17 menggambarkan aktivitas dimana admin dapat mengunduh data suara yang sudah didonasikan pengguna. Admin dapat memasukkan range tanggal, kemudian menekan tombol download. Website akan secara otomatis mengunduh data donasi tersebut.

3.3.5 Perancangan Tampilan Website

Dibawah ini merupakan perancangan tampilan website dari sisi pengguna dan sisi admin.



Gambar 3.18 Perancangan halaman awal

Tampilan awal ketika pengguna mengunjungi website ini. Tampilan ini berisi penjelasan singkat mengenai website ini. Terdapat menu di kanan atas, pengguna dapat melihat tampilan about dan team. Pengguna juga dapat melakukan login apabila sudah memiliki akun dan register jika belum memiliki akun.

A screenshot of a web browser displaying the 'Nihonesia' website. The browser's address bar shows 'https://nihonesia.co.id'. The website has a navigation menu with links for 'Home', 'About', 'Team', 'Login', and 'Register'. The 'Register' page is active, featuring a form with the following fields: 'Username', 'Password', 'Confirm Password', 'Secret Question' (a dropdown menu), and 'Secret Answer'. A 'Register' button is located at the bottom of the form.

Gambar 3.19 Perancangan halaman register

Gambar 3.19 merupakan tampilan register. Pada dasarnya tampilan register sudah disediakan oleh laravel, namun tampilan memiliki beberapa perubahan untuk menyesuaikan dengan website. Di tampilan ini berisi form untuk mengisi username, password, confirm password, dan secret answer. Secret question menggunakan dropdown button untuk membatasi variasi pertanyaan. Secret answer dan secret question digunakan untuk memvalidasi pengguna saat pengguna menekan forgot password.

A screenshot of a web browser displaying the 'Nihonesia' website. The browser's address bar shows 'https://nihonesia.co.id'. The website has a navigation menu with links for 'Home', 'About', 'Team', 'Login', and 'Register'. The 'Login' page is active, featuring a form with the following fields: 'Username' and 'Password'. Below the password field is a checkbox labeled 'Remember Me'. At the bottom of the form are two buttons: 'Login' and 'Forgot Your Password'.

Gambar 3.20 Perancangan halaman log in

Sama halnya dengan tampilan register, tampilan login juga sudah disediakan oleh laravel, tampilan ini diubah pada form masukan yaitu username. Tampilan juga disesuaikan dengan website. Pada tampilan ini pengguna dapat melakukan login dengan mengisi username dan password kemudian menekan tombol login.



The screenshot shows a web browser window with the address bar displaying 'https://nihonesia.co.id'. The page title is 'Nihonesia'. In the top right corner, there are navigation links: 'Home', 'About', 'Team', 'Login', and 'Register'. The main content area features a form titled 'Forgot Password'. Inside this form, there is a label 'Username' followed by a text input field. Below the input field is a button labeled 'Get Secret Question'.

Gambar 3.21 Perancangan halaman forgot password



The screenshot shows the same web browser window as the previous one, but the page content has changed. The main content area now features a form titled 'User Secret Question'. Inside this form, there is a label 'Secret Question' followed by a text input field. Below the input field is a button labeled 'Submit Answer'.

Gambar 3.22 Perancangan halaman forgot password (lanjutan)

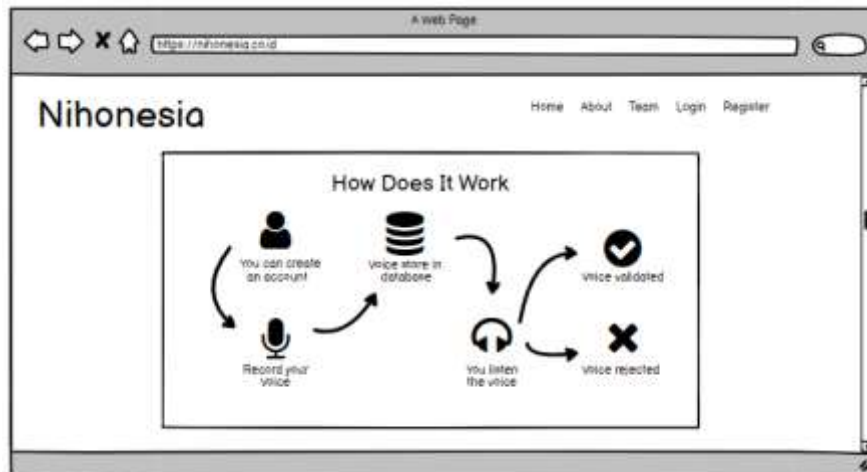


Gambar 3.23 Perancangan halaman forgot password (lanjutan)

Pada Gambar 3.21 sampai 3.23 merupakan tampilan ketika pengguna menekan tombol forgot password. Pengguna pertama kali akan di arahkan ke tampilan pada Gambar 3.21, pengguna diharuskan mengisi username. Kemudian pengguna akan di arahkan ke tampilan pada Gambar 3.22 dimana pengguna diharuskan mengisi secret answer. Apabila secret answer sesuai dengan data di dalam database maka pengguna akan diarahkan untuk mengisi kata sandi baru. Ketika pengguna sudah mengisi kata sandi tersebut pengguna akan diarahkan ke tampilan login.

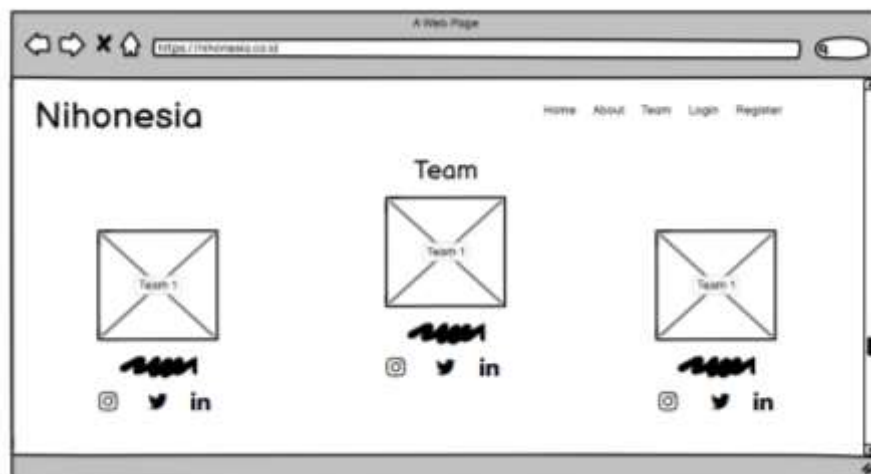


Gambar 3.24 Perancangan halaman about



Gambar 3.25 Perancangan halaman about (lanjutan)

Pada tampilan about yang dapat dilihat pada Gambar 3.24 dan 3.25 berisi penjelasan mengenai website ini, terdapat pula Gambar alur kerja website, sehingga pengguna mendapatkan informasi sederhana terkait website ini.



Gambar 3.26 Perancangan halaman team

Gambar 3.26 merupakan tampilan data diri singkat tim yang membuat website ini, serta kontak atau akun sosial media dari tim tersebut.

