

Bank sampah ialah suatu terobosan yang berfokus pada pengelolaan sampah rumah

tangga. Dimana sampah rumah tangga tersebut mampu di kelola dan memiliki timbal

balik yang menguntungkan baik bagi rumah tangga yang mengeluarkan sampah,

maupun pemerintah atau pengelola sampah. dengan hadirnya terobosan bank sampah

ini diharapkan sampah-sampah rumah tangga dapat menguntungkan berbagai

pihak.

Berdirinya bank sampah ini terpicu karena keprihatinan dan ketidaknyamanan

masyarakat terhadap lingkunganya yang dari hari ke hari makin banyak sampah yang

menumpuk, dikarenakan pembersihan yang dilakukan lebih sedikit dengan sampah

yang dihasilkan, terutama dalam sampah rumah tangga. Oleh karena itu bank sampah

hadir dengan maksud untuk memilah dan memilih(yang masih bisa digunakan,

di daur ulang, atau masih bermanfaat) sampah-sampah rumah tangga tersebut.

Panduan Cerdas Mengenal Aplikasi Bank Sampah.

By Arrizal Furqona Gifary dan Bakti Qilan Mufid Copyright

c 2020 Kreatif Industri Nusantara.

1.2 Definisi Bank Sampah

Bank sampah adalah suatu tempat yang digunakan untuk mengumpulkan sampah

yang sudah di filter. Hasil dari pengumpulan tersebut akan disetorkan ke tps(untuk

sampah yang tidak dapat digunakan kembali) atau ke tempat pembuatan kerajinan

dari barang-barang bekas. Bank sampah dikelola menggunakan sistem seperti perbankan

yang dilakukan oleh petugas. Penyetor atau nasabah ialah warga masyarakat

yang berada di sekitar lokasi bank yang dimana nasabah akan mendapatkan rekening

seperti halnya rekening pada bank.[?]

Menurut peraturan menteri negara lingkungan hidup RI nomor 13 tahun 2012

tentang pedoman pelaksanaan Reduce, Reuse, dan Recycle melalui bank sampah,

yang di atur dalam pasa 1 ayat 2 dengan bunyi yaitu: Bank sampah adalah tempat

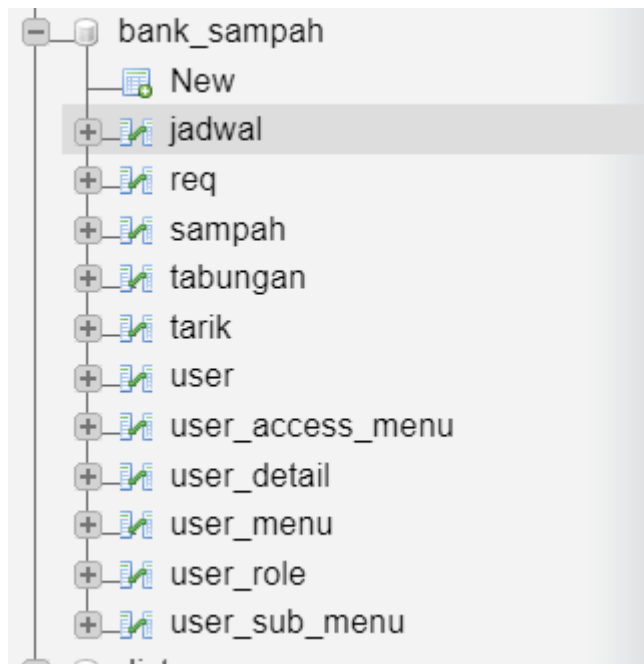
pemilahan dan pengumpulan sampah yang dapat didaur ulang dan/atau diguna ulang

yang memiliki nilai ekonomi.[?]

Langkah-langkah membuat sebuah aplikasi bank sampah menggunakan codeigniter dan android

Langkah awal dari pembuatan aplikasi bank sampah yaitu membuat sebuah database terlebih dahulu pada <http://localhost/phpmyadmin/index.php> disini kita membuat
















sebuah nama databasenya adalah bank sampah, dalam database bank_sampah kita sudah membuat beberapa table yang dapat berelasi dari satu table ke table yang lainnya. Bisa kita lihat pada gambar di bawah ini ada 11 table yang akan secara otomatis menyimpan data-data yang ada pada bank sampah.



Jadwal

Pada table jadwal ini kita membuat id nya AUTO_INCREMENT agar saat id tersebut di inputkan akan secara otomatis nilainya akan bertambah.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	req_id	int(11)			No	None			Change Drop More
3	petugas_id	int(11)			No	None			Change Drop More

<div><div><div></div><div></div><div></div></div></div>					id	req_id	petugas_id		
<input type="checkbox"/>		Edit		Copy		Delete	10	4	42
<input type="checkbox"/>		Edit		Copy		Delete	12	5	43
<input type="checkbox"/>		Edit		Copy		Delete	13	6	42
<input type="checkbox"/>		Edit		Copy		Delete	14	5	42
<input type="checkbox"/>		Edit		Copy		Delete	16	8	42

Pada gambar di atas bisa kita lihat hasilnya, disitu kita hanya perlu menginputkan data pada req_id dan petugas_id maka pada id nilainya akan bertambah secara otomatis

Req

Pada table req juga sama kita menambahkan AUTO_INCREMENT pada idnya

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
<input type="checkbox"/> 2	user_id	varchar(20)	utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/> 3	alamat	varchar(128)	utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/> 4	tgl_jemput	date			No	None			Change Drop More
<input type="checkbox"/> 5	status	varchar(30)	utf8mb4_general_ci		No	None			Change Drop More

<div><div>←</div><div>T</div><div>→</div></div>						id	user_id	alamat	tgl_jemput	status
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	4	34	jl. sarimanah 45	2019-12-28	diambil		
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	5	39	Jl. cijerokaso 50	2019-12-31	diambil		
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	6	39	Jl. cijerokaso 50	2019-12-26	diambil		
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	7	34	jl. sarimanah 45	2019-12-31	Belum diambil		
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	8	34	jl. sarimanah 45	2020-01-17	Belum diambil		

Sampah

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id	int(11)		No	None		AUTO_INCREMENT	Change Drop Primary ▼ More
<input type="checkbox"/>	2	nama	varchar(128) utf8mb4_general_ci		No	None			Change Drop Primary ▼ More
<input type="checkbox"/>	3	harga	int(11)		No	None			Change Drop Primary ▼ More

					id	nama	harga
<input type="checkbox"/>	Edit	Copy	Delete		1	Besi	23000
<input type="checkbox"/>	Edit	Copy	Delete		2	Botol Plastik	6000
<input type="checkbox"/>	Edit	Copy	Delete		4	Kertas	5500
<input type="checkbox"/>	Edit	Copy	Delete		5	Gelas Plastik	5000
<input type="checkbox"/>	Edit	Copy	Delete		6	Ban Luar	15000
<input type="checkbox"/>	Edit	Copy	Delete		7	Kardus	7500
<input type="checkbox"/>	Edit	Copy	Delete		8	Karung	4500
<input type="checkbox"/>	Edit	Copy	Delete		9	Botol Kaca	8000
<input type="checkbox"/>	Edit	Copy	Delete		10	Alumunium	6000
<input type="checkbox"/>	Edit	Copy	Delete		11	Kaleng	5500

Tabungan

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id	int(11)		No	None		AUTO_INCREMENT	Change Drop Primary ▼ More
<input type="checkbox"/>	2	user_id	int(11)		No	None			Change Drop Primary ▼ More
<input type="checkbox"/>	3	setoran	text latin1_general_ci		No	None			Change Drop Primary ▼ More
<input type="checkbox"/>	4	penarikan	text latin1_general_ci		No	None			Change Drop Primary ▼ More
<input type="checkbox"/>	5	tanggal	int(11)		No	None			Change Drop Primary ▼ More

					id	user_id	setoran	penarikan	tanggal	
<input type="checkbox"/>		Edit		Copy		Delete	1	39	23000	1577496326
<input type="checkbox"/>		Edit		Copy		Delete	2	39	12000 500	1577496334
<input type="checkbox"/>		Edit		Copy		Delete	3	34	25000	1577496348
<input type="checkbox"/>		Edit		Copy		Delete	4	39	5000	1577496348
<input type="checkbox"/>		Edit		Copy		Delete	5	34	46000	1577978814
<input type="checkbox"/>		Edit		Copy		Delete	10	39	2000	1579018725
<input type="checkbox"/>		Edit		Copy		Delete	11	34	6000	1579459274
<input type="checkbox"/>		Edit		Copy		Delete	12	34	5000	1579463163

Tarik

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id	int(11)		No	None		AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2	user_id	int(11)		No	None			Change Drop More
<input type="checkbox"/>	3	penarikan	text		No	None			Change Drop More
<input type="checkbox"/>	4	tanggal	int(11)		No	None			Change Drop More
<input type="checkbox"/>	5	status	varchar(128)	utf8mb4_general_ci	No	None			Change Drop More

<div><div><div>←</div><div>T</div><div>→</div></div></div>						id	user_id	penarikan	tanggal	status
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	1	39	2000	1579013097	disetujui		
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	2	39	2000	1579013235	belum disetujui		
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	5	34	6000	1579459022	disetujui		
<div><div><div></div></div></div>	<div><div><div></div></div></div> Edit	<div><div><div></div></div></div> Copy	<div><div><div></div></div></div> Delete	6	34	5000	1579463143	disetujui		

User

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id	int(20)		No	None		AUTO_INCREMENT	Change Drop Primary Unique More
<input type="checkbox"/>	2	name	varchar(128)	utf8mb4_general_ci	No	None			Change Drop Primary Unique More
<input type="checkbox"/>	3	email	varchar(128)	utf8mb4_general_ci	No	None			Change Drop Primary Unique More
<input type="checkbox"/>	4	image	varchar(128)	utf8mb4_general_ci	No	None			Change Drop Primary Unique More
<input type="checkbox"/>	5	password	varchar(256)	utf8mb4_general_ci	No	None			Change Drop Primary Unique More
<input type="checkbox"/>	6	role_id	int(11)		No	None			Change Drop Primary Unique More
<input type="checkbox"/>	7	is_active	int(1)		No	None			Change Drop Primary Unique More
<input type="checkbox"/>	8	date_created	int(11)		No	None			Change Drop Primary Unique More

	id	name	email	image	password	role_id	is_active	date_created
	10	Bakti Qilan Mufid	bakti.qilan17@gmail.com	AMXU0T8.jpg	\$2y\$10\$snymZEIGQFixsxZtXepZJemFjp3Q3SKyDBmYGzxbW...	1	1	1573351...
	34	Nasabah pertama	nasabah1@banksampah.com	default.jpg	\$2y\$10\$5JXHMPmO4P6DSWHxytdehve4YetiULMTzl2uVmp7S...	2	1	1576637...
	39	Nasabah kedua	nasabah2@banksampah.com	default.jpg	\$2y\$10\$GsFX67lX7Dlax6GddamX/esRjYGTvIQonvTtB2cxcEg...	2	1	1576750...
	42	Petugas tiga	petugas3@banksampah.com	default.jpg	\$2y\$10\$5ZYskSNsWQLyFJQOkx5bleqPfwDSLuz7zIVwsXbjnsL...	3	1	1576810...
	43	Petugas satu	petugas1@banksampah.com	default.jpg	\$2y\$10\$JkIDMjtkqEQgMAIXDS4dO2lbelJ7Tzr0YNBKysraRF...	3	1	1576814...
	44	Nasabah ketiga	nasabah3@banksampah.com	default.jpg	\$2y\$10\$T2OM1Su/K5Ox BONVPJhuNM0OIXTQm1UR0DWdw9yGi...	2	1	1578010...
	45	Petugas dua	petugas2@banksampah.com	default.jpg	\$2y\$10\$H.8mDmLqO8KzpeZ7UnUigeMcnYHENB6kPDqu4pcMLTR...	3	1	1578013...
	49	Nasabah keempat	nasabah4@banksampah.com	default.jpg	\$2y\$10\$2r3U0EQNLnGDCbFJv7T.VugfMkt9kFqa5umm45BNM4i...	2	1	1579006...

User_access_menu

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop Primary More
<input type="checkbox"/> 2	role_id	int(11)			No	None			Change Drop Primary More
<input type="checkbox"/> 3	menu_id	int(11)			No	None			Change Drop Primary More

				id	role_id	menu_id
<input type="checkbox"/>	Edit	Copy	Delete	1	1	1
<input type="checkbox"/>	Edit	Copy	Delete	2	1	2
<input type="checkbox"/>	Edit	Copy	Delete	3	2	2
<input type="checkbox"/>	Edit	Copy	Delete	4	1	3
<input type="checkbox"/>	Edit	Copy	Delete	7	2	13
<input type="checkbox"/>	Edit	Copy	Delete	9	3	14
<input type="checkbox"/>	Edit	Copy	Delete	10	3	2
<input type="checkbox"/>	Edit	Copy	Delete	12	2	15
<input type="checkbox"/>	Edit	Copy	Delete	14	3	15
<input type="checkbox"/>	Edit	Copy	Delete	19	1	15
<input type="checkbox"/>	Edit	Copy	Delete	20	1	32
<input type="checkbox"/>	Edit	Copy	Delete	21	2	33
<input type="checkbox"/>	Edit	Copy	Delete	23	2	34

User_detail

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop
<input type="checkbox"/> 2	no_rek	varchar(20)	utf8mb4_general_ci		Yes	NULL			Change Drop
<input type="checkbox"/> 3	nama	varchar(128)	utf8mb4_general_ci		No	None			Change Drop
<input type="checkbox"/> 4	alamat	varchar(128)	utf8mb4_general_ci		Yes	NULL			Change Drop
<input type="checkbox"/> 5	scan_ktp	varchar(128)	utf8mb4_general_ci		Yes	NULL			Change Drop
<input type="checkbox"/> 6	scan_kk	varchar(128)	utf8mb4_general_ci		Yes	NULL			Change Drop
<input type="checkbox"/> 7	user_id	int(11)			No	None			Change Drop
<input type="checkbox"/> 8	role_id	int(11)			No	None			Change Drop