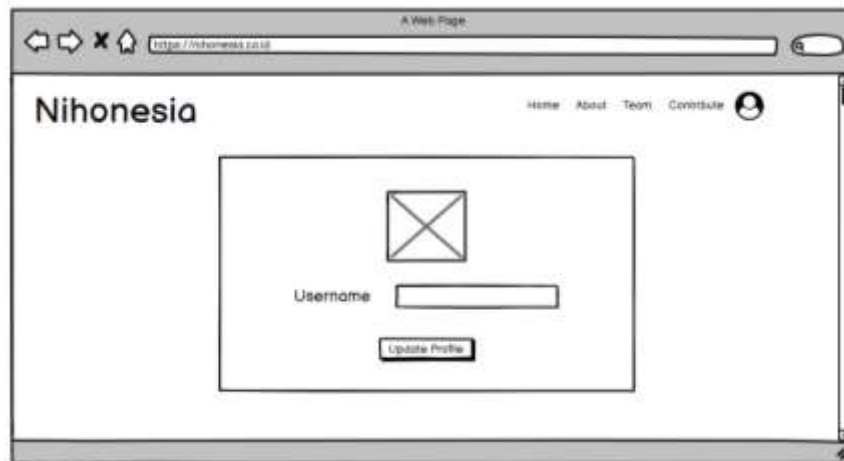


Gambar 3.27 Perancangan halaman utama (contributor)



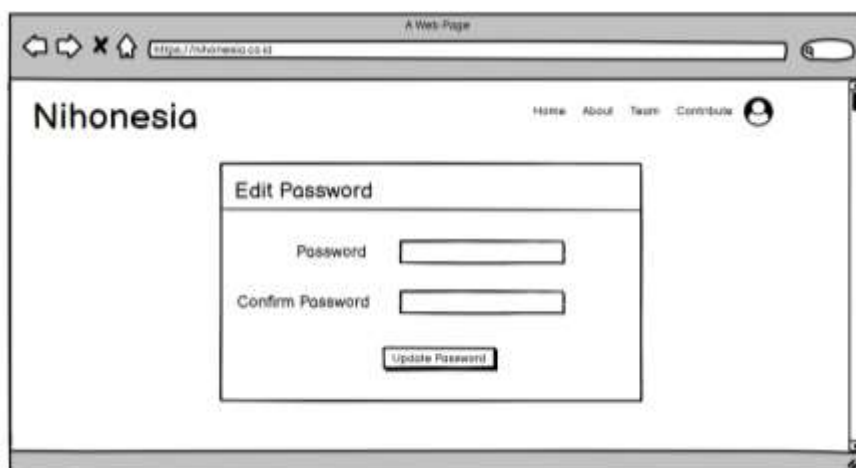
Gambar 3.28 Perancangan halaman utama (validator)

Gambar 3.27 dan Gambar 3.28 merupakan tampilan utama ketika pengguna telah melakukan login pada website. Terdapat alert pada tampilan utama, serta terdapat menu di kanan atas. Perbedaan dari kedua gambar adalah pada menu di kanan atas. Pengguna dengan role kontributor memiliki menu contribute, sedangkan pengguna dengan role validator memiliki menu validate. Terdapat pula profile pengguna, dimana ketika profile tersebut ditekan maka akan menampilkan daftar untuk mengubah data diri seperti mengubah kata sandi, username serta gambar profil atau log out.



Gambar 3.29 Perancangan halaman ubah profile

Pada Gambar 3.29 merupakan tampilan ubah profile, dimana pengguna dapat mengubah gambar profil serta username. Ketika salah satu atau semua data telah diubah maka pengguna dapat menekan tombol update profile.



Gambar 3.30 Perancangan halaman ubah kata sandi

Terdapat form berupa password dan confirm password pada tampilan ubah kata sandi. Tampilan ini memungkinkan pengguna untuk mengubah kata sandi. Terdapat tombol update password yang bisa digunakan pengguna untuk melakukan perubahan kata sandi. Data tersebut akan dikirim dan disimpan ke database.



Gambar 3.31 Perancangan halaman validate

Gambar 3.31 menampilkan halaman validate, halaman ini hanya bisa di akses pengguna dengan role validator. Terdapat kata bahasa jepang dan pelafalan serta arti kata bahasa jepang tersebut. Kemudian terdapat ikon play dan tombol yes atau no. Kata tersebut digunakan sebagai acuan pengguna dalam validasi suara. Ketika pengguna menekan ikon play, maka rekaman donasi akan dimainkan. Apabila pengguna yang mendengar suara rekaman tersebut sesuai dengan kata bahasa jepang maka pengguna dapat menekan tombol yes, namun apabila suara rekaman tidak sesuai maka pengguna dapat menekan tombol no. Pengguna akan mendapatkan *feedback* berupa toast bahwa rekaman suara telah berhasil divalidasi.



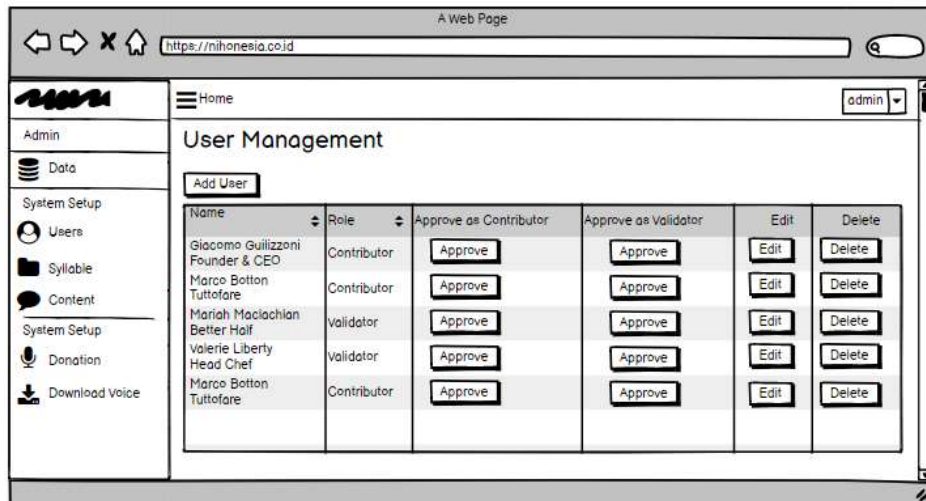
Gambar 3.32 Perancangan halaman contribute

Pengguna dengan role kontributor dapat mendonasikan suara mereka pada tampilan ini. Pada tampilan ini terdapat kata bahasa jepang dan cara membaca kata serta arti dari kata tersebut. Kata tersebut akan dibaca oleh pengguna sembari

merekam suara mereka. Pengguna juga dapat mendengar contoh kata tersebut dengan menekan tombol play di akhir kata. Ketika pengguna mendonasikan suara mereka, pengguna diharuskan menekan ikon mikrofon. Tampilan akan memunculkan tombol save apabila pengguna sudah merekam suara mereka selama 5 detik. Tombol tersebut akan mengirimkan data suara serta data lainnya untuk disimpan di database. Pengguna akan mendapatkan *feedback* berupa toast bahwa suara telah berhasil disimpan.

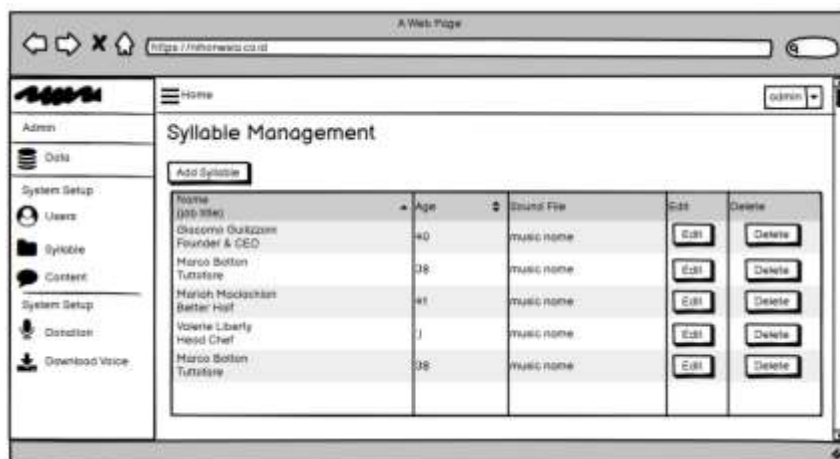
Gambar 3.33 Perancangan halaman admin

Gambar 3.33 merupakan tampilan halaman utama admin. Pada tampilan ini terdapat side menu, side menu ini akan mengarahkan admin ke management pengguna, syllable, konten, dll. Terdapat empat kotak pada tampilan ini yang berfungsi untuk menampilkan jumlah pengguna, jumlah data bahasa jepang, jumlah donasi, dan jumlah donasi yang sudah divalidasi. Tampilan ini juga berisi penjelasan atau daftar agenda admin.



Gambar 3.34 Perancangan halaman user management

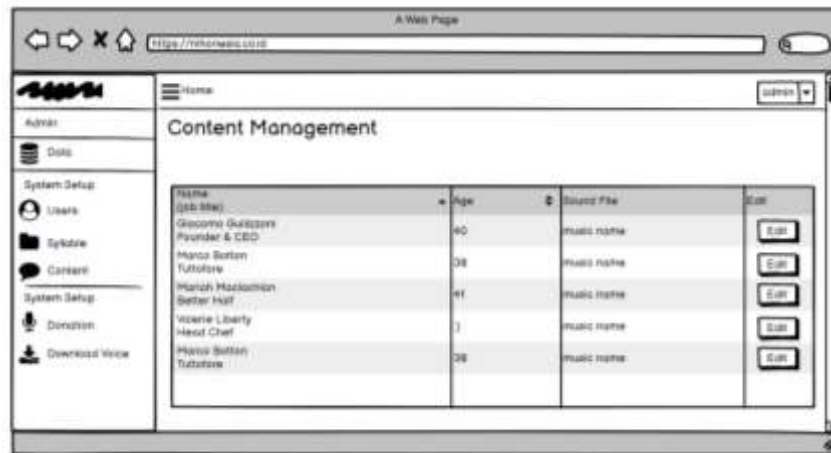
Tampilan pada Gambar 3.34 merupakan User Management, tampilan ini menampilkan daftar data diri pengguna dalam bentuk tabel. Terdapat tombol approve yang dapat digunakan admin untuk memverifikasi pengguna sesuai dengan role. Terdapat pula tombol edit dan delete. Tombol edit digunakan untuk mengubah data pengguna sedangkan tombol delete digunakan untuk menghapus pengguna. Di atas tabel terdapat tombol add user, tombol ini digunakan admin untuk menambahkan pengguna. Setiap kali admin melakukan perubahan website akan menampilkan toast sebagai *feedback*.



Gambar 3.35 Perancangan halaman syllable management

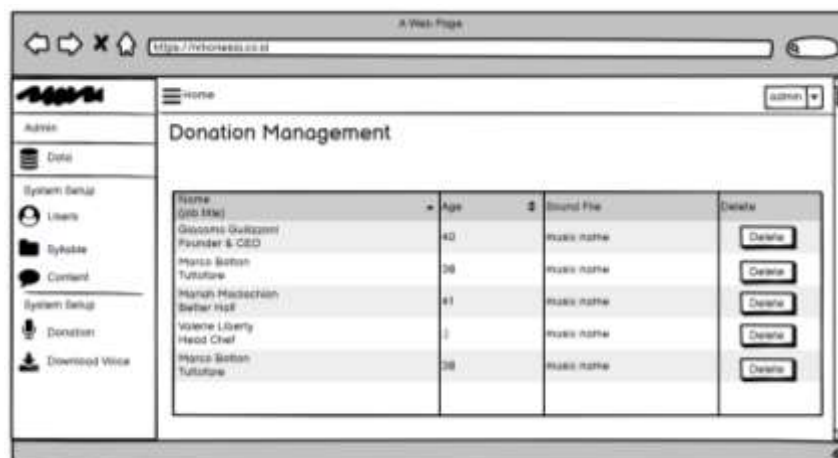
Terdapat daftar kata bahasa jepang dalam bentuk tabel pada tampilan Syllable Management. Terdapat pula tombol edit dan delete yang digunakan untuk mengubah dan menghapus data kata bahasa jepang. Admin juga dapat menambahkan data kata bahasa jepang dengan menekan tombol add syllable.

Perubahan yang dilakukan admin akan disimpan ke database serta *feedback* berupa toast.



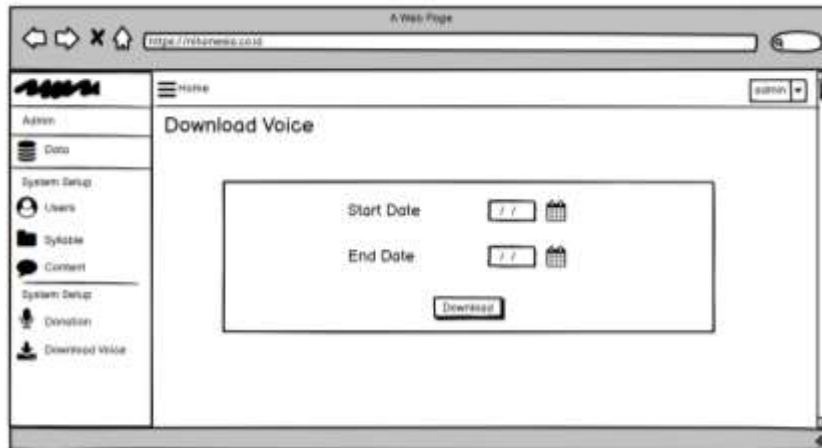
Gambar 3.36 Perancangan halaman content management

Gambar 3.36 menunjukkan tampilan Content Management, tampilan ini berisi daftar konten dalam bentuk tabel. Pada tampilan ini hanya terdapat tombol edit untuk mengubah data konten pada database. Data dalam konten ini akan ditampilkan pada halaman utama website.



Gambar 3.37 Perancangan halaman donation management

Tampilan Donation Management digunakan untuk menampilkan daftar data donasi pengguna. Data yang ditampilkan berupa kata bahasa jepang, username, tanggal donasi, dan status validasi, serta file suara donasi pengguna. Terdapat tombol delete pada setiap data donasi yang berfungsi untuk menghapus donasi ketika admin menekan tombol tersebut.



Gambar 3.38 Perancangan halaman download voice

Gambar 3.38 merupakan tampilan download voice management, terdapat form start date dan end date dengan tipe data tanggal. Admin akan memilih range tanggal ketika ingin mengunduh data donasi. Setelah mengisi tanggal, admin dapat menekan tombol download dan website akan menyiapkan data serta mengunduh data tersebut.