Formulir Data													
				id	no_rek	nama	alamat	scan_ktp	scan_kk	user_id	role_id		
<input type="checkbox"/>		Edit		Copy		Delete	15	BS0001	Nasabah pertama	jl. sarimanah 45	KTP.jpg	kartu-keluarga.jpg	34
<input type="checkbox"/>		Edit		Copy		Delete	20	BS0002	Nasabah kedua	Jl. cijerokaso 50	download.jpg	kk1.jpg	39
<input type="checkbox"/>		Edit		Copy		Delete	22	0	Petugas tiga	jl. sariasih	default.png	default.png	42
<input type="checkbox"/>		Edit		Copy		Delete	23	0	Petugas satu	-	default.png	default.png	43
<input type="checkbox"/>		Edit		Copy		Delete	24	BS0003	Nasabah ketiga	Jl. Pasirhuni	ktp2.jpg	kk.png	44
<input type="checkbox"/>		Edit		Copy		Delete	25	0	Petugas dua	Jl. cibogo	default.png	default.png	45
<input type="checkbox"/>		Edit		Copy		Delete	29	0	Nasabah keempat	Jl. cijerokaso	room2.png	120006.jpg	49

User_menu

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop
<input type="checkbox"/> 2	menu	varchar(128)	utf8mb4_general_ci		No	None			Change Drop

					id	menu		
<input type="checkbox"/>		Edit		Copy		Delete	1	Admin
<input type="checkbox"/>		Edit		Copy		Delete	2	User
<input type="checkbox"/>		Edit		Copy		Delete	3	Menu
<input type="checkbox"/>		Edit		Copy		Delete	13	Tabungan
<input type="checkbox"/>		Edit		Copy		Delete	14	Petugas
<input type="checkbox"/>		Edit		Copy		Delete	15	Info
<input type="checkbox"/>		Edit		Copy		Delete	33	Request
<input type="checkbox"/>		Edit		Copy		Delete	34	Data Detail User

User_role

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id	int(11)		No	None		AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2	role	varchar(128)	utf8mb4_general_ci	No	None			Change Drop More

					id	role
<input type="checkbox"/>	Edit	Copy	Delete		1	Administrator
<input type="checkbox"/>	Edit	Copy	Delete		2	Nasabah
<input type="checkbox"/>	Edit	Copy	Delete		3	Petugas
<input type="checkbox"/>	Edit	Copy	Delete		6	caa

User_sub_menu

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id	int(11)		No	None		AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2	menu_id	int(11)		No	None			Change Drop More
<input type="checkbox"/>	3	title	varchar(128)	utf8mb4_general_ci	No	None			Change Drop More
<input type="checkbox"/>	4	url	varchar(128)	utf8mb4_general_ci	No	None			Change Drop More
<input type="checkbox"/>	5	icon	varchar(128)	utf8mb4_general_ci	No	None			Change Drop More

					id	menu_id	title	url	icon
<input type="checkbox"/>	Edit	Copy	Delete		2	2	My Profile	user	fas fa-fw fa-user
<input type="checkbox"/>	Edit	Copy	Delete		3	2	Edit Profile	user/edit	fas fa-fw fa-user-edit
<input type="checkbox"/>	Edit	Copy	Delete		4	3	Menu Management	menu/index	fas fa-fw fa-folder
<input type="checkbox"/>	Edit	Copy	Delete		5	3	Submenu Management	menu/submenu	fas fa-fw fa-folder-open
<input type="checkbox"/>	Edit	Copy	Delete		7	1	Role	admin/role	fas fa-fw fa-user-tie
<input type="checkbox"/>	Edit	Copy	Delete		11	11	Coba	coba/coba	fab fa-fw fa-youtube
<input type="checkbox"/>	Edit	Copy	Delete		13	2	Ubah Password	user/ubahpassword	fas fa-fw fa-key
<input type="checkbox"/>	Edit	Copy	Delete		14	1	Aktivasi Nasabah	admin/aktivasi	fas fa-fw fa-user-check
<input type="checkbox"/>	Edit	Copy	Delete		15	13	Cek Saldo	tabungan	fas fa-fw fa-money-check
<input type="checkbox"/>	Edit	Copy	Delete		16	15	Harga Sampah	info	fas fa-fw fa-dollar-sign
<input type="checkbox"/>	Edit	Copy	Delete		17	14	Pengambilan Sampah	petugas	fas fa-fw fa-handshake
<input type="checkbox"/>	Edit	Copy	Delete		20	0			
<input type="checkbox"/>	Edit	Copy	Delete		24	1	Input Sampah	admin/sampah	fas fa-fw fa-keyboard
<input type="checkbox"/>	Edit	Copy	Delete		25	34	Data Detail	data	fas fa-fw fa-user-tie
<input type="checkbox"/>	Edit	Copy	Delete		26	33	Meminta Jemput Sampah	request	fas fa-fw fa-pen-alt
<input type="checkbox"/>	Edit	Copy	Delete		27	1	Jemput Sampah	admin/jemput	fas fa-fw fa-pen-alt
<input type="checkbox"/>	Edit	Copy	Delete		29	14	Histori Pengambilan	petugas/histori	fas fa-fw fa-history
<input type="checkbox"/>	Edit	Copy	Delete		30	1	Histori Penjemputan	admin/histori	fas fa-fw fa-history
<input type="checkbox"/>	Edit	Copy	Delete		31	13	Request Penarikan Saldo	tabungan/tarik	fas fa-fw fa-money-bill-wav
<input type="checkbox"/>	Edit	Copy	Delete		32	1	Penarikan Saldo Nasabah	admin/tarik	fas fa-fw fa-dollar-sign

Console

Source codenya pada database.php :

```
/*
| -----
| -----
| DATABASE CONNECTIVITY SETTINGS
| -----
| -----
| This file will contain the settings needed to access
| your database.
|
| For complete instructions please consult the 'Database Connection'
| page of the User Guide.
|
| -----
| -----
| EXPLANATION OF VARIABLES
| -----
| -----
|
|   ['dsn']           The full DSN string describe a connection to the database.
|   ['hostname']      The hostname of your database server.
|   ['username']       The username used to connect to the database
|   ['password']       The password used to connect to the database
|   ['database']       The name of the database you want to connect to
|   ['dbdriver']       The database driver. e.g.: mysqli.
|                       Currently supported:
|                       cubrid, ibase, mssql, mysql, mysqli,
|                       oci8,
```

```
|          odbc, pdo, postgre, sqlite, sqlite3,
sqlsrv
|  ['dbprefix'] You can add an optional prefix, which
|               will be added
|               to the table name when using the Que
ry Builder class
|  ['pconnect'] TRUE/FALSE - Whether to use a persist
ent connection
|  ['db_debug'] TRUE/FALSE - Whether database errors
should be displayed.
|  ['cache_on'] TRUE/FALSE - Enables/disables query c
aching
|  ['cachedir'] The path to the folder where cache fi
les should be stored
|  ['char_set'] The character set used in communicati
ng with the database
|  ['dbcollat'] The character collation used in commu
nicating with the database
|               NOTE: For MySQL and MySQLi databases,
|               this setting is only used
|               as a backup if your server is running
|               PHP < 5.2.3 or MySQL < 5.0.7
|               (and in table creation queries made w
ith DB Forge).
|               There is an incompatibility in PHP wi
th mysql_real_escape_string() which
|               can make your site vulnerable to SQL
injection if you are using a
|               multi-
byte character set and are running versions lower than
these.
|               Sites using Latin-1 or UTF-
8 database character set and collation are unaffected.
|  ['swap_pre'] A default table prefix that should be
swapped with the dbprefix
|  ['encrypt']  Whether or not to use an encrypted co
nnection.
|
```

```

|         'mysql' (deprecated), 'sqlsrv' and 'pdo/sq
lsrv' drivers accept TRUE/FALSE
|         'mysqli' and 'pdo/mysqli' drivers accept an
array with the following options:
|
|         'ssl_key'      - Path to the private key
file
|         'ssl_cert'     - Path to the public key
certificate file
|         'ssl_ca'       - Path to the certificate
authority file
|         'ssl_cacpath'  - Path to a directory con
taining trusted CA certificates in PEM format
|         'ssl_cipher'   - List of *allowed* ciphe
rs to be used for the encryption, separated by colons
(':')
|         'ssl_verify'   - TRUE/FALSE; Whether ver
ify the server certificate or not
|
|     ['compress'] Whether or not to use client compress
ion (MySQL only)
|     ['stricton'] TRUE/FALSE - forces 'Strict Mode' con
nections
|
|                                     - good for ensuring strict
SQL while developing
|     ['ssl_options'] Used to set various SSL options th
at can be used when making SSL connections.
|     ['failover'] array - A array with 0 or more data f
or connections if the main should fail.
|     ['save_queries'] TRUE/FALSE - Whether to "save" al
l executed queries.
|
|         NOTE: Disabling this will also effecti
vely disable both
|
|         $this->db-
>last_query() and profiling of DB queries.
|
|         When you run a query, with this settin
g set to TRUE (default),
|
|         CodeIgniter will store the SQL stateme
nt for debugging purposes.

```

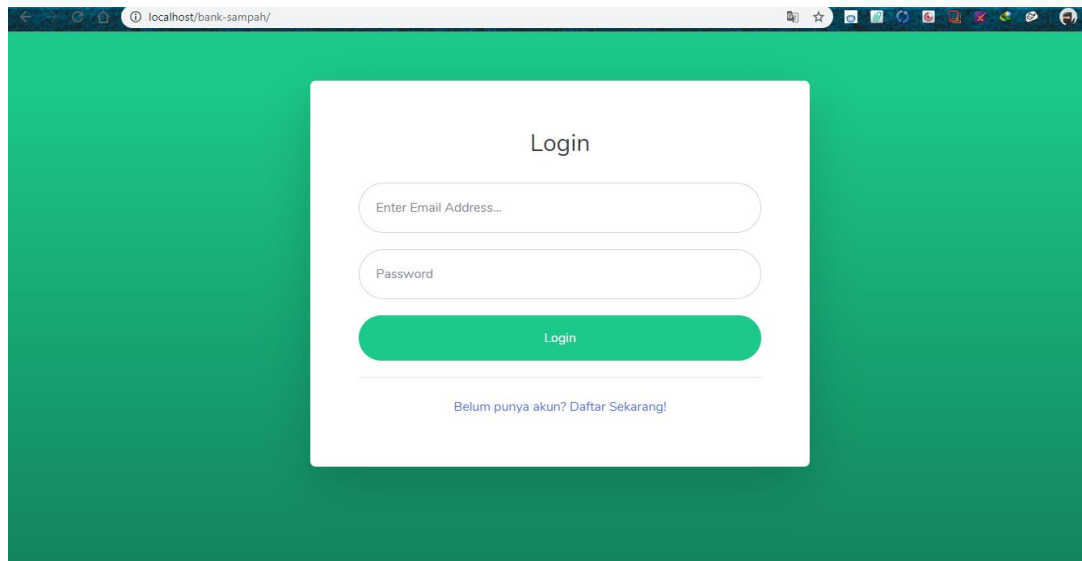
```

|           However, this may cause high memory us
age, especially if you run
|           a lot of SQL queries ... disable this
to avoid that problem.
|
| The $active_group variable lets you choose which con
nection group to
| make active. By default there is only one group (th
e 'default' group).
|
| The $query_builder variables lets you determine whet
her or not to load
| the query builder class.
*/
$active_group = 'default';
$query_builder = TRUE;

$db['default'] = array(
    'dsn'      => '',
    'hostname' => 'localhost',
    'username' => 'root',
    'password' => '',
    'database' => 'bank_sampah',
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => (ENVIRONMENT !== 'production'),
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
    'encrypt'  => FALSE,
    'compress' => FALSE,
    'stricton' => FALSE,
    'failover' => array(),
    'save_queries' => TRUE
);

```

Pertama kita akan membuka <http://localhost/bank-sampah/>, dan akan muncul tampilan login sebagai berikut :



Gambar di atas merupakan tampilan awal login untuk Admin, nasabah dan petugas

Ini adalah source code yang memanggil <http://localhost/bank-sampah/>

```
$config['base_url'] = 'http://localhost/bank-sampah/';
```

Source code untuk halaman login

```
class Auth extends CI_Controller
{
    public function __construct()
    {
```

```

        parent::__construct();
        $this->load->library('form_validation');
    }

    public function index()
    {
        if ($this->session->userdata('email')) {
            redirect('user');
        }

        $this->form_validation->
>set_rules('email', 'Email', 'required|trim|valid_email', [
            'required' => 'email harus diisi!'
        ]);
        $this->form_validation->
>set_rules('password', 'Password', 'required|trim', [
            'required' => 'password harus diisi!'
        ]);

        if ($this->form_validation->run() == false) {
            $data['title'] = 'Login Page';
            $this->load->
>view('templates/auth_header', $data);
            $this->load->view('auth/login');
            $this->load->
>view('templates/auth_footer');
        } else {

            $this->_login();
        }
    }

    private function _login()
    {
        $email = $this->input->post('email');
        $password = $this->input->post('password');
    }

```

```

        $user = $this->db-
>get_where('user', ['email' => $email])->row_array();

        if ($user) {

            if ($user['is_active'] == 1) {
                if (password_verify($password, $user['
password'])) {

                    $data = [
                        'id' => $user['id'],
                        'email' => $user['email'],
                        'role_id' => $user['role_id']
                    ];
                    $this->session-
>set_userdata($data);
                    if ($user['role_id'] == 1) {
                        redirect('admin/aktivasi');
                    }
                    if ($user['role_id'] == 3) {
                        redirect('petugas');
                    } else {
                        redirect('tabungan');
                    }
                } else {
                    $this->session-
>set_flashdata('message', '<div class="alert alert-
danger mx-auto" role="alert">Password salah!</div>');
                    redirect('auth');
                }
            } else {
                $this->session-
>set_flashdata('message', '<div class="alert alert-
danger mx-
auto" role="alert">Email tidak aktif</div>');
                redirect('auth');
            }
        } else {
            $this->session-
>set_flashdata('message', '<div class="alert alert-

```



```

danger mx-
auto" role="alert">Email tidak terdaftar!</div>');
        redirect('auth');
    }
}

```

Ketika Masyarakat yang ingin mendaftar sebagai nasabah atau sebagai petugas kebersihan maka masyarakat harus melakukan registrasi terlebih dahulu untuk membuat akun pada aplikasi bank sampah. Maka kita buka <http://localhost/bank-sampah/auth/registration> maka akan tampil halaman seperti gambar dibawah ini

Source code untuk halaman register

```

public function registration()
{
    $this->load->model('Auth_model', 'auth');
}

```

```

        if ($this->session->userdata('email')) {
            redirect('user');
        }

        $this->form_validation->
>set_rules('name', 'Name', 'required|trim', [
            'required' => 'nama harus diisi!',
        ]);
        $this->form_validation->
>set_rules('alamat', 'Alamat', 'required', [
            'required' => 'alamat harus diisi!',
        ]);
        $this->form_validation->
>set_rules('email', 'Email', 'required|trim|valid_email|is_unique[user.email]', [
            'required' => 'email harus diisi!',
            'valid_email' => 'format email salah!',
            'is_unique' => 'email ini sudah terdaftar!'
        ],
        );

        $this->form_validation->
>set_rules('password1', 'Password', 'required|trim|min_length[3]|matches[password2]', [
            'required' => 'password harus diisi!',
            'matches' => 'password tidak sama!',
            'min_length' => 'password terlalu pendek!'
        ]);
        $this->form_validation->
>set_rules('password2', 'Password', 'required|trim|matches[password1]', [
            'required' => 'password harus diisi!'
        ]);

        if ($this->form_validation->run() == false) {
            $data['title'] = 'Pendaftaran Bank Sampah'
        };

        $this->load-
>view('templates/auth_header', $data);

```

```

        $this->load->view('auth/registration');
        $this->load-
>view('templates/auth_footer');
    } else {
        $data = [
            'name' => htmlspecialchars($this-
>input->post('name', true)),
            'email' => htmlspecialchars($this-
>input->post('email', true)),
            'image' => 'default.jpg',
            'password' => password_hash($this-
>input->post('password1'), PASSWORD_DEFAULT),
            'role_id' => $this->input-
>post('radiob', true),
            'is_active' => 1,
            'date_created' => time(),
        ];
        $data1 = [
            'no_rek' => 0,
            'nama' => htmlspecialchars($this-
>input->post('name', true)),
            'alamat' => htmlspecialchars($this-
>input->post('alamat', true)),
            'scan_ktp' => 'default.png',
            'scan_kk' => 'default.png',
            'user_id' => $this->auth-
>create('user', $data),
            'role_id' => $this->input-
>post('radiob', true),
        ];
        $insert1 = $this->auth-
>create('user_detail', $data1);
        $this->session-
>set_flashdata('message', '<div class="alert alert-
success mx-
auto" role="alert">Selamat! Akun anda telah berhasil d
ibuat. Silahkan Login!</div>');
        redirect('auth');
    }

```

```
}
```

Daftar akun sebagai:

Nasabah Petugas

nasa1

bandung

nasa1@gmail.com

...