3.3.6 Kode Website Platform Pengumpulan Dataset

A. Register

```
public function register(Request $request)
{
    $this->validator($request->all())->validate();

    event(new Registered($user = $this->create($request->all())));

    $this->guard()->login($user);

    if ($response = $this->registered($request, $user)) {
        return $response;
    }

    return $request->wantJson()
        ? new Response('', 201)
        : redirect($this->redirectPath());
}
```

Gambar 3.39 Kode register users

```
protected function create(array $data)
{
    return User::create([
        'username' => $data['username'],
        // 'email' => $data['email'],
        'role' => "Contributor",
        'remember_token' => Str::random(60),
        'password' => Hash::make($data['password']),
        'secret_answer' => $data['secret_answer'],
        'secret_question' => $data['secret_question'],
    ]);
}
```

Gambar 3.40 Kode register users (lanjutan)

Pada umumnya proses login dan register telah disediakan oleh laravel, namun pada proses register terdapat beberapa perubahan agar register sesuai dengan ketentuan. Proses pada Gambar 3.39 yaitu data yang telah diisi pengguna akan dilakukan validasi input pada RegistersUsers.php, kemudian data hasil keluaran berupa array di kirim ke fungsi create yang dapat dilihat pada Gambar 3.40. Fungsi create akan memasukan data tersebut ke database, jika proses berhasil maka pengguna akan di arahkan ke tampilan utama.

B. Profile Management

```
// define relationship method
public function donations(){
    return $this->hasMany(Donations::class);
}
```

Gambar 3.41 Kode relation tabel users

Kode pada Gambar 3.41 menjelaskan tentang hubungan antara tabel user dengan tabel donations, dengan relation *one to many*. Maksud dari hasMany adalah satu pengguna memiliki banyak donasi.

```

// for direct to page edit profile
public function edit(Request $request){
    return view('edit',[
        'user' => $request->user()
    ]);
}

// for update profile
public function update(UpdateProfileRequest $request){
    $user = User::where('id', $request->id)->first();
    if($request->hasFile('avatar'))
    {
        $fileName = date('Ynd') . "-" . $request->avatar->getClientOriginalName();
        $request->avatar->storeAs('images', $fileName, 'public');
        $user->update([
            'imageUrl' => $fileName
        ]);
    }
    $user->update($request->all());
    return redirect()->route('homepage')->with('success_update', "Your profile has been updated");
}

```

Gambar 3.42 Kode ubah profile

Kegunaan fungsi edit pada Gambar 3.42 adalah mengarahkan tampilan website ke edit.blade.php sehingga pengguna dapat mengubah profile. Fungsi update digunakan untuk melakukan perubahan data dan menyimpannya di database. Hal pertama yang dilakukan adalah mencari data pada database yang sesuai dengan id *request*. Kemudian data tersebut dimasukan ke variabel *\$user*. Untuk membedakan setiap gambar profil pengguna, penyimpanan data gambar diberi nama tanggal dan nama gambar tersebut, kemudian disimpan di dalam storage php. Data tersebut disimpan ke dalam database. Setelah perubahan selesai dilakukan, maka pengguna akan diarahkan ke tampilan awal, tidak lupa pengguna akan mendapatkan *feedback* berupa toast yang menandakan bahwa perubahan profile berhasil dilakukan.

```
// direct to edit password view
public function editpass(Request $request){
    // dd($request->user);
    return view('editpas',[
        'user' => $request->user()
    ]);
}

// for update password
public function updatepass(Request $request)
{
    if($request->validate([
        'password' => ['required', 'string', 'min:8', 'confirmed'],
    ]))
    {
        $user = User::where('username', $request->username)->first();
        $user->password = Hash::make($request->password);
        $user->save();
    }
    return redirect('/')->with('success_update', "Your password has been changed");
}
```

Gambar 3.43 Kode ubah kata sandi

Ketika pengguna ingin mengubah kata sandi, maka fungsi `editpass` akan dipanggil. Fungsi ini dipakai untuk mengarahkan pengguna ke `editpas.blade.php`, dimana `editpas.blade.php` ini berisi tampilan untuk mengubah kata sandi. Ketika data sudah diisi maka fungsi `updatepass` akan melakukan validasi input serta mengambil data sesuai dengan *request* dan memasukan data tersebut ke variabel `$user`, jika data sudah valid maka kata sandi akan di *hash* dan disimpan ke database. Kemudian pengguna akan di arahkan ke tampilan awal dengan *feedback* berupa toast.

```
// Route untuk update profile
Route::group(['middleware' => 'auth'], function (){
    Route::get('profile', 'ProfileController@edit')->name('profile.edit');

    Route::patch('profile', 'ProfileController@update')->name('profile.update');

    Route::get('editpass', 'ProfileController@editpass')->name('profile.editpassword');

    Route::patch('editpass', 'ProfileController@updatepass')->name('profile.updatepassword');
});
```

Gambar 3.44 Route management profile

Potongan kode pada Gambar 3.44 digunakan sebagai route untuk menangani *request* di url dan mengarahkannya untuk menampilkan halaman atau data tertentu. Kode ini digunakan untuk mengarahkan pengguna ke halaman ubah profil dan ubah kata sandi serta memproses data tersebut untuk di simpan di database. Route pada Gambar 3.44 digabung ke

dalam grup dan menggunakan middleware untuk melakukan pengecekan apakah pengguna sudah login.

C. Syllable Management

```
use SoftDeletes;
use Blameable;

protected $fillable = ['in_jws', 'in_rws', 'bahasa_translation', 'sound_file_url', 'bahasa'];

protected $dates = ['deleted_at', 'created_at'];

protected $hidden = ['in_jws'];

// define relationship method
public function donations(){
    return $this->hasMany(Donations::class);
}
```

Gambar 3.45 Eloquent model tabel terms

Potongan kode pada Gambar 3.45 merupakan model dari tabel terms. Fungsi fillable pada kode diatas adalah atribut yang dapat diisi ketika memasukan atau mengubah data. Kemudian dates yang berfungsi untuk menambahkan tanggal secara otomatis ketika data dihapus dan ditambahkan. Sedangkan hidden digunakan untuk menyembunyikan atribut ketika memanggil fungsi tertentu. Pada fungsi donations juga terdapat hubungan antara tabel terms dengan tabel donations. Dimana relation ini adalah *one to many*. Sehingga dapat diartikan bahwa satu data terms memiliki banyak data donasi.

```
public function index()
{
    // check if user has role admin
    if(Auth::user()->role !== 'Admin'){
        abort(404);
    }else {
        // $terms = Terms::all()
        $terms = Terms::paginate(10);
        return view('administator.syllable', compact('terms'));
    }
}
```

Gambar 3.46 Fungsi index pada Syllable Controller

Fungsi index digunakan untuk mengarahkan admin ke tampilan syllable. Terdapat sebuah if yang digunakan untuk melakukan pengecekan bahwa pengguna tersebut memiliki role admin. Apabila pengguna tidak

memiliki role admin website akan menampilkan pesan error berupa 404 Not Found, namun jika pengguna merupakan admin maka data akan diambil dari database dengan menggunakan paginate. fungsi paginate yang digunakan sebagai lazy loader untuk meringankan kerja website. Semua data yang telah di ambil dari database akan dimasukan ke variabel terms dan dikirimkan ke tampilan syllable dengan fungsi compact.

```
public function store(Request $request)
{
    // Untuk validasi input
    $this->validator($request->all())->validate();
    // dd($request->all());
    $fileName = 'sound_' . $request->fileupload->getClientOriginalName();
    Terms::create([
        'in_jws' => $request->jws,
        'in_rws' => $request->rws,
        'bahasa_translation' => $request->bahasa_translation,
        'bahasa' => $request->bahasa,
        'sound_file_url' => $fileName,
    ]);
    // Disimpan di File storage
    $request->fileupload->storeAs('sound', $fileName, 'public');
    $terms = Terms::where('in_jws', $request->jws)->first();
    $terms->save();
    $terms = Terms::paginate(10);

    return view('administrator.syllable', compact('terms'))->with('success_crud', 'Your Terms Has Been Created');
}
```

Gambar 3.47. Fungsi store pada syllable management

Fungsi store digunakan untuk menyimpan data terms yang telah ditambahkan admin. Sebelum data dimasukan tidak lupa untuk melakukan validasi input. Kemudian data suara disimpan di storage dan data lainnya akan disimpan di database. Ketika proses penyimpanan ke database selesai maka website akan menampilkan data terms dengan fungsi paginate dan mengarahkan tampilan ke syllable management.

```
public function softdeleteterms(Request $request){
    $deleted = Terms::where('in_jws', $request->in_jws)->first();
    $deleted->delete(); // for delete data
    $deleted->save(); // for deleted_by
    return back()->with("success_crud", "Your Terms Has Been Deleted");
}
```

Gambar 3.48. Fungsi soft delete pada syllable management

Fungsi softdeleteterms digunakan untuk menghapus data terms dengan soft delete. Soft delete ini hanya mengubah data dengan mengisi data pada deleted_at dan tidak terhapus dari database. Ketika data sudah diubah maka admin akan diarahkan ke tampilan sebelumnya dengan *feedback* berupa toast.

```

public function updatesyllable(Request $request){
    // for input validation
    $this->validatedit($request->all())->validate();

    $editterms = Terms::where('id', $request->id)->first();
    $editterms->update([
        'in_jws' => $request->jws,
        'in_rws' => $request->rws,
        'bahasa_translation' => $request->bahasa_translation,
        'bahasa' => $request->bahasa,
    ]);

    // Untuk cek apakah ada file yg di ubah
    if($request->hasFile('fileupload'))
    {
        $fileName = "sound_" . $request->fileupload->getClientOriginalName();
        $request->fileupload->storeAs('sound', $fileName, 'public');
        $editterms->update([
            'sound_file_url' => $fileName
        ]);
    }
    $editterms->save();
    return back()->with('success_crud', 'Your Terms Successfully Updated!');
}

```

Gambar 3.49. Kode untuk mengubah data donasi

Fungsi pada Gambar 3.49 digunakan untuk mengubah data terms apabila admin melakukan perubahan data. Data yang didapat akan dilakukan validasi input, kemudian dengan fungsi update maka data akan diubah sesuai dengan id terms. Untuk data suara, dilakukan pengecekan terlebih dahulu untuk mengetahui apakah ada data suara yang diubah. Jika iya maka data akan disimpan di storage dan di database. Tidak lupa untuk mengarahkan admin ke tampilan sebelumnya dan memberikan *feedback* berupa toast.

```

class BlameableObserver
{
    public function updating(Terms $model)
    {
        $model->updated_by = Auth::user()->id;
    }

    public function creating(Terms $model)
    {
        $model->created_by = Auth::user()->id;
        $model->updated_by = Auth::user()->id;
    }

    public function deleting(Terms $model){
        $model->deleted_by = Auth::user()->id;
    }
}

```

Gambar 3.50 Observer pada tabel terms

Saat admin mengubah, menghapus, maupun menambahkan data ke tabel terms. Atribut deleted_by, created_by, dan updated_by akan secara otomatis terisi tanpa perlu di program. Hal ini terjadi karena penggunaan observer pada tabel terms. Setiap kali ada perubahan pada tabel terms, maka observer akan mengidentifikasi perubahan tersebut. Sebagai contoh ketika data terms dihapus, maka observer akan memanggil fungsi deleting dan secara otomatis mengisi atribut deleted_by dengan id user. Begitu pula dengan create dan update.

```

// Route untuk syllable
Route::resource('/syllable', 'SyllableController');
// route untuk softdelete syllable
Route::post('/deleteterms', 'SyllableController@softdeleteterms')->name('deleteterms');
// route untuk update syllable
Route::post('/update', 'SyllableController@updatesyllable')->name('updateterms');

```

Gambar 3.51 Route syllable management

Potongan kode pada Gambar 3.51 adalah route untuk memanggil tampilan atau proses tertentu. Pada route deleteterms berfungsi untuk menghapus data terms, route update berfungsi untuk mengubah data pada tabel terms dan route syllable berfungsi untuk menampilkan serta menambahkan data pada tabel terms.

D. Donation Management

```
use SoftDeletes;

protected $fillable = ['donation_file_url', 'user_id', 'terms_id'];

//define inverse relationship method
public function users(){
    return $this->belongsTo(User::class, 'user_id');
}

//define inverse relationship method
public function terms(){
    return $this->belongsTo(Terms::class);
}
```

Gambar 3.52 Eloquent model tabel donations

Gambar 3.52 merupakan potongan kode eloquent model dari tabel donations. Atribut yang dapat diisi oleh pengguna adalah `donation_file_url`, `user_id`, dan `terms_id`. Terdapat dua relation pada tabel donations, yaitu hubungan terhadap tabel user dan hubungan terhadap tabel terms. Arti `belongsTo` pada kode di atas adalah relation *many to one*, dimana banyak donasi dimiliki oleh satu terms dan banyak donasi dimiliki oleh satu users.

```
public function index()
{
    // $donation = Donations::with('users')->with('terms')->get(); udah oke
    $donation = Donations::with('users')->with('terms')->paginate(10);
    return view('administator.donation', compact('donation'));
}

public function destroy(Request $request, $id){
    $donation = Donations::where('id', $request->id)->first();
    $donation->deleted_by = Auth::user()->id;
    $donation->delete();
    $donation->save();
    return back()->with("success_crud", "This Donations Has Been Deleted");
}
```

Gambar 3.53 Fungsi index dan soft delete pada donations management

Fungsi index berguna untuk mengarahkan admin ke tampilan donations, selain itu proses pengambilan data juga dilakukan. Proses pengambilan data dilakukan dengan lazy loader. Pada fungsi destroy, data donasi dihapus secara soft delete dari database. Kemudian admin akan mendapatkan *feedback* berupa toast proses berhasil dijalankan.

```
// Route untuk donation
Route::resource('/donation','DonationController');
```

Gambar 3.54 Route donations management

Potongan kode pada Gambar 3.54 adalah route untuk donations. Dimana route yang ditunjukkan pada sub-URL ‘/donation’ akan diproses oleh *function* yang terdapat pada DonationController.