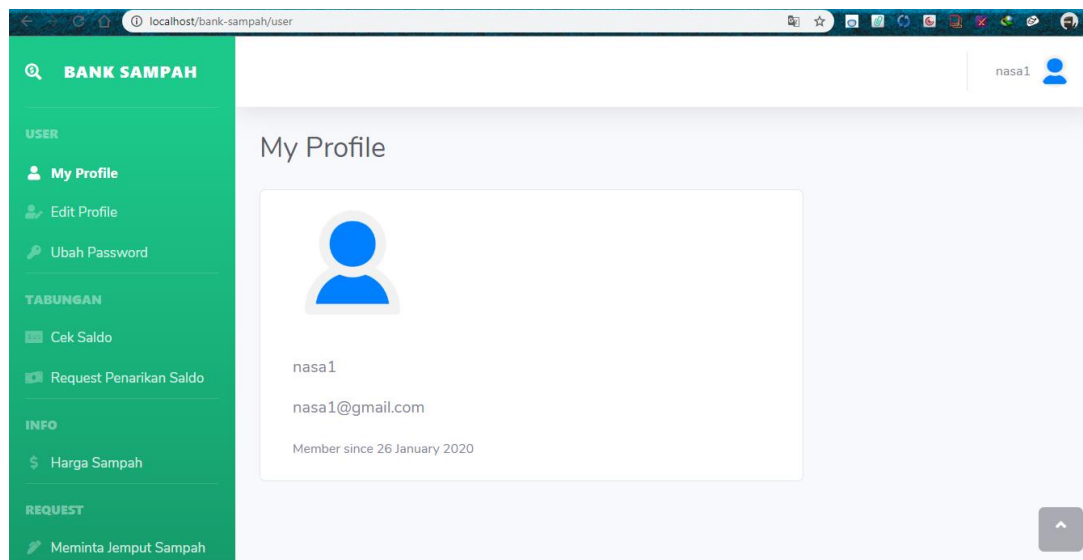
...

Daftar

[Sudah punya akun ? Silahkan Login!](#)

Pada gambar diatas kita melakukan registrasi sebagai nasabah, lalu klik daftar maka akan muncul pemberitahuan seperti ini:

Selamat! Akun anda telah berhasil dibuat.
Silahkan Login!



Setelah berhasil membuat akun maka kita akan melanjutkan proses login sebagai nasabah Setelah itu kita masuk ke <http://localhost/bank-sampah/user> karna berhasil melakukan login sebagai nasabah

Source code nya :

```
class User extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        cek_role();
    }
    public function index()
    {
        $data['title'] = 'My Profile';
        $data['user'] = $this->db-
>get_where('user', ['email' => $this->session-
>userdata('email')])->row_array();

        $this->load->view('templates/header', $data);
    }
}
```

```

        $this->load->view('templates/sidebar', $data);
        $this->load->view('templates/topbar', $data);
        $this->load->view('user/index', $data);
        $this->load->view('templates/footer');
    }

    public function edit()
    {
        $data['title'] = 'Edit Profile';
        $data['user'] = $this->db-
>get_where('user', ['email' => $this->session-
>userdata('email')])->row_array();

        $this->form_validation-
>set_rules('name', 'Nama Lengkap', 'required|trim', [
            'required' => 'nama harus diisi'
        ]);

        if ($this->form_validation->run() == false) {
            $this->load-
>view('templates/header', $data);
            $this->load-
>view('templates/sidebar', $data);
            $this->load-
>view('templates/topbar', $data);
            $this->load->view('user/edit', $data);
            $this->load->view('templates/footer');
        } else {
            $name = $this->input->post('name');
            $email = $this->input->post('email');
            $nama = $this->input->post('name');
            $id = $this->input->post('id');

            //untuk cek foto
            $upload_image = $_FILES['foto']['name'];

            if ($upload_image) {
                $config['allowed_types'] = 'jpg|png|gi
f';

```

```

        $config['max_size']      = '3000';
        $config['upload_path']   = './asset
s/img/profile/';

        $this->load-
>library('upload', $config);

        if ($this->upload-
>do_upload('foto')) {
            $old_image = $data['user']['image
'];
            if ($old_image != 'default.jpg') {
                unlink(FCPATH . ('assets/img/p
rofile/') . $old_image);
            }

            $new_image = $this->upload-
>data('file_name');
            $this->db-
>set('image', $new_image);
            } else {
                echo $this->upload-
>display_errors();
            }
        }

        $this->db->set('name', $name);
        $this->db->where('email', $email);
        $this->db->update('user');
        $this->db->set('nama', $nama);
        $this->db->where('user_id', $id);
        $this->db->update('user_detail');

        $this->session-
>set_flashdata('message', '<div class="alert alert-
success mx-
auto" role="alert">Profile anda telah diperbaharui!</d
iv>');

        redirect('user');

```

```

    }
}

public function ubahPassword()
{
    $data['title'] = 'Ubah Password';
    $data['user'] = $this->db-
>get_where('user', ['email' => $this->session-
>userdata('email')])->row_array();

    $this->form_validation-
>set_rules('password_sekarang', 'Password Sekarang', '
required|trim', [
        'required' => 'password sekarang harus dii
si'
    ]);
    $this->form_validation-
>set_rules('password_baru1', 'Password Baru', 'require
d|trim|min_length[3]|matches[password_baru2]', [
        'required' => 'password baru harus diisi!'
    ],
        'min_length' => 'password terlalu pendek!'
    ]);
    $this->form_validation-
>set_rules('password_baru2', 'Konfirmasi Password Baru
', 'required|trim|min_length[3]|matches[password_baru1
]', [
        'required' => 'konfirmasi password harus d
isi!',
        'matches' => 'password tidak sama!'
    ]);

    if ($this->form_validation->run() == false) {
        $this->load-
>view('templates/header', $data);
        $this->load-
>view('templates/sidebar', $data);
        $this->load-
>view('templates/topbar', $data);
    }
}

```



```

        $this->load-
>view('user/ubahpassword', $data);
        $this->load->view('templates/footer');
    } else {
        $password_sekarang = $this->input-
>post('password_sekarang');
        $password_baru = $this->input-
>post('password_baru1');

        if (!password_verify($password_sekarang, $
data['user']['password'])) {
            $this->session-
>set_flashdata('message', '<div class="alert alert-
danger mx-
auto" role="alert">Masukan Password sekarang dengan be
nar!</div>');
            redirect('user/ubahpassword');
        } else {
            if ($password_sekarang == $password_ba
ru) {
                $this->session-
>set_flashdata('message', '<div class="alert alert-
danger mx-
auto" role="alert">Password tidak boleh sama!</div>');
                redirect('user/ubahpassword');
            } else {
                $password_hash = password_hash($pa
ssword_baru, PASSWORD_DEFAULT);

                $this->db-
>set('password', $password_hash);
                $this->db->where('email', $this-
>session->userdata('email'));
                $this->db->update('user');

                $this->session-
>set_flashdata('message', '<div class="alert alert-
success mx-
auto" role="alert">Password telah diubah!</div>');

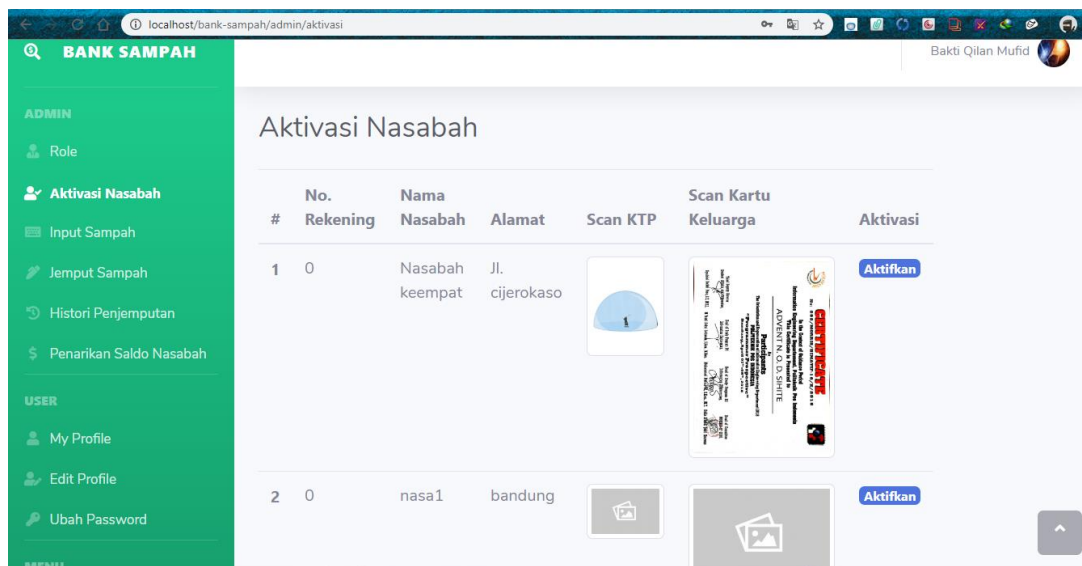
```

```

        redirect('user/ubahpassword');
    }
}
}
}
}

```

Sebelum nasabah bisa melakukan proses menabung sampah pada aplikasi bank sampah, nasabah harus melengkapi terlebih dahulu scan KTP dan scan Kartu Keluarga setelah persyaratan itu sudah lengkap maka admin akan melakukan aktivasi pada nasabah yang sudah memenuhi semua persyaratan, setelah itu nasabah akan mendapatkan no.rekening dan dapat melakukan proses menabung. Setelah persyaratan sudah lengkap maka Admin akan melakukan aktivasi pada <http://localhost/bank-sampah/admin/aktivasi> seperti gambar di bawah ini



Source Codenya pada controlles >> admin.php :

```
class Admin extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        cek_role();
        $this->load-
>model('Aktivasi_model', 'aktivasi', true);
        $this->load-
>model('Admin_model', 'admin', true);
        $this->load->helper(array('form', 'url'));
    }
}
```

```
public function aktivasi()
{
    $data['title'] = 'Aktivasi Nasabah';
    $data['user'] = $this->db-
>get_where('user', ['email' => $this->session-
>userdata('email')])->row_array();

    $data['hasil'] = $this->db-
>get_where('user_detail', ['role_id' => '2'])-
>result_array();
    $data['aktivasi'] = $this->admin-
>getAktivasi();

    $this->load->view('templates/header', $data);
    $this->load->view('templates/sidebar', $data);
    $this->load->view('templates/topbar', $data);
    $this->load->view('admin/aktivasi', $data);
    $this->load->view('templates/footer');
}
```

```

        public function editrek($id)
        {
            $this->load-
>model('Aktivasi_model', 'aktivasi');
            $this->aktivasi->editRek($id);
            $this->session-
>set_flashdata('message', '<div class="alert alert-
success mx-
auto" role="alert">Akun Diaktifkan!</div>');
            redirect('admin/aktivasi');
        }

```

Source code pada models >> aktivasi_model.php :

```

class Aktivasi_model extends CI_Model
{
    public function getUser()
    {
        # $this->db->order_by('user_detail.id', 'ASC');
        # return $this->db->from('user_detail')
        # -
>join('user', 'user.id=user_detail.user_id')
        # ->get()
        # ->result();
        $this->db->select('*');
        $this->db->from('user_detail');
        $this->db-
>join('user', 'user.id=user_detail.user_id');

        $query = $this->db->get();
        return $query->result();
    }
    function get_user($user_id)
    {
        $this->db->select('*');
        $this->db->from('user');
    }
}

```

```

        $this->db->join('user_detail', 'user_detail.user_id=user.id');
        $this->db->where('user.id', $user_id);
        $query = $this->db->get();
        return $query;
    }

    public function editRek($id)
    {
        $data = [
            'no_rek' => getAutoNumber('user_detail', 'no_rek', 'BS', 6),
        ];
        $this->db->where('id', $id);
        $this->db->update('user_detail', $data);
    }
}

```

Setelah aktivasi berhasil nasabah mendapatkan no rekening dan bisa melakukan proses menabung, maka kita akan masuk ke <http://localhost/bank-sampah/request> pada halaman ini nasabah

akan mengatur tanggal penjemputan sampah seperti gambar di bawah ini :

BANK SAMPAH

USER

- My Profile
- Edit Profile
- Ubah Password

TABUNGAN

- Cek Saldo
- Request Penarikan Saldo

INFO

- Harga Sampah

REQUEST

- Meminta Jemput Sampah

nasa1

Meminta Penjemputan Sampah

Permintaan Penjemputan telah ditambahkan!

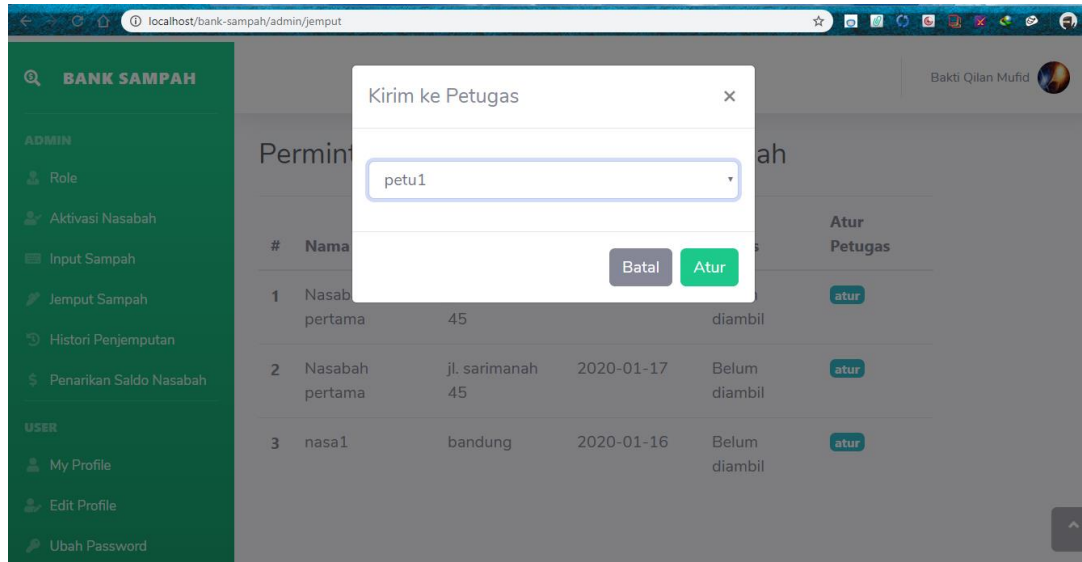
mm/dd/yyyy

Jemput

#	Tanggal Penjemputan	Status
1	16-01-2020	Belum diambil

Setelah nasabah mengatur jadwal pengambilan sampah maka akan masuk ke halaman admin pada <http://localhost/bank-sampah/admin/jemput> dan admin akan mengatur petugas mana

yang akan melakukan pengambilan sampah seperti pada gambar di bawah ini :



Source codenya :

```
public function jemput()
{
    $data['title'] = 'Permintaan Jemput Sampah dari Nasabah';
    $data['user'] = $this->db->get_where('user', ['email' => $this->session->userdata('email')])->row_array();

    $data['jemput'] = $this->admin->getAll();
    $data['nama'] = $this->admin->getJadwal();
    $data['petugas'] = $this->db->get_where('user', ['role_id' => '3'])->result_array();

    $this->load->view('templates/header', $data);
    $this->load->view('templates/sidebar', $data);
}
```

```

$this->load->view('templates/topbar', $data);
$this->load->view('admin/jemput', $data);
$this->load->view('templates/footer');
}

```

Setelah di atur penjemputan sampah maka akan muncul pemberitahuan seperti ini:

Data Penjemputan berhasil di jadwalkan!

Lalu masuk ke <http://localhost/bank-sampah/petugas>, pada halaman ini akan muncul pemberitahuan tanggal penjemputan sampah seperti gambar dibawah ini :

#	Tanggal Penjemputan	No. Rekening	Nama Nasabah	Alamat	Status	Tindakan
1	2020-01-16	BS0004	nasa1	bandung	Belum diambil	<button>Ambil sekarang</button>

Source codenya :

```

class Petugas extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        cek_role();
    }
}

```



```

        $this->load-
>model('Petugas_model', 'petugas');
    }

    public function index()
    {
        $data['title'] = 'Jadwal Pengambilan';
        $data['user'] = $this->db-
>get_where('user', ['email' => $this->session-
>userdata('email')])->row_array();

        $data['ambil'] = $this->petugas->getAmbil();
        $data['jenis'] = $this->petugas->getJenis();
        $data['nasabah'] = $this->petugas-
>getNasabah();
        $data['petugas'] = $this->db->get('sampah')-
>result_array();

        $this->load->view('templates/header', $data);
        $this->load->view('templates/sidebar', $data);
        $this->load->view('templates/topbar', $data);
        $this->load->view('petugas/index', $data);
        $this->load->view('templates/footer');
    }

    public function input()
    {
        $data['title'] = 'Jadwal Pengambilan';
        $data['user'] = $this->db-
>get_where('user', ['email' => $this->session-
>userdata('email')])->row_array();

        $data['ambil'] = $this->petugas->getAmbil();
        $data['petugas'] = $this->db->get('sampah')-
>result_array();

        $setoranjmlh['v1'] = (int) $this->input-
>post('j_sampah', true);

```

```

        $setoranjmlh['v2'] = (int) $this->input-
>post('b_sampah', true);
        $setoran = $setoranjmlh['v1'] * $setoranjmlh['
v2'];
        $user_id = (int) $this->input-
>post('id', true);

        $this->form_validation-
>set_rules('j_sampah', 'Jenis Sampah', 'required', [
            'required' => 'Jenis Sampah harus diisi!'
        ]);
        $this->form_validation-
>set_rules('b_sampah', 'Berat Sampah', 'required', [
            'required' => 'Berat Sampah harus diisi!'
        ]);

        if ($this->form_validation->run() == false) {
            $this->session-
>set_flashdata('message', '<div class="alert alert-
danger mx-
auto" role="alert">Pengambilan Gagal!!</div>');
            $this->load-
>view('templates/header', $data);
            $this->load-
>view('templates/sidebar', $data);
            $this->load-
>view('templates/topbar', $data);
            $this->load->view('petugas/index', $data);
            $this->load->view('templates/footer');
        } else {
            $query = [
                'user_id' => $user_id,
                'setoran' => $setoran,
                'tanggal' => time()
            ];
            $query2 = [
                'status' => 'diambil'
            ];

```

```

        $this->db->insert('tabungan', $query);
        $this->db->where('id', $_POST['req_id']);
        $this->db->update('req', $query2);
        $this->session->
>set_flashdata('message', '<div class="alert alert-
success mx-
auto" role="alert">Pengambilan berhasil!!</div>');
        redirect('petugas/index');
    }
}

public function histori()
{
    $data['title'] = 'Histori Pengambilan';
    $data['user'] = $this->db->
>get_where('user', ['email' => $this->session->
>userdata('email')])>row_array();

    $data['histori'] = $this->petugas->
>getHistori();

    $this->load->view('templates/header', $data);
    $this->load->view('templates/sidebar', $data);
    $this->load->view('templates/topbar', $data);
    $this->load->view('petugas/histori', $data);
    $this->load->view('templates/footer');
}
}

```

Maka petugas akan melakukan pengambilan sampah sesuai jadwal yang telah ditentukan nya. Ketika petugas melakukan pengambilan

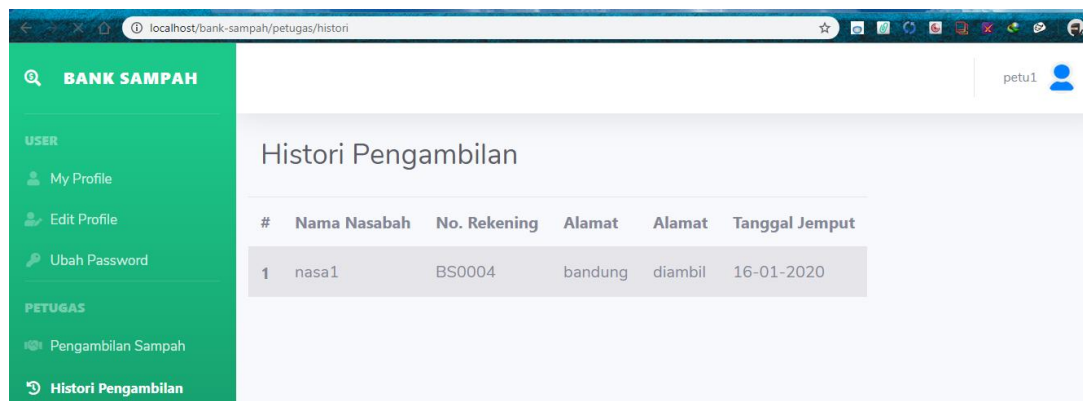
sampah maka akan menginputkan data-data sampah yang di ambil oleh petugas tersebut seperti pada gambar dibawah ini :

The screenshot shows a web application interface for 'BANK SAMPAH'. On the left is a dark green sidebar with navigation links: 'USER' (My Profile, Edit Profile, Ubah Password) and 'PETUGAS' (Pengambilan Sampah, Histori Pengambilan). Below these is an 'INFO' section (Harga Sampah, Logout). The main content area is titled 'Masukan data-data Nasabah' and contains a form with the following fields: a text input with '9', a text input with '50', a dropdown menu for 'Jenis Sampah' with 'Botol Kaca' selected, and a text input for 'Berat Sampah (kg)' with '20'. At the bottom of the form are two buttons: 'Batal' (grey) and 'Tambah' (green). The background shows a blurred view of a 'Jadwal' (Schedule) table and a 'Tindakan' (Action) button labeled 'Ambil sekarang'.

Setelah memasukan data-data lalu klik tambah maka akan muncul pemberitahuan seperti ini :

Pengambilan berhasil!!!

Lalu masuk ke <http://localhost/bank-sampah/petugas/histori> untuk melihat histori dari proses pengambilan sampah



Source codenya pada controllers >> petugas.php :

```
public function histori()
{
    $data['title'] = 'Histori Pengambilan';
    $data['user'] = $this->db-
>get_where('user', ['email' => $this->session-
>userdata('email')])->row_array();

    $data['histori'] = $this->petugas-
>getHistori();

    $this->load->view('templates/header', $data);
    $this->load->view('templates/sidebar', $data);
    $this->load->view('templates/topbar', $data);
    $this->load->view('petugas/histori', $data);
    $this->load->view('templates/footer');
}
}
```

Source code pada models >> Petugas_model.php :

```
class Petugas_model extends CI_Model
{
    public function getJenis()
```

```

    {
        $query = " SELECT *
                    FROM sampah";

        return $this->db->query($query)-
>result_array();
    }

    public function getAmbil()
    {
        $this->db-
>select('user_detail.*, req.tgl_jemput, req.id as req_i
d, req.status, jadwal.petugas_id, user.id');
        $this->db->from('user_detail');
        $this->db->where('req.status != "diambil"');
        $this->db-
>join('req', 'user_detail.user_id = req.user_id');
        $this->db-
>join('jadwal', 'req.id = jadwal.req_id');
        $this->db-
>join('user', 'user_detail.user_id = user.id');
        $query = $this->db-
>get_where('', ['petugas_id' => $this->session-
>userdata('id')]);
        return $query->result();
    }

    public function getNasabah()
    {
        $this->db-
>select('user_detail.*, tabungan.*');
        $this->db->from('user_detail');
        $this->db-
>join('tabungan', 'user_detail.id = tabungan.user_id')
;

        $query = $this->db->get();
        return $query->result();
    }
}

```

```

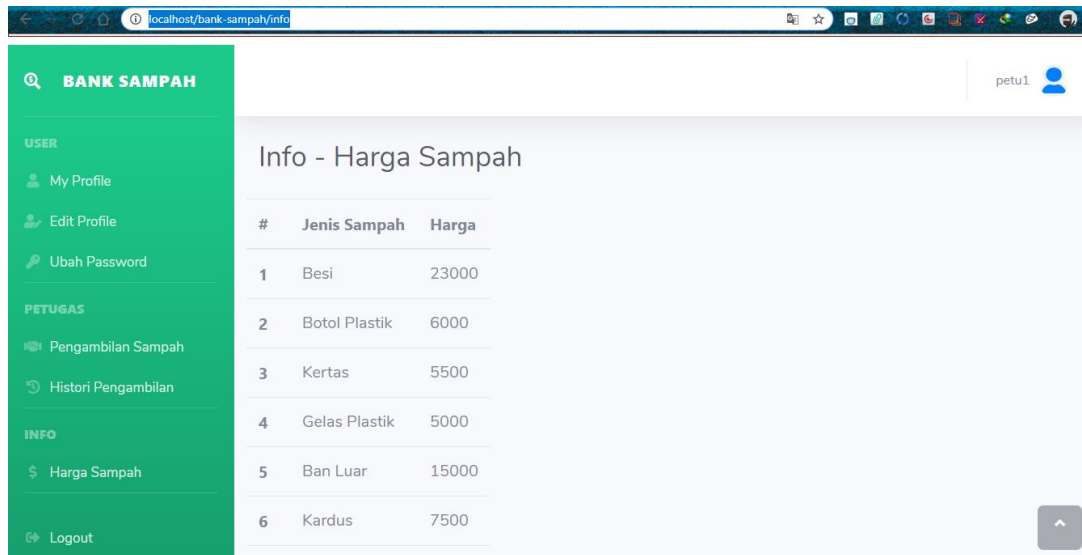
    public function getHistori()
    {
        $this->db->select('jadwal.*, req.*, user.name, user_detail.no_re
k');
        $this->db->from('jadwal', 'req');
        $this->db->where('req.status = "diambil" ');
        $this->db->join('req', 'jadwal.req_id = req.id');
        $this->db->join('user', 'req.user_id = user.id');
        $this->db->join('user_detail', 'req.user_id = user_detail.user_i
d');

        $query = $this->db->get_where('', ['petugas_id' => $this->session-
>userdata('id')]);
        return $query->result();
    }
}

```

Jika kita ingin mengetahui info harga sampah pada bank sampah, kita hanya perlu ke <http://localhost/bank-sampah/info> di halaman tersebut sudah terdapat seluruh info harga-harga sampah pada aplikasi bank sampah. Kita menambah info tentang harga sampah agar masyarakat/nasabah lebih semangat lagi dalam melakukan

proses menabung sampah dan lebih peka terhadap lingkungan sekitar terutama dalam hal menjaga kebersihan sampah.