E. Content Management

```
class Contents extends Model
{
    //
    use SoftDeletes;

    protected $fillable = ['reference_key', 'value', 'created_by', 'updated_by'];

    protected $dates = ['deleted_at', 'created_at', 'updated_at'];
}
```

Gambar 3.55 Eloquent model data tabel contents

Gambar 3.55 merupakan eloquent model dari tabel content. Reference_key merupakan penanda untuk setiap konten pada website, value berisi konten website, created_by dan updated_by berisi id admin yang mengubah atau membuat konten tersebut. Semua data tersebut merupakan eloquent variabel fillable. Variabel dates berisi atribut deleted_at, created_at, dan updated_at yang nantinya atribut tersebut akan berisi data tanggal.

```

public function index()
{
    // check user has role admin
    if(Auth::user()->role !== 'Admin'){
        abort(404);
    } else {
        $content = Contents::all(); // coba doang
        // dd($terms);
        return view('administator.content', compact('content'));
    }
}

public function contentupdate(Request $request)
{
    $editcontent = Contents::where('id', $request->id)->first();
    $editcontent->update([
        'reference_key' => $request->rkey,
        'value' => $request->value,
        'updated_by' => Auth::user()->id,
    ]);
    $editcontent->save();
    // redirect
    return back()->with('success_crud', "This Content Has Heen Updated.");
}

```

Gambar 3.56 Fungsi index dan ubah konten

Fungsi index digunakan untuk mengarahkan admin ke tampilan content, tidak lupa untuk melakukan pengecekan role. Jika role pengguna bukan merupakan admin maka pengguna akan diarahkan ke tampilan 404 Not Found. Sedangkan admin akan diarahkan ke tampilan Content Management. Fungsi contentupdate digunakan untuk memproses perubahan pada data konten. Editcontent merupakan variabel yang akan menyimpan data yang sesuai dengan id *request*. Kemudian proses perubahan data di lakukan dan disimpan di database. Setelah itu, admin akan diarahkan ke tampilan sebelumnya dengan memberikan toast sebagai *feedback*.

```

function get_username($refkey) {
    $data = DB::Table('contents')->select('value')->where
    ('reference_key',$refkey)->pluck('value')->first();
    // $d = $data->value;
    return $data;
}

```

Gambar 3.57 Helper pada contents management

Isi dari konten tidak di proses pada ContentController melainkan melalui helper. Fungsi helper ini berguna untuk mendapatkan isi konten

yang sesuai dengan `reference_key` yang telah ditentukan. Penggunaan helper berfungsi untuk memudahkan programmer dalam membuat kode yang dapat digunakan kembali atau *reusable code*. Setiap kali website dibuka maka helpers akan berjalan di latar belakang dan mengambil serta menampilkan data yang telah di dapat.

```
// Route untuk content
Route::resource('/content', 'ContentController')->only(['store', 'edit',
'destroy', 'index']);
Route::post('/editcontent', 'ContentController@contentupdate')->name
('updatecontent');
```

Gambar 3.58 Route contents management

Route pada Gambar 3.58 adalah route untuk konten serta route untuk mengubah konten dengan nama route `updatecontent` dan proses dilakukan pada `ContentController`.

F. Contribute

```
public function index()
{
    // Cek apakah user status udah di acc
    if(Auth::user()->is_locked === 1 || Auth::user()->role !== "Contributor"){
        return redirect("/")->with('not_authorized', "Sorry you don't have access");
    }
    else {
        $terms = Terms::inRandomOrder()->first(); // random data
        return view('layouts.contribute', compact('terms'));
    }
}
```

Gambar 3.59 Fungsi index pada Contribute Controller

Potongan kode pada Gambar 3.59 digunakan untuk mengarahkan pengguna ke tampilan donasi atau contribute. Sebelum diarahkan, dilakukan pengecekan hak akses pengguna serta role pengguna. Apabila role pengguna bukan Kontributor atau pengguna tidak memiliki akses maka pengguna akan mendapatkan *feedback* berupa toast peringatan bahwa pengguna tidak memiliki akses ke tampilan tersebut. Namun apabila pengguna memenuhi kriteria tersebut maka pengguna akan diarahkan ke tampilan contribute untuk melakukan donasi.

```

public function saveAudio(Request $request)
{
    // Untuk input data sound
    $files = $request->file('audio');
    $fileName = Auth::user()->username . "_" . $request->rws . "_" . date('Ymd_his') . ".mp3";
    Donations::create([
        'user_id' => Auth::user()->id,
        'terms_id' => $request->id,
        'donation_file_url' => $fileName,
    ]);
    $files->storeAs('sound', $fileName, 'public');
    // $files->storeAs('sound', 'test.mp3', 'public');
    return view('auth.login');
    // $user->save();
}

```

Gambar 3.60 Fungsi untuk menyimpan donasi

Fungsi saveAudio digunakan untuk menyimpan suara donasi pengguna. Variabel files akan menyimpan data suara dan penamaan file suara adalah gabungan dari nama pengguna, rws terms, serta tanggal donasi tersebut. Data akan disimpan ke tabel donations dan file suara akan disimpan pada storage php. Kemudian pengguna akan diarahkan ke tampilan sebelumnya.

G. Toast

```

{{-- Toastr --}}
<script src="https://cdnjs.cloudflare.com/ajax/libs/toastr.js/latest/js/toastr.min.js"></script>
@if(Session::has('success_crud'))
    <script>
        toastr.success("{!!Session::get('success_crud')!!}");
    </script>
@endif

```

Gambar 3.61 Javascript untuk toast

Toast pada beberapa tampilan menggunakan toastr, toastr merupakan javascript library untuk notifikasi. Pada tampilan ini library diambil dari cdn. Kemudian setiap kali website memiliki session tertentu maka toast akan ditampilkan. Toast ini digunakan sebagai *feedback* untuk pengguna dan admin setelah kondisi tertentu.

3.3.7 Implementasi

A. Tampilan Utama



Gambar 3.62 Hasil tampilan halaman awal

Hasil tampilan utama website, dengan beberapa menu di bagian kanan atas, tampilan ini adalah tampilan saat pengguna mengunjungi website sebelum login. Pada tampilan utama, terdapat deskripsi singkat mengenai website tersebut.

B. About



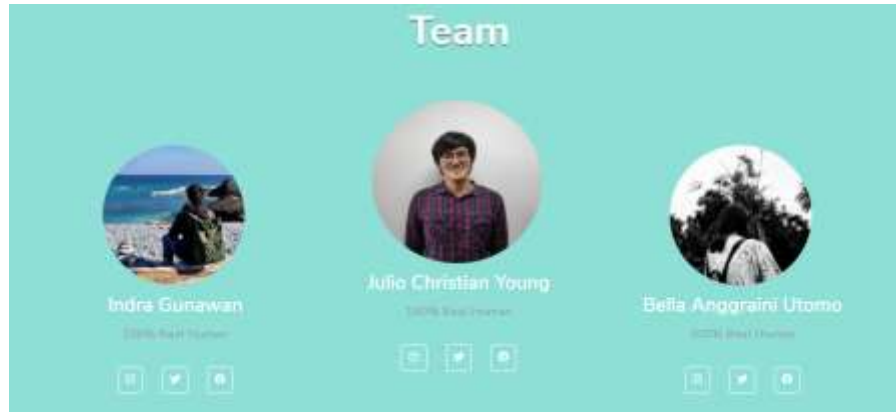
Gambar 3.63 Hasil tampilan about



Gambar 3.64 Hasil tampilan about (lanjutan)

Tampilan about, berisi penjelasan mengenai website ini. Pengguna juga dapat melihat bagaimana alur kerja website ini serta bagaimana data donasi suara di simpan dan di proses. Tampilan about dapat dilihat dengan cara menekan menu about di kanan atas atau *scroll* tampilan website utama.