Source codenya info.php :

```
class Info extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        cek_role();
    }

    public function index()
    {
        $data['title'] = 'Info - Harga Sampah';
        $data['user'] = $this->db->get_where('user', ['email' => $this->session->userdata('email')])->row_array();

        $data['info'] = $this->db->get('sampah')->result_array();
    }
}
```

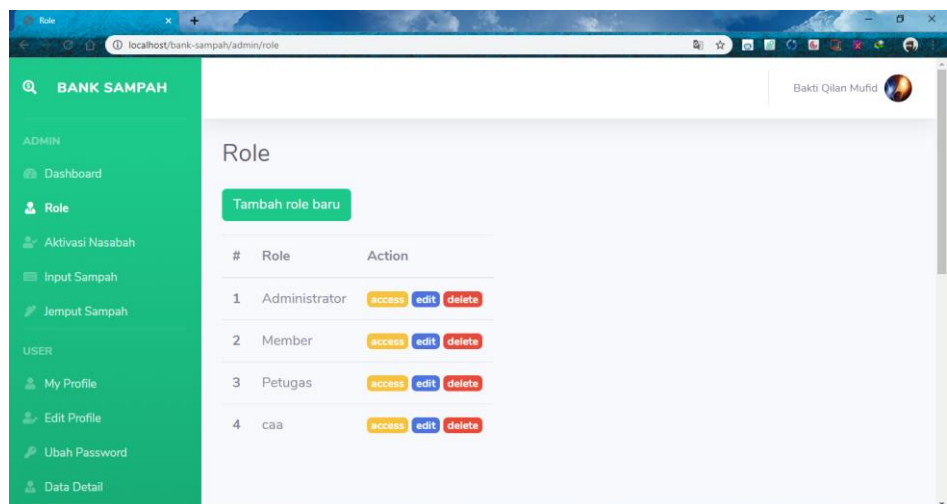


```

$this->load->view('templates/header', $data);
$this->load->view('templates/sidebar', $data);
$this->load->view('templates/topbar', $data);
$this->load->view('info/index', $data);
$this->load->view('templates/footer');
}
}

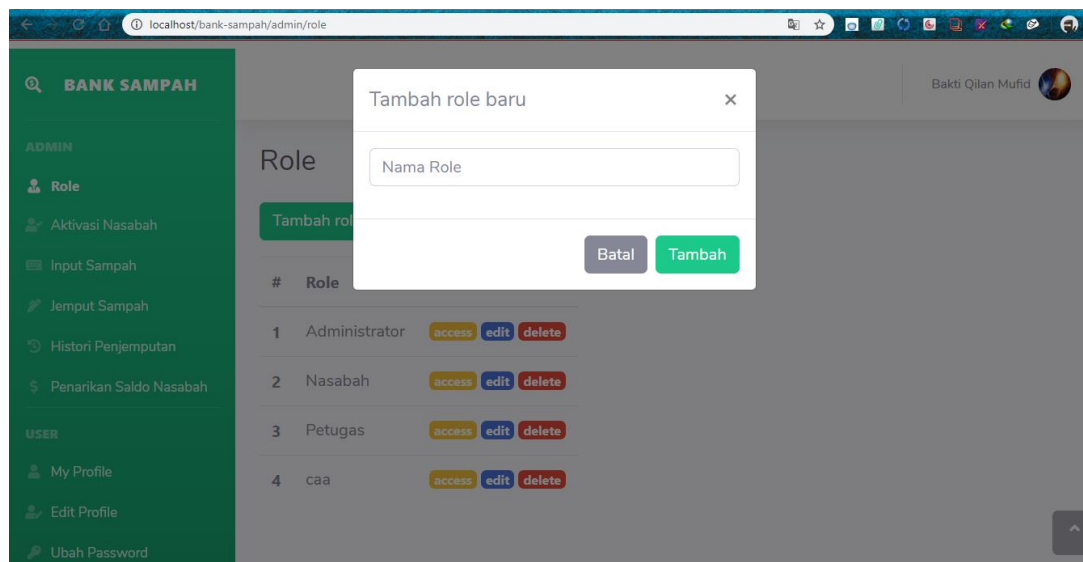
```

a. Tampilan Page Role

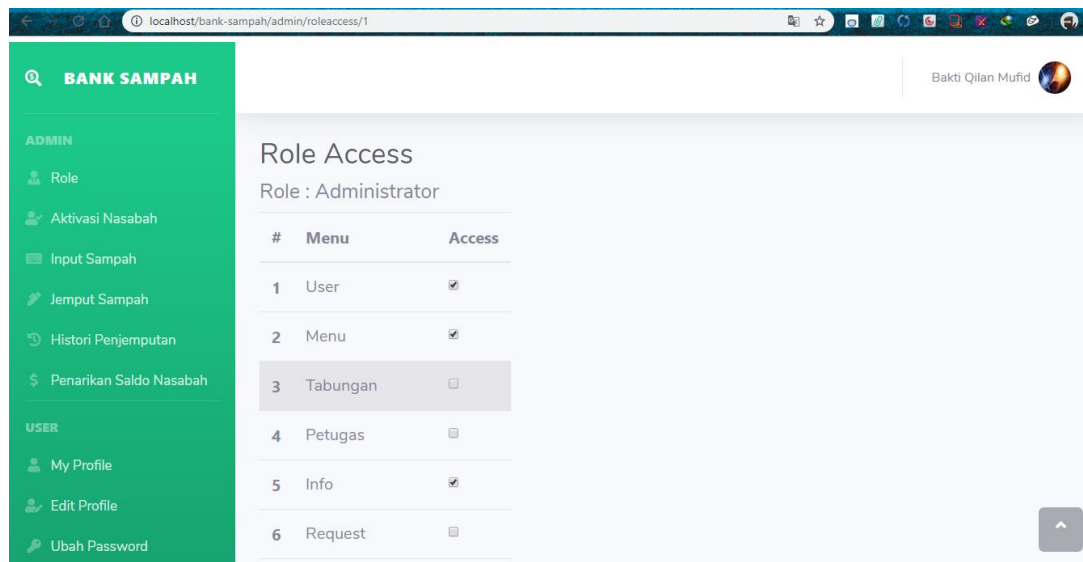


Petunjuk Pemakaian :

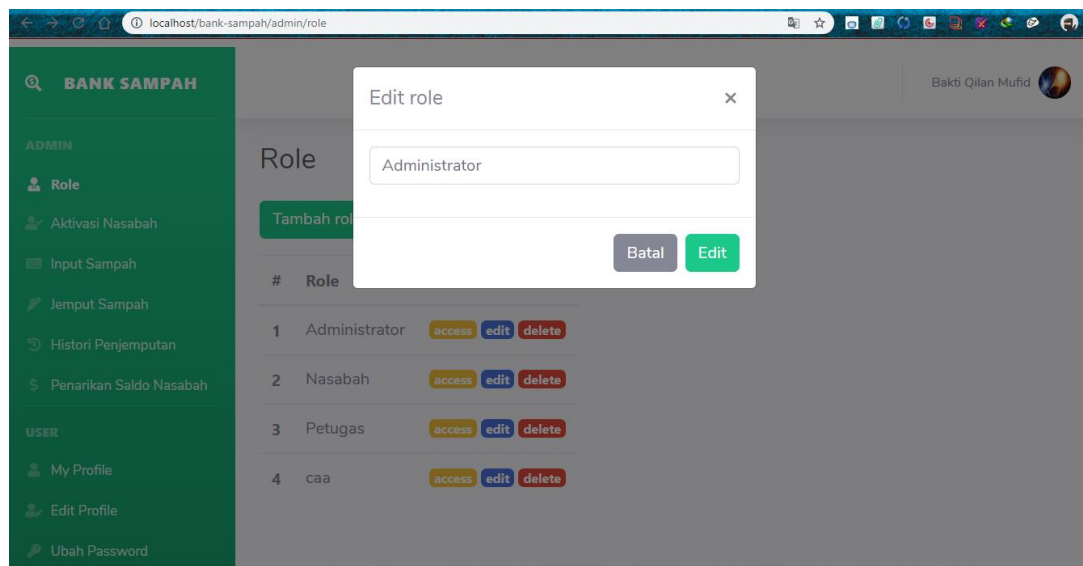
1. Admin mengelola data role
2. Admin dapat melakukan tambah, access, edit dan delete pada role



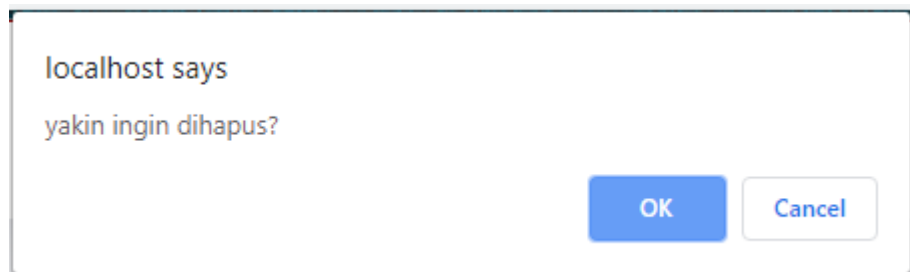
Jika mau menambahkan role baru



<http://localhost/bank-sampah/admin/roleaccess/1>



Jika akan melakukan edit pada data role



Jika akan menghapus data role akan muncul tampilan seperti gambar di atas

Source code pada controllers >> admin.php :

```
public function role()
{
    $data['title'] = 'Role';
    $data['user'] = $this->db-
>get_where('user', ['email' => $this->session-
>userdata('email')])->row_array();
```

```

        $data['role'] = $this->db->get('user_role')-
>result_array();

        $this->form_validation-
>set_rules('role', 'Role', 'required');

        if ($this->form_validation->run() == false) {
            $this->load-
>view('templates/header', $data);
            $this->load-
>view('templates/sidebar', $data);
            $this->load-
>view('templates/topbar', $data);
            $this->load->view('admin/role', $data);
            $this->load->view('templates/footer');
        } else {
            $this->db-
>insert('user_role', ['role' => $this->input-
>post('role')]);
            $this->session-
>set_flashdata('message', '<div class="alert alert-
success mx-
auto" role="alert">Role baru telah ditambahkan!</div>'
);
            redirect('admin/role');
        }
    }

    public function roleAccess($role_id)
    {
        $data['title'] = 'Role Access';
        $data['user'] = $this->db-
>get_where('user', ['email' => $this->session-
>userdata('email')])>row_array();

        $data['role'] = $this->db-
>get_where('user_role', ['id' => $role_id])-
>row_array();
    }
}

```

```

        $this->db->where('id != 1');
        $data['menu'] = $this->db->get('user_menu')-
>result_array();

        $this->load->view('templates/header', $data);
        $this->load->view('templates/sidebar', $data);
        $this->load->view('templates/topbar', $data);
        $this->load->view('admin/role-access', $data);
        $this->load->view('templates/footer');
    }

    public function changeAccess()
    {
        $menu_id = $this->input->post('menuId');
        $role_id = $this->input->post('roleId');

        $data = [
            'role_id' => $role_id,
            'menu_id' => $menu_id
        ];

        $result = $this->db-
>get_where('user_access_menu', $data);

        if ($result->num_rows() < 1) {
            $this->db-
>insert('user_access_menu', $data);
        } else {
            $this->db-
>delete('user_access_menu', $data);
        }

        $this->session-
>set_flashdata('message', '<div class="alert alert-
success mx-
auto" role="alert">Akses telah berubah!</div>');
    }

    public function hapus($id)

```

```

    {
        $this->load->model('Admin_model', 'role');
        $this->role->hapusRole($id);
        $this->session->
>set_flashdata('message', '<div class="alert alert-
success mx-
auto" role="alert">Role berhasil dihapus</div>');
        redirect('admin/role');
    }

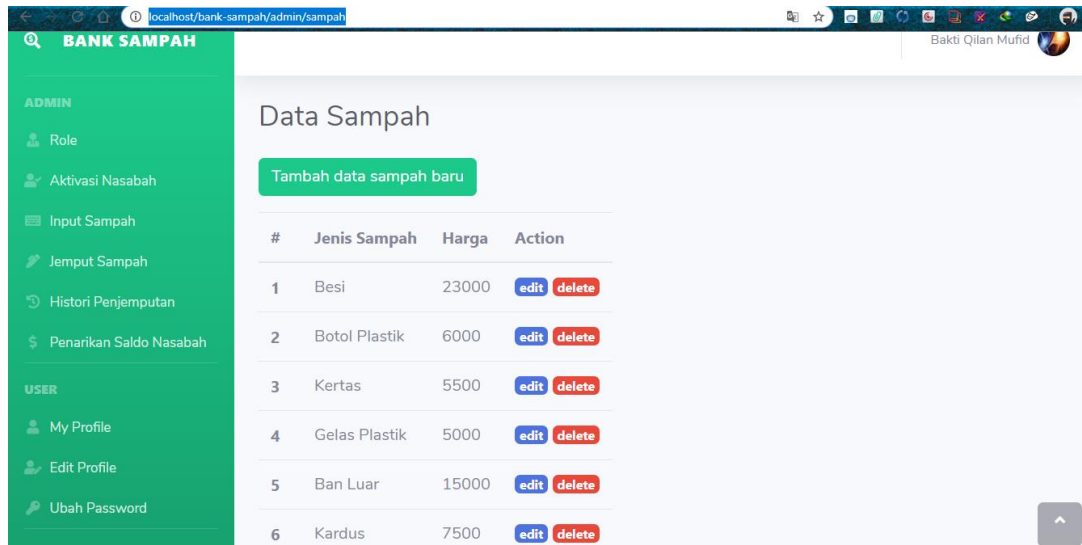
    public function edit()
    {
        $this->form_validation->
>set_rules('role', 'Role', 'required');

        if ($this->form_validation->run() == false) {
            $this->session->
>set_flashdata('message', '<div class="alert alert-
danger mx-
auto" role="alert">Role gagal di edit!</div>');
            redirect('admin/role');
        } else {
            $data = [
                'role' => $this->input->post('role')
            ];
            $this->db->where('id', $_POST['id']);
            $this->db->update('user_role', $data);
            $this->session->
>set_flashdata('message', '<div class="alert alert-
success mx-
auto" role="alert">Role berhasil di edit!</div>');
            redirect('admin/role');
        }
    }
}

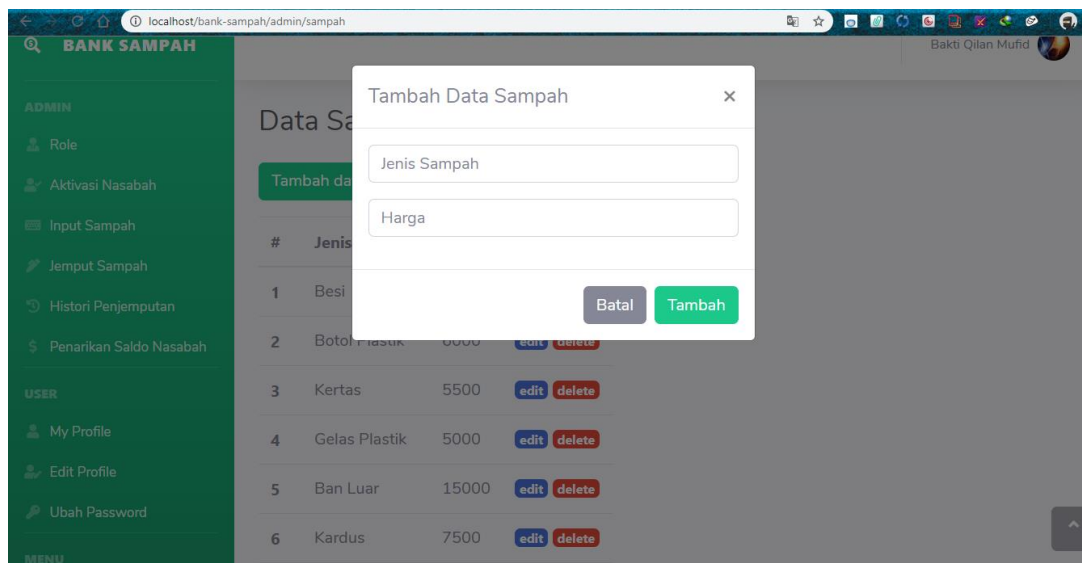
```

Admin akan masuk ke <http://localhost/bank-sampah/admin/sampah> untuk menambahkan, mengedit dan

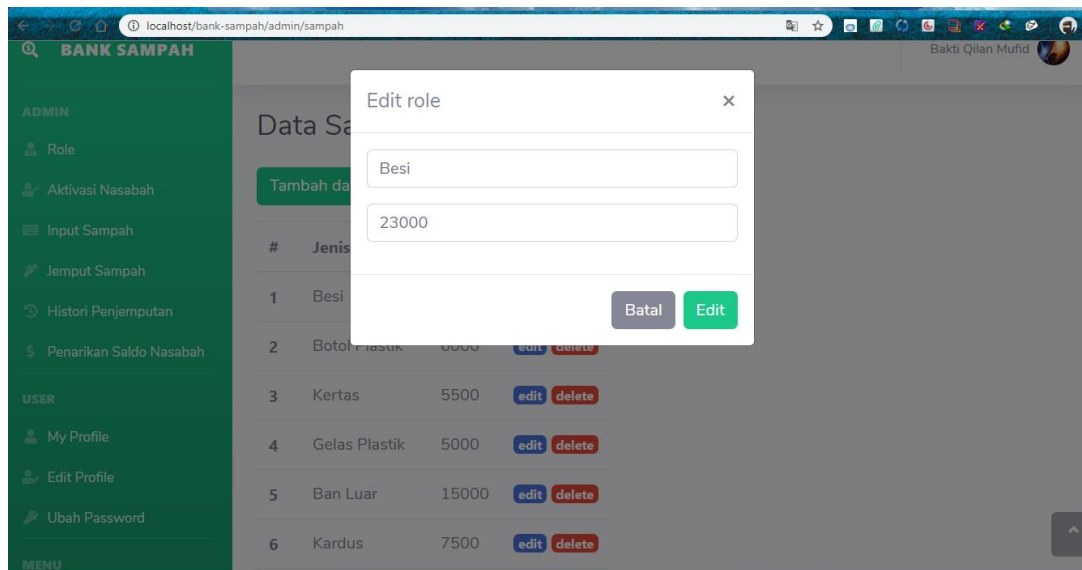
mendelete data-data sampah yang akan di tampilkan pada menu info harga sampah



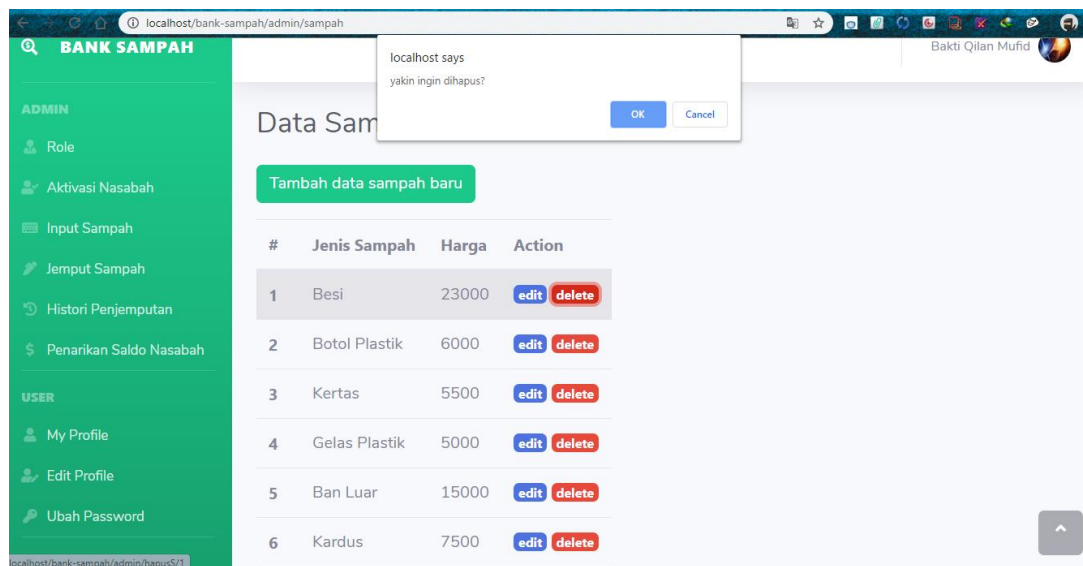
Jika admin akan melakukan proses menambahkan sampah



Jika admin akan mengedit data sampah



Jika admin akan melakukan proses menghapus data sampah maka akan muncul pemberitahuan seperti pada gambar dibawah ini



Source codenya pada admin.php :

```
public function sampah()
{
    $data['title'] = 'Data Sampah';
    $data['user'] = $this->db->get_where('user', ['email' => $this->session->userdata('email')])->row_array();

    $data['sampah'] = $this->db->get('sampah')->result_array();

    $this->form_validation->set_rules('nama', 'Jenis Sampah', 'required', [
        'required' => 'Jenis Sampah harus diisi!'
    ]);

    $this->form_validation->set_rules('harga', 'Harga', 'required', [
        'required' => 'Harga harus diisi!'
    ]);

    if ($this->form_validation->run() == false) {
```

```

        $this->load-
>view('templates/header', $data);
        $this->load-
>view('templates/sidebar', $data);
        $this->load-
>view('templates/topbar', $data);
        $this->load->view('admin/sampah', $data);
        $this->load->view('templates/footer');
    } else {
        $query = [
            'nama' => $this->input->post('nama'),
            'harga' => $this->input-
>post('harga'),
        ];
        $this->db->insert('sampah', $query);
        $this->session-
>set_flashdata('message', '<div class="alert alert-
success mx-
auto" role="alert">Data Sampah telah ditambahkan!</div
>');
        redirect('admin/sampah');
    }
}

public function hapusS($id)
{
    $this->load->model('Admin_model', 'sampah');
    $this->sampah->hapusSampah($id);
    $this->session-
>set_flashdata('message', '<div class="alert alert-
success mx-
auto" role="alert">Data Sampah berhasil dihapus</div>'
);
    redirect('admin/sampah');
}

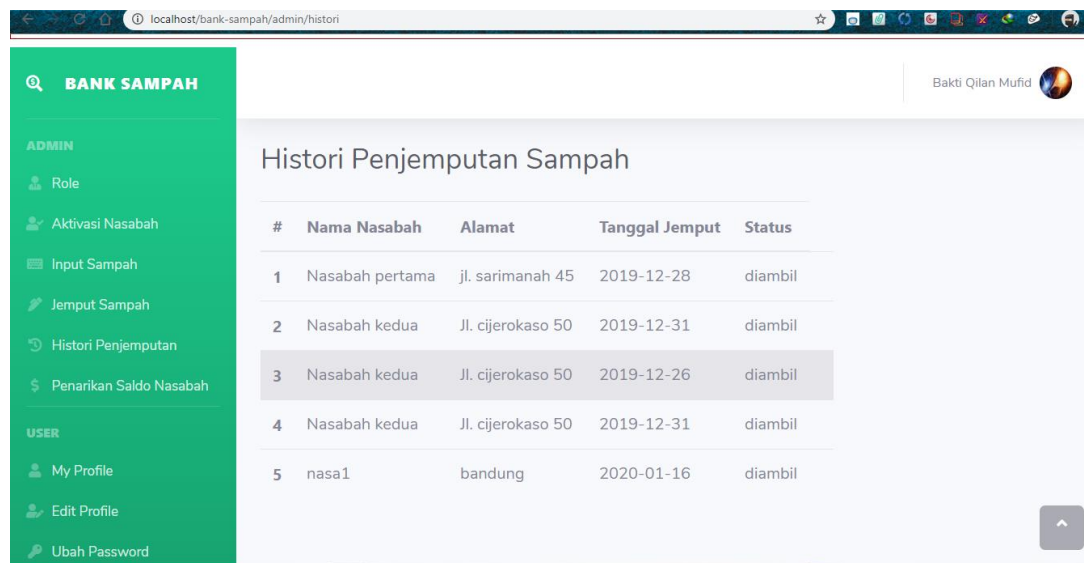
public function editS()
{

```

```
        $this->form_validation->
>set_rules('nama', 'Jenis Sampah', 'required', [
        'required' => 'Jenis Sampah harus diisi!'
    ]);

    if ($this->form_validation->run() == false) {
        $this->session->
>set_flashdata('message', '<div class="alert alert-
danger mx-
auto" role="alert">Data Sampah gagal di edit!</div>');
        redirect('admin/sampah');
    } else {
        $data = [
            'nama' => $this->input->post('nama'),
            'harga' => $this->input->post('harga')
        ];
        $this->db->where('id', $_POST['id']);
        $this->db->update('sampah', $data);
        $this->session->
>set_flashdata('message', '<div class="alert alert-
success mx-
auto" role="alert">Data Sampah berhasil di edit!</div>
');
        redirect('admin/sampah');
    }
}
```

Jika admin akan masuk ke halaman <http://localhost/bank-sampah/admin/histori> untuk mengetahui histori penjemputan sampah



The screenshot shows a web application titled "BANK SAMPAH". The left sidebar is green and contains navigation links for ADMIN (Role, Aktivasi Nasabah, Input Sampah, Jemput Sampah, Histori Penjemputan, Penarikan Saldo Nasabah) and USER (My Profile, Edit Profile, Ubah Password). The main content area is titled "Histori Penjemputan Sampah" and displays a table with the following data:

#	Nama Nasabah	Alamat	Tanggal Jemput	Status
1	Nasabah pertama	Jl. sarimannah 45	2019-12-28	diambil
2	Nasabah kedua	Jl. cijerokaso 50	2019-12-31	diambil
3	Nasabah kedua	Jl. cijerokaso 50	2019-12-26	diambil
4	Nasabah kedua	Jl. cijerokaso 50	2019-12-31	diambil
5	nasa1	bandung	2020-01-16	diambil

Source codenya pada admin.php :

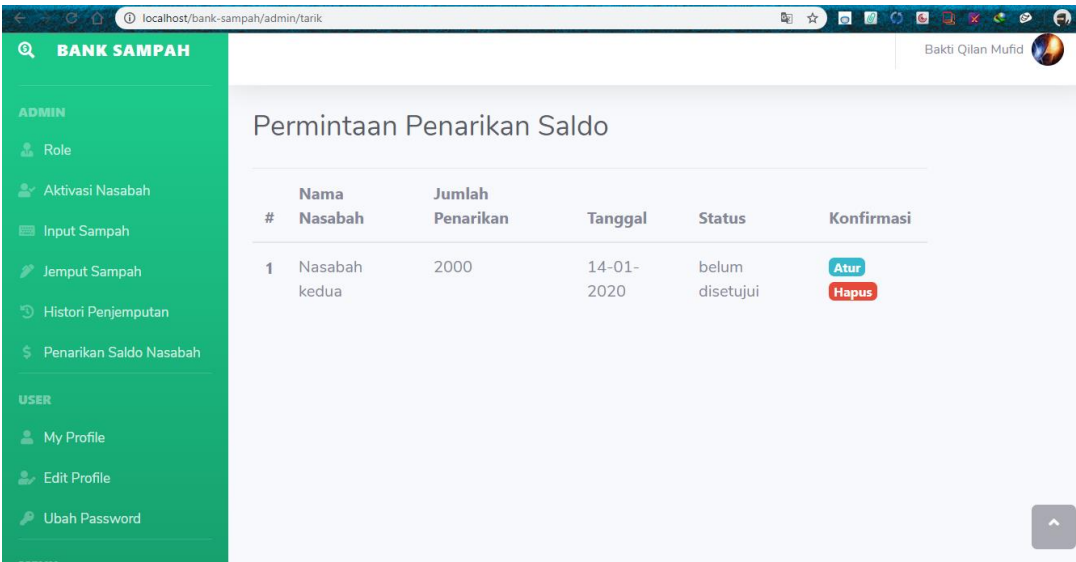
```
public function histori()
{
    $data['title'] = 'Histori Penjemputan Sampah';
    $data['user'] = $this->db->get_where('user', ['email' => $this->session->userdata('email')])->row_array();

    $data['histori'] = $this->admin->getHistori();

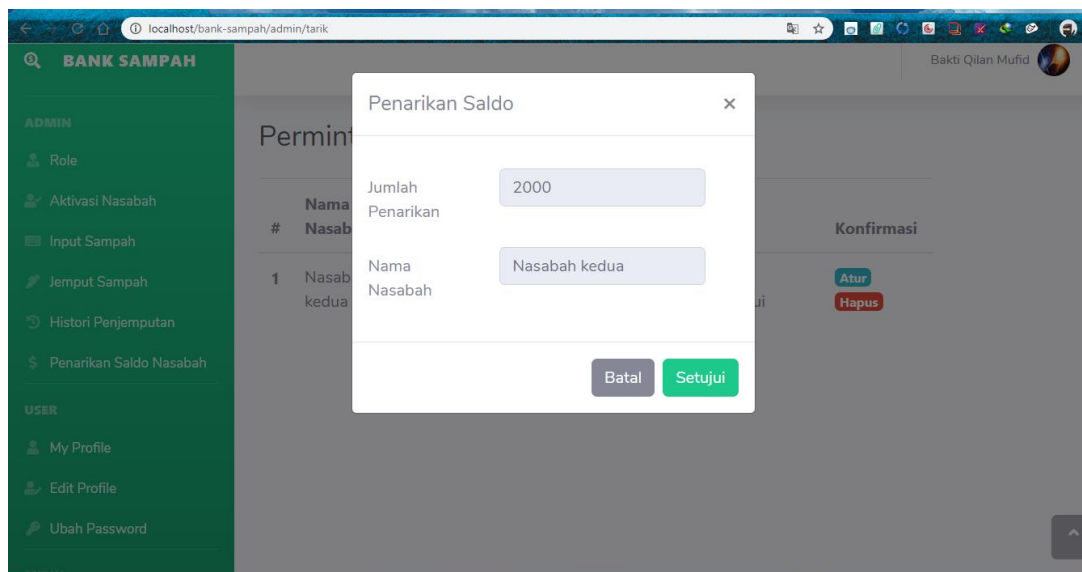
    $this->load->view('templates/header', $data);
    $this->load->view('templates/sidebar', $data);
    $this->load->view('templates/topbar', $data);
    $this->load->view('admin/histori', $data);
}
```

```
} $this->load->view('templates/footer');
```

Kita masuk ke <http://localhost/bank-sampah/admin/tarik> dimana admin akan mengatur saldo dari nasabah yang telah melakukan proses menabung pada aplikasi bank sampah



Jika jika nasabah akan melakukan penarikan saldo maka admin akan mengkonfirmasi saldo tersebut disetujui atau tidak disetujui maka akan muncul seperti ini :



Jika penarikan saldo tersebut disetujui maka akan muncul pemberitahuan seperti ini :

Penarikan Saldo berhasil!!