C. Team

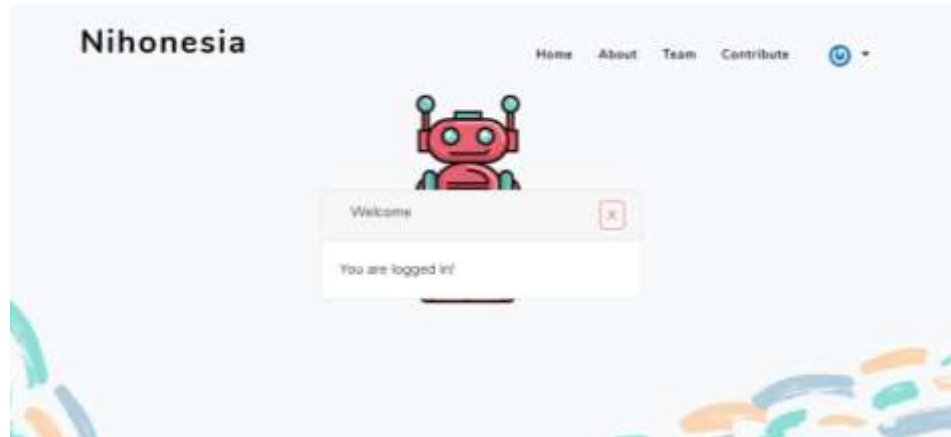


Gambar 3.65 Hasil tampilan team

Tampilan team berisi data diri singkat pembuat website. Pada tampilan ini terdapat foto tim yang membuat website serta data diri singkat. Di tampilan ini juga terdapat kontak media sosial yang dapat dihubungi.

D. Register

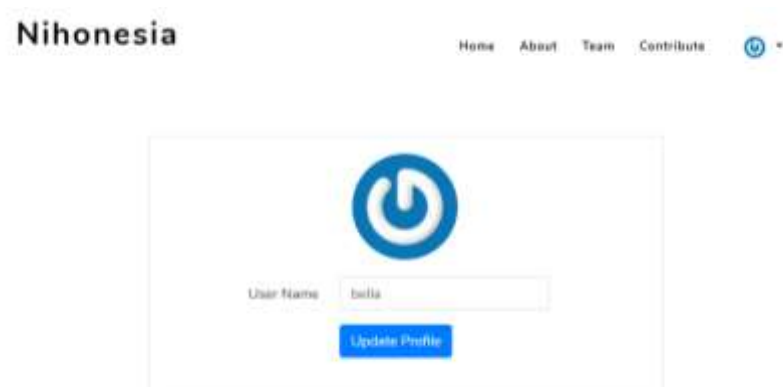
Gambar 3.66 Hasil tampilan register



Gambar 3.67 Hasil tampilan utama

Hasil tampilan form register dapat dilihat pada Gambar 3.66 yaitu form dengan username, password, confirm password dan secret answer dalam bentuk input text dan secret question menggunakan dropdown button. Ketika pengguna menekan tombol register maka data yang sudah diisi akan di kirim ke controller untuk di validasi dan disimpan ke database. Menu yang terdapat di kanan atas difungsikan untuk pengguna yang ingin berpindah halaman. Apabila pengguna telah berhasil melakukan register maka pengguna akan diarahkan ke tampilan utama dengan *feedback* berupa pemberitahuan selamat datang. Menu pada bagian kanan atas memiliki perbedaan sesuai dengan role pengguna. Tampilan utama dapat dilihat pada Gambar 3.67.

E. Edit Profile



Gambar 3.68 Hasil tampilan ubah profile



Gambar 3.69 Hasil toast setelah mengubah profile

Gambar 3.68 merupakan tampilan ubah profile. Pengguna dapat mengubah gambar profil dengan menekan lingkaran yang terdapat di atas form username dan pengguna juga dapat mengubah username. Ketika pengguna menekan tombol update profile, data yang sudah diubah akan dikirimkan dan disimpan di database, kemudian pengguna akan mendapatkan *feedback* berupa toast bahwa profile telah berhasil di update. Tampilan toast dapat dilihat pada Gambar 3.69.

F. Edit Password

Gambar 3.70 Hasil tampilan ubah kata sandi



Gambar 3.71 Hasil toast setelah mengubah kata sandi

Pengguna dapat mengubah kata sandi dengan cara menekan tombol edit password pada dropdown profile. Tampilan ubah kata sandi dapat dilihat pada Gambar 3.70. Pada tampilan ubah kata sandi terdapat new password dan confirm new password. Ketika pengguna menekan tombol change password, data akan dikirim ke controller dan dilakukan pengecekan serta hash kata sandi, kemudian pengguna akan di arahkan ke tampilan utama dengan *feedback* berupa toast yang dapat dilihat pada Gambar 3.71.

G. Contribute



Gambar 3.72 Hasil tampilan contribute



Gambar 3.73 Hasil tampilan contribute setelah merekam selama 5 detik

Pada Gambar 3.72 merupakan tampilan contribute, dimana pada tampilan ini pengguna dapat mendonasikan suara mereka dengan cara menekan ikon mikrofon. Ketika pengguna menekan ikon mikrofon, website akan merekam suara pengguna selama 5 detik. Kemudian tombol save dan hasil rekaman suara akan ditampilkan, tampilan ini dapat dilihat pada Gambar 3.73. Pengguna dapat menekan tombol save untuk menyimpan rekaman atau menekan tombol mikrofon untuk mengulang rekaman. Data suara akan disimpan ke database. Pengguna juga dapat mendengar contoh pengucapan kata bahasa jepang dengan menekan ikon play yang terletak di akhir kata.

H. Validate



Gambar 3.74 Hasil tampilan validate

Terdapat kata bahasa jepang, pelafalan dan arti dari kata tersebut pada tampilan validate. Tampilan ini hanya bisa diakses oleh pengguna

yang memiliki role validator. Pada tampilan ini terdapat ikon play, ikon ini berfungsi untuk memutar rekaman suara dari pengguna yang sudah mendonasikan suara mereka. Apabila suara tersebut sesuai dengan kata bahasa jepang, pengguna dapat menekan tombol yes. Namun, jika suara tidak sesuai maka pengguna dapat menekan tombol no. Kemudian data akan dikirim ke controller dan disimpan di database.

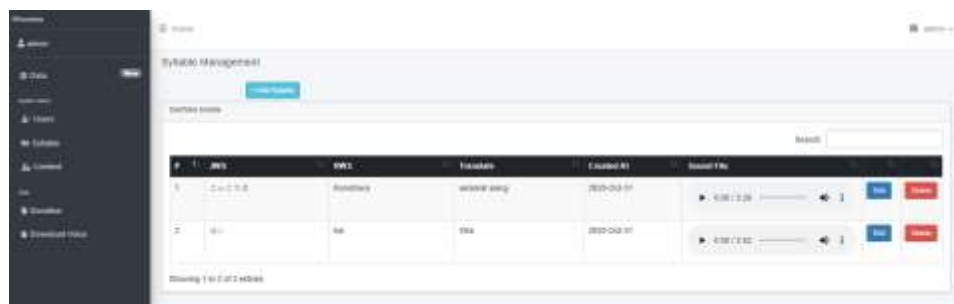
I. Tampilan Utama Admin



Gambar 3.75 Hasil tampilan halaman utama admin

Gambar 3.75 merupakan hasil tampilan dashboard atau tampilan utama admin. Terdapat empat kotak yang masing–masing menampilkan jumlah data yang terkumpul. Pada tampilan ini juga terdapat persegi panjang yang digunakan untuk meletakkan agenda atau tugas admin. Persegi panjang ini berfungsi sebagai pengingat agenda admin. Selain itu, pada tampilan ini juga terdapat side menu yang akan mengarahkan admin ke management website.