Source codenya pada controllers >> admin.php :

```
public function tarik()
{
    $data['title'] = 'Permintaan Penarikan Saldo';
    $data['user'] = $this->db-
>get_where('user', ['email' => $this->session-
>userdata('email')])->row_array();

    $data['penarikan'] = $this->admin-
>penarikan();
```

```

        $this->load->view('templates/header', $data);
        $this->load->view('templates/sidebar', $data);
        $this->load->view('templates/topbar', $data);
        $this->load->view('admin/tarik');
        $this->load->view('templates/footer');
    }

    public function saldoTarik()
    {
        $this->form_validation->
>set_rules('id', 'Id', 'required', [
            'required' => 'id Harus dipilih!'
        ]);

        if ($this->form_validation->run() == false) {
            $this->session->
>set_flashdata('message', '<div class="alert alert-
danger mx-
auto" role="alert">Penarikan Saldo gagal di Validasi!<
/div>');

            redirect('admin/tarik');
        } else {
            $query = [
                'user_id' => $this->input->
>post('user_id'),
                'penarikan' => $this->input->
>post('penarikan'),
                'tanggal' => time()
            ];
            $query2 = [
                'status' => 'disetujui'
            ];
            $this->db->insert('tabungan', $query);
            $this->db->where('id', $this->input->
>post('id'));
            $this->db->update('tarik', $query2);
            $this->session->
>set_flashdata('message', '<div class="alert alert-

```

```

success mx-
auto" role="alert">Penarikan Saldo berhasil!</div>');
        redirect('admin/tarik');
    }
}

public function hapusTarik()
{
    $id = $this->input->post('id');
    $this->admin->hapus_tarik($id);
    $this->session-
>set_flashdata('message', '<div class="alert alert-
info mx-
auto" role="alert">Data Penarikan tidak disetujui!</di
v>');
    redirect('admin/tarik');
}
}

```

Source codenya pada controllers >> registrasi.php :

```

class Tabungan extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        cek_role();
        $this->load-
>model('Tabungan_model', 'tabungan');
    }

    public function index()
    {
        $data['title'] = 'Tabungan';
        $data['user'] = $this->db-
>get_where('user', ['email' => $this->session-
>userdata('email')])>row_array();
    }
}

```



```

        $data['tabungan'] = $this->tabungan-
>getTabungan();
        $data['nasabah'] = $this->tabungan-
>getNasabah();
        $data['total'] = $this->tabungan->getTotal();

        $this->load->view('templates/header', $data);
        $this->load->view('templates/sidebar', $data);
        $this->load->view('templates/topbar', $data);
        $this->load->view('tabungan/index', $data);
        $this->load->view('templates/footer');
    }

    public function tarik()
    {
        $data['title'] = 'Meminta Penarikan Saldo';
        $data['user'] = $this->db-
>get_where('user', ['email' => $this->session-
>userdata('email')])->row_array();

        $data['nasabah'] = $this->db-
>get_where('user_detail', ['user_id' => $this-
>session->userdata('id')])->row_array();
        $data['histori'] = $this->tabungan->histori();

        $this->form_validation-
>set_rules('tarik', 'Jumlah Penarikan', 'required|trim
', [
            'required' => 'berapa jumlah yang ingin di
tarik?'
        ]);

        if ($this->form_validation->run() == false) {
            $this->load-
>view('templates/header', $data);
            $this->load-
>view('templates/sidebar', $data);
            $this->load-
>view('templates/topbar', $data);

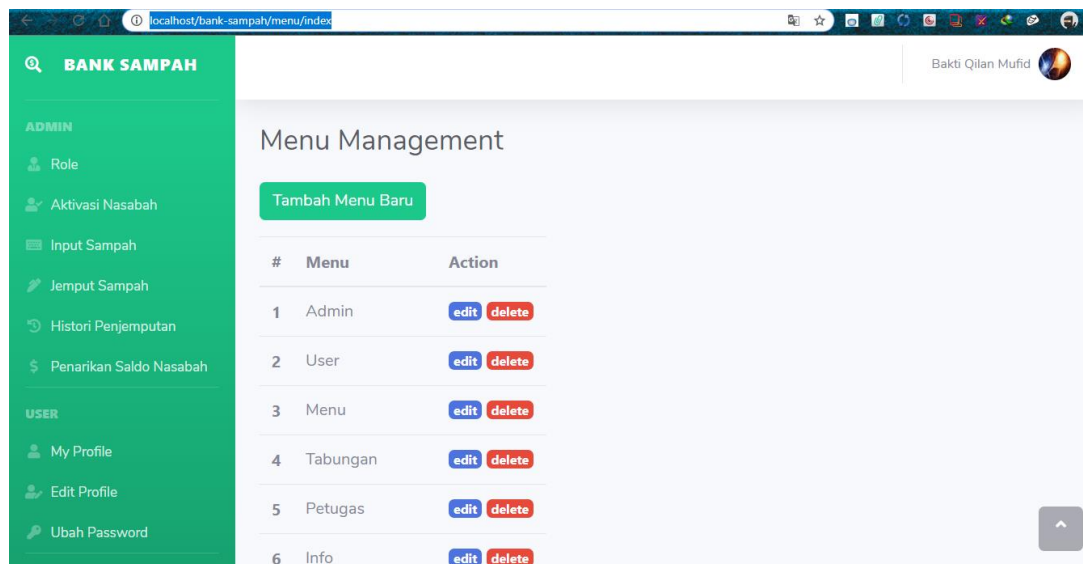
```

```

        $this->load-
>view('tabungan/tarik', $data);
        $this->load->view('templates/footer');
    } else {
        $isi = [
            'user_id' => $this->input-
>post('user_id'),
            'penarikan' => $this->input-
>post('tarik'),
            'tanggal' => time(),
            'status' => 'belum disetujui'
        ];
        $this->db->insert('tarik', $isi);
        $this->session-
>set_flashdata('message', '<div class="alert alert-
success mx-
auto" role="alert">Permintaan Penjemputan telah ditamb
ahkan!</div>');
        redirect('tabungan/tarik');
    }
}
}

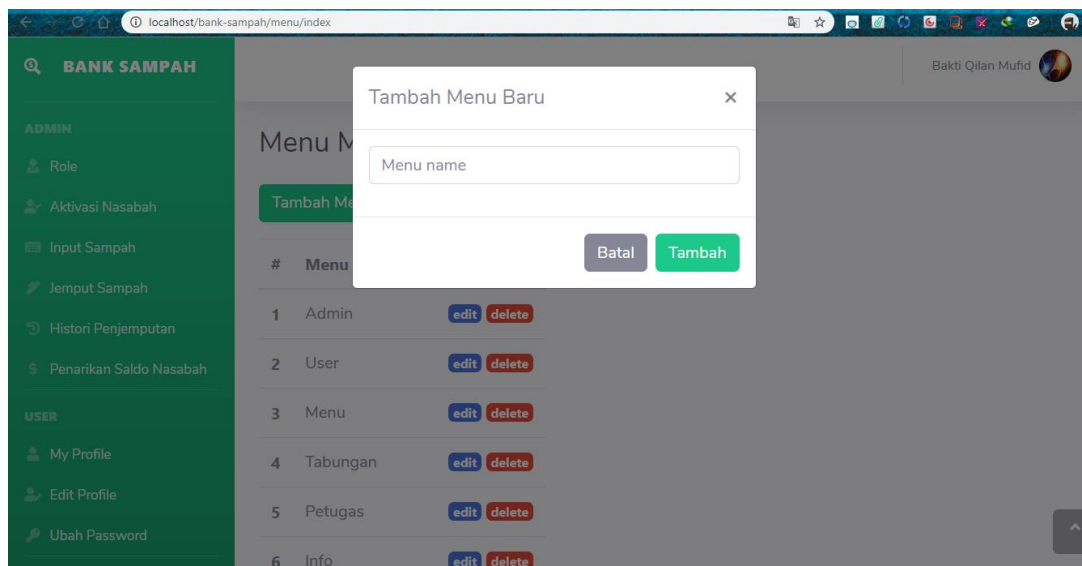
```

Kita masuk ke <http://localhost/bank-sampah/menu/index> pada menu ini admin dapat mengelola menu management seperti gambar dibawah ini :

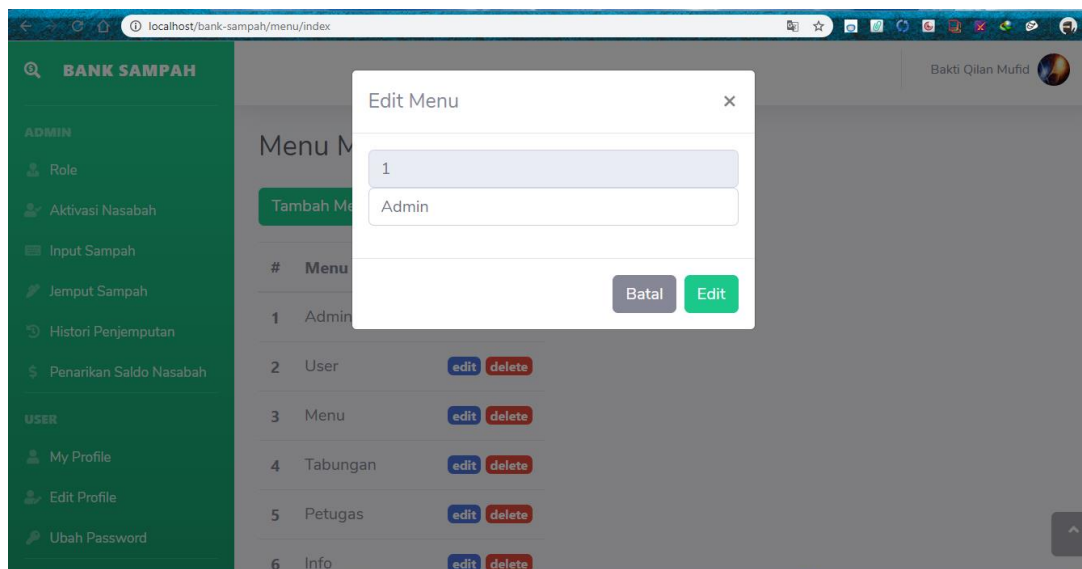


Menu management ini digunakan untuk mengatur setiap menu yang ada pada admin, nasabah dan petugas

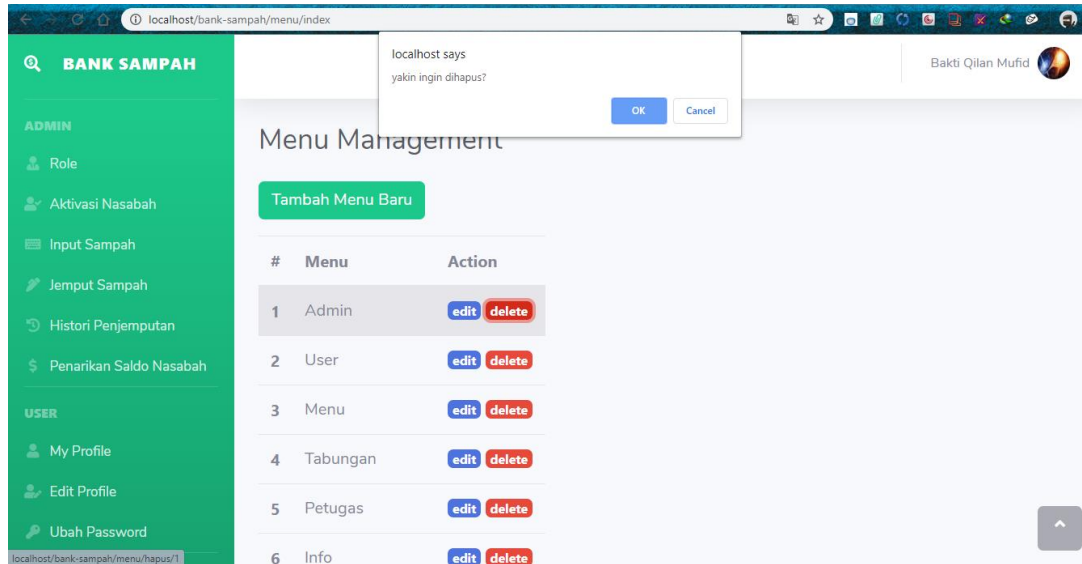
Klik tombol tambah jika admin akan melakukan penambahan pada menu management



Klik tombol edit jika admin akan melakukan proses edit pada menu management



Klik tombol delete jika admin akan menghapus data menu management



Jika tombol delete di klik maka akan muncul pemberitahuan seperti ini :

localhost says
yakin ingin dihapus?