J. Syllable Management



Gambar 3.76 Hasil tampilan syllable management

Edit Syllable

JWS

RWS

Translate

Bahasa

File Input

Gambar 3.77 Modal untuk mengubah data syllable

Add New Syllable

JWS

RWS

Translate

Kategori Bahasa

File Input

Gambar 3.78 Modal untuk menambahkan data syllable

Gambar 3.76 merupakan tampilan pada Syllable Management, terdapat tabel yang berisi daftar data bahasa jepang. Terdapat pula tombol edit dan delete pada masing – masing data. Tombol edit digunakan untuk mengubah data bahasa jepang, sedangkan tombol delete digunakan untuk menghapus data. Untuk dapat mengubah data bahasa jepang, admin dapat menekan tombol edit kemudian modal edit syllable yang telah berisi data akan ditampilkan, modal edit syllable dapat dilihat pada Gambar 3.77. Ketika pengguna telah selesai mengubah data dan menekan tombol update profile maka data yang diubah akan diproses dan disimpan ke database, tidak lupa

toast ditampilkan sebagai *feedback*. Apabila admin menekan tombol delete maka data akan terhapus. Admin dapat pula menambahkan data bahasa jepang dengan cara menekan tombol add syllable pada bagian atas daftar tabel. Form dalam Add modal akan ditampilkan, kemudian admin dapat mengisi form tersebut. Admin dapat menekan tombol add syllable untuk menyimpan data yang telah diisi ke dalam database. Add modal dapat dilihat pada Gambar 3.78.

K. Content Management

[illegible]

Gambar 3.79 Hasil tampilan content management

Edit Content

Reference_Key

about long

Value

he voice issued by humans is a medium for communicating with fellow humans. The sound issued is basically unique to each individual.

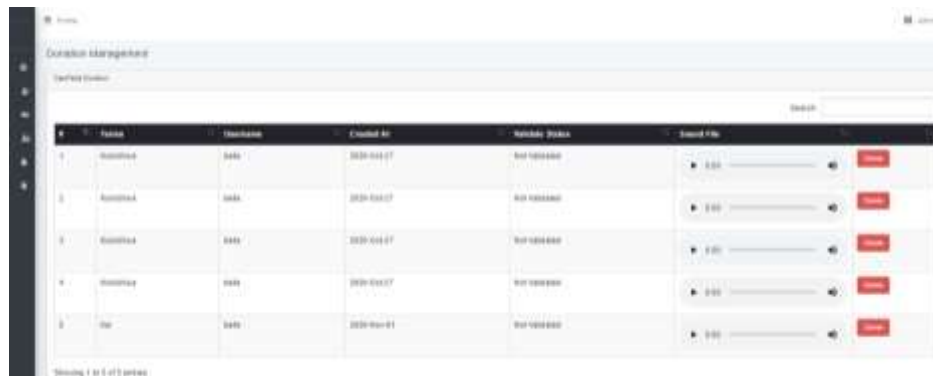
Edit Content

Gambar 3.80 Modal untuk mengubah data konten

Tampilan konten merupakan salah satu tampilan yang dapat dilihat oleh admin. Terdapat data konten yang dijabarkan melalui tabel, terdapat pula tombol edit untuk mengubah isi konten. Edit modal akan ditampilkan apabila admin menekan tombol edit, pada modal ini data yang dapat diubah

adalah value atau isi dari konten tersebut. Data dengan atribut `reference_key` tidak dapat diubah. Admin akan menekan tombol edit content dan proses penyimpanan akan dijalankan di latar belakang. Tidak lupa sebagai *feedback* setiap perubahan data, website akan menampilkan toast.

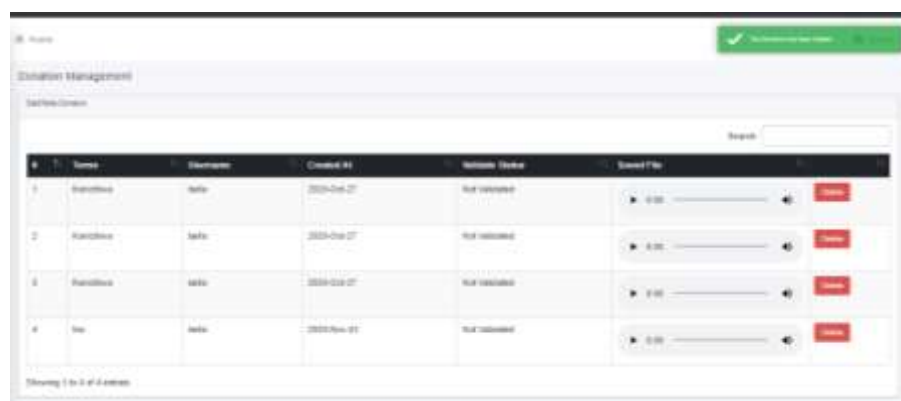
L. Donation Management



The screenshot shows a web application interface for 'Donation Management'. It features a table with the following columns: 'No', 'Status', 'Donor Name', 'Created At', 'Remaining Balance', and 'Amount Due'. There are five rows of data. Each row has a 'Delete' button on the right. The status of the first four rows is 'Completed', and the fifth is 'Due'. The 'Created At' dates are 2020-09-07 for the first four and 2020-09-01 for the last. The 'Remaining Balance' is 'Not calculated' for all. The 'Amount Due' is 0.00 for all, with a slider control and a 'Delete' button for each.

No	Status	Donor Name	Created At	Remaining Balance	Amount Due
1	Completed	John	2020-09-07	Not calculated	0.00
2	Completed	John	2020-09-07	Not calculated	0.00
3	Completed	John	2020-09-07	Not calculated	0.00
4	Completed	John	2020-09-07	Not calculated	0.00
5	Due	John	2020-09-01	Not calculated	0.00

Gambar 3.81 Hasil tampilan donations management



Gambar 3.82 Toast saat donasi di hapus

Gambar 3.81 merupakan tampilan data donasi dalam bentuk tabel. Setiap data terdapat tombol delete yang berfungsi untuk menghapus data donasi tersebut. Apabila data donasi di hapus maka admin akan mendapatkan *feedback* berupa toast yang terdapat di kanan atas.

3.3.8 Kendala Yang Ditemukan

Kendala yang ditemukan selama kerja magang adalah sebagai berikut:

1. Kurangnya pemahaman mengenai bahasa yang digunakan dalam pembuatan website sehingga dibutuhkan usaha untuk memahami maksud dari kode tersebut

2. Kurangnya pemahaman dan pengalaman dalam penggunaan Git sehingga terdapat beberapa kesalahan serta beberapa potongan kode yang tidak sinkron satu sama lain.
3. Kondisi saat ini yang mengharuskan bekerja dari rumah, sehingga kerja magang sangat bergantung pada kecepatan internet di rumah.

3.3.9 Solusi Atas Kendala Yang Ditemukan

Berdasarkan kendala yang ditemukan selama kerja magang, berikut adalah solusi yang dapat digunakan untuk mengatasi kendala tersebut.

1. Mengumpulkan, mencari, dan mempelajari kembali informasi atau materi baik dari internet, buku, atau contoh aplikasi yang terkait dengan bahasa pemrograman yang digunakan. Serta dapat bertanya kepada rekan kerja untuk mencari pembelajaran.
2. Memastikan kembali kode yang telah dibuat serta pembuatan repository yang rapih sehingga memudahkan dalam mengatasi kesalahan atau hilangnya kode.
3. Membeli paket data tambahan atau mencari tempat yang memiliki internet yang kuat, tidak lupa untuk menerapkan protokol kesehatan di tempat umum.