Source codenya pada menu.php :

```
class Menu extends CI_Controller
{
    public function __construct()
    {
```

```

        parent::__construct();
        cek_role();
    }
    public function index()
    {
        $data['title'] = 'Menu Management';
        $data['user'] = $this->db-
>get_where('user', ['email' => $this->session-
>userdata('email')])->row_array();

        $data['menu'] = $this->db->get('user_menu')-
>result_array();

        $this->form_validation-
>set_rules('menu', 'Menu', 'required', [
            'required' => 'Menu harus diisi'
        ]);

        if ($this->form_validation->run() == false) {
            $this->load-
>view('templates/header', $data);
            $this->load-
>view('templates/sidebar', $data);
            $this->load-
>view('templates/topbar', $data);
            $this->load->view('menu/index', $data);
            $this->load->view('templates/footer');
        } else {
            $this->db-
>insert('user_menu', ['menu' => $this->input-
>post('menu')]);
            $this->session-
>set_flashdata('message', '<div class="alert alert-
success mx-
auto" role="alert">Menu baru telah ditambahkan!</div>'
);
            redirect('menu');
        }
    }
}

```

```

        public function edit()
        {
            $this->form_validation-
>set_rules('id', 'Id', 'required', [
                'required' => 'tidak ada id'
            ]);
            $this->form_validation-
>set_rules('menu', 'Menu', 'required');
            if ($this->form_validation->run() == FALSE) {
                $this->session-
>set_flashdata('message', '<div class="alert alert-
danger mx-
auto" role="alert">Menu gagal di edit!</div>');
                redirect('menu');
            } else {
                $data = array(
                    "menu" => $_POST['menu'],
                );
                $this->db->where('id', $_POST['id']);
                $this->db->update('user_menu', $data);
                $this->session-
>set_flashdata('message', '<div class="alert alert-
success mx-
auto" role="alert">Menu berhasil di edit!</div>');
                redirect('menu');
            }
        }

        public function hapus($id)
        {
            $this->load->model('Menu_model', 'menu');
            $this->menu->hapusMenu($id);
            $this->session-
>set_flashdata('message', '<div class="alert alert-
success mx-
auto" role="alert">Menu berhasil dihapus</div>');
            redirect('menu');
        }
    }

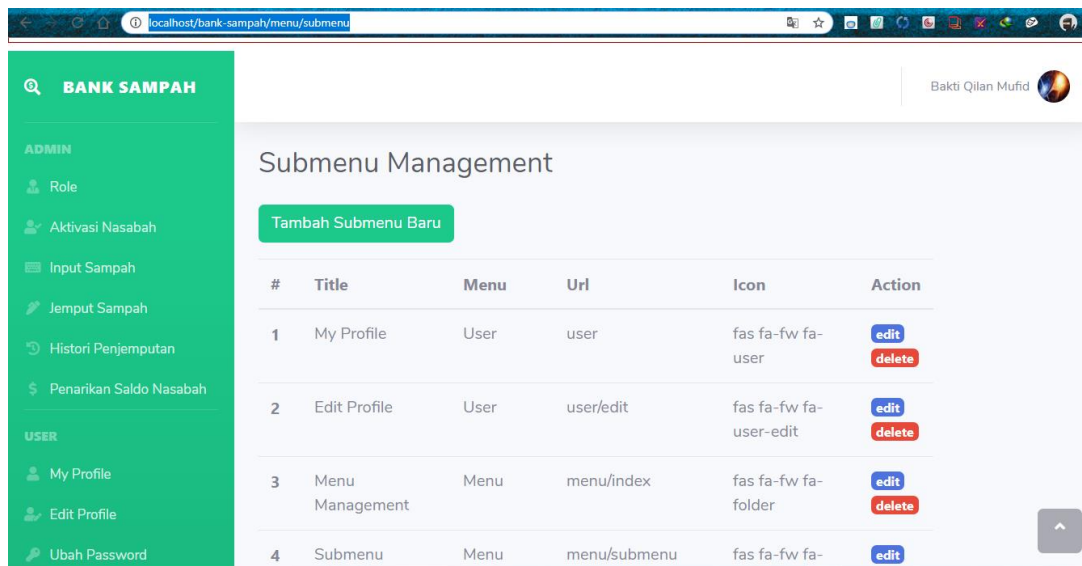
```

```

public function hapussm($id)
{
    $this->load->model('Menu_model', 'menu');
    $this->menu->hapusSubMenu($id);
    $this->session->
    set_flashdata('message', '<div class="alert alert-success mx-auto" role="alert">Submenu berhasil dihapus</div>');
    redirect('menu/submenu');
}

```

Masuk ke <http://localhost/bank-sampah/menu/submenu> disini admin melakukan pengelolaan sub menu



BANK SAMPAH

ADMIN

- Role
- Aktivasi Nasabah
- Input Sampah
- Jemput Sampah
- Histori Penjemputan
- Penarikan Saldo Nasabah

USER

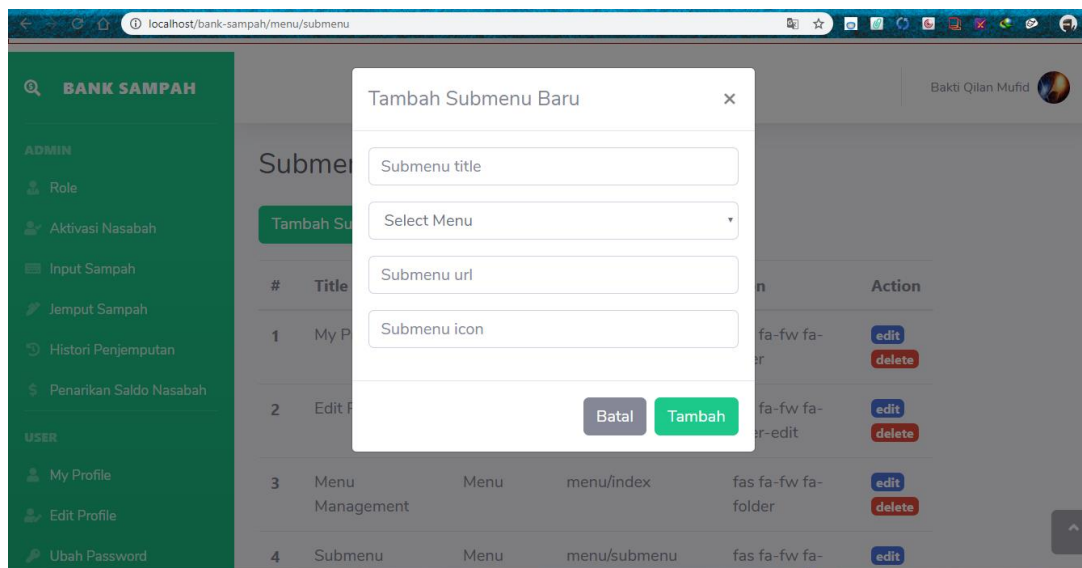
- My Profile
- Edit Profile
- Ubah Password

Submenu Management

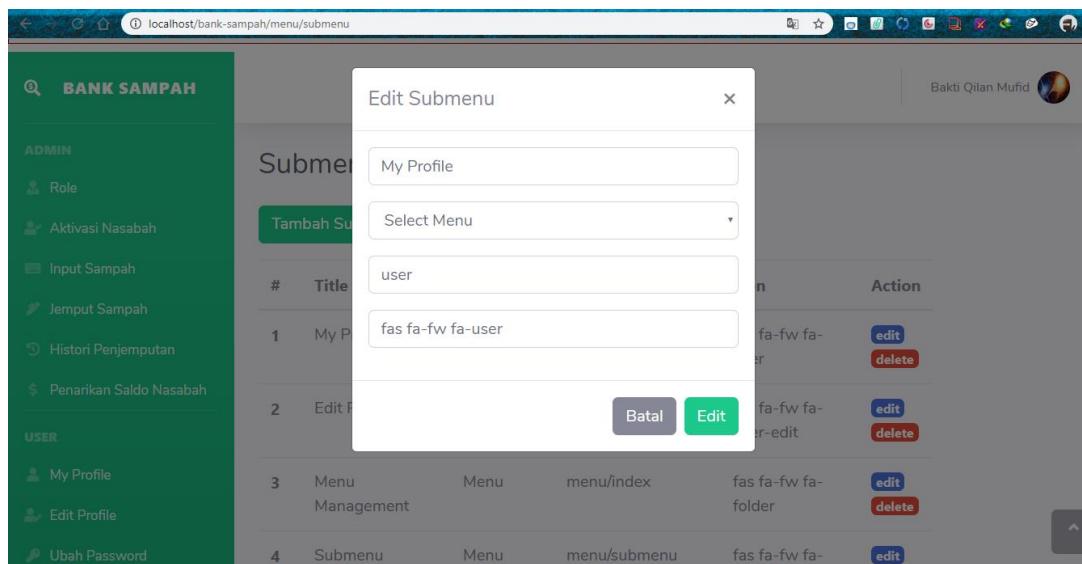
Tambah Submenu Baru

#	Title	Menu	Url	Icon	Action
1	My Profile	User	user	fas fa-fw fa-user	edit delete
2	Edit Profile	User	user/edit	fas fa-fw fa-user-edit	edit delete
3	Menu Management	Menu	menu/index	fas fa-fw fa-folder	edit delete
4	Submenu	Menu	menu/submenu	fas fa-fw fa-	edit

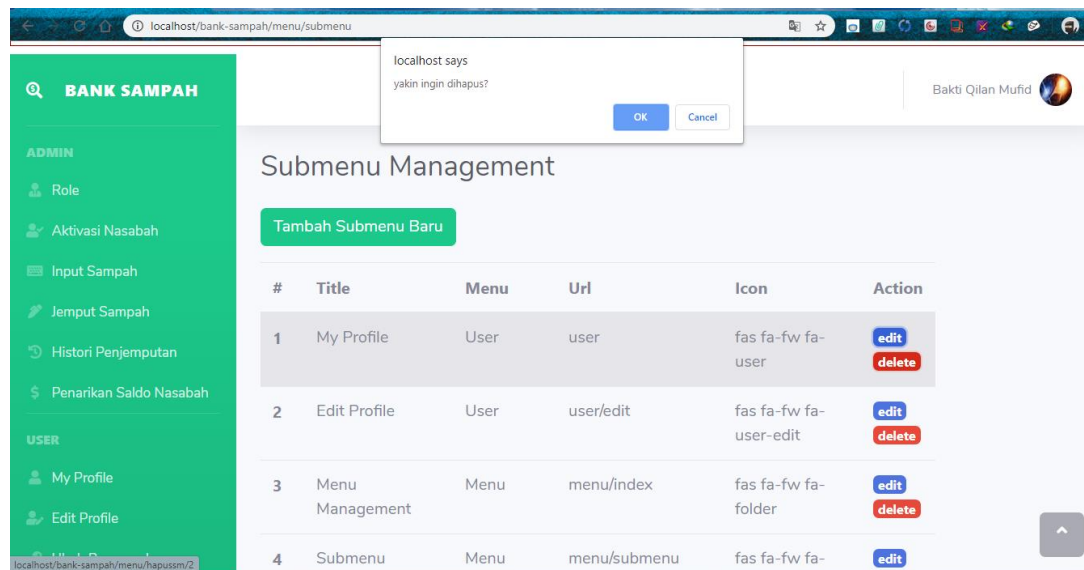
Jika akan menambahkan sub menu



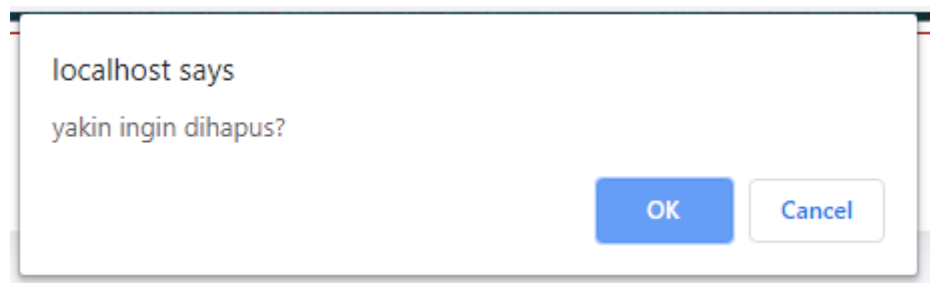
Jika akan mengedit sub menu



Jika akan menghapus data sub menu



Maka akan muncul pemberitahuan seperti ini



Source codenya pada menu.php :

```
public function submenu()  
{  
    $data['title'] = 'Submenu Management';  
}
```

```

        $data['user'] = $this->db-
>get_where('user', ['email' => $this->session-
>userdata('email')])->row_array();

        $this->load->model('Menu_model', 'menu');
        $data['subMenu'] = $this->menu->getSubMenu();
        $data['menu'] = $this->db->get('user_menu')-
>result_array();

        $this->form_validation-
>set_rules('title', 'Title', 'required', [
            'required' => 'title harus diisi'
        ]);
        $this->form_validation-
>set_rules('menu_id', 'Menu', 'required', [
            'required' => 'menu harus diisi'
        ]);
        $this->form_validation-
>set_rules('url', 'Url', 'required', [
            'required' => 'url harus diisi'
        ]);
        $this->form_validation-
>set_rules('icon', 'Icon', 'required', [
            'required' => 'icon harus diisi'
        ]);

        if ($this->form_validation->run() == false) {

            $this->load-
>view('templates/header', $data);
            $this->load-
>view('templates/sidebar', $data);
            $this->load-
>view('templates/topbar', $data);
            $this->load->view('menu/submenu', $data);
            $this->load->view('templates/footer');
        } else {
            $data = [

```

```

        'title' => $this->input->post('title'),
        'menu_id' => $this->input->post('menu_id'),
        'url' => $this->input->post('url'),
        'icon' => $this->input->post('icon')
    ];
    $this->db->insert('user_sub_menu', $data);
    $this->session->flashdata('message', '<div class="alert alert-success mx-auto" role="alert">Submenu baru telah ditambahkan!</div>');
    redirect('menu/submenu');
}

public function editism()
{
    $this->form_validation->set_rules('id', 'Id', 'required', [
        'required' => 'tidak ada id'
    ]);
    $this->form_validation->set_rules('menu_id', 'Menu', 'required');
    $this->form_validation->set_rules('title', 'Title', 'required');
    $this->form_validation->set_rules('url', 'Title', 'required');
    $this->form_validation->set_rules('icon', 'Icon', 'required');

    if ($this->form_validation->run() == FALSE) {
        $this->session->flashdata('message', '<div class="alert alert-danger mx-auto" role="alert">Menu gagal di edit!</div>');
        redirect('menu/submenu');
    } else {

```

```
        $data = [  
            'title' => $this->input->  
>post('title'),  
            'menu_id' => $this->input->  
>post('menu_id'),  
            'url' => $this->input->post('url'),  
            'icon' => $this->input->post('icon')  
        ];  
        $this->db->where('id', $_POST['id']);  
        $this->db->update('user_sub_menu', $data);  
        $this->session->  
>set_flashdata('message', '<div class="alert alert-  
success mx-  
auto" role="alert">Menu berhasil di edit!</div>');  
        redirect('menu/submenu');  
    }  
}  
}
```

1.1 Analisis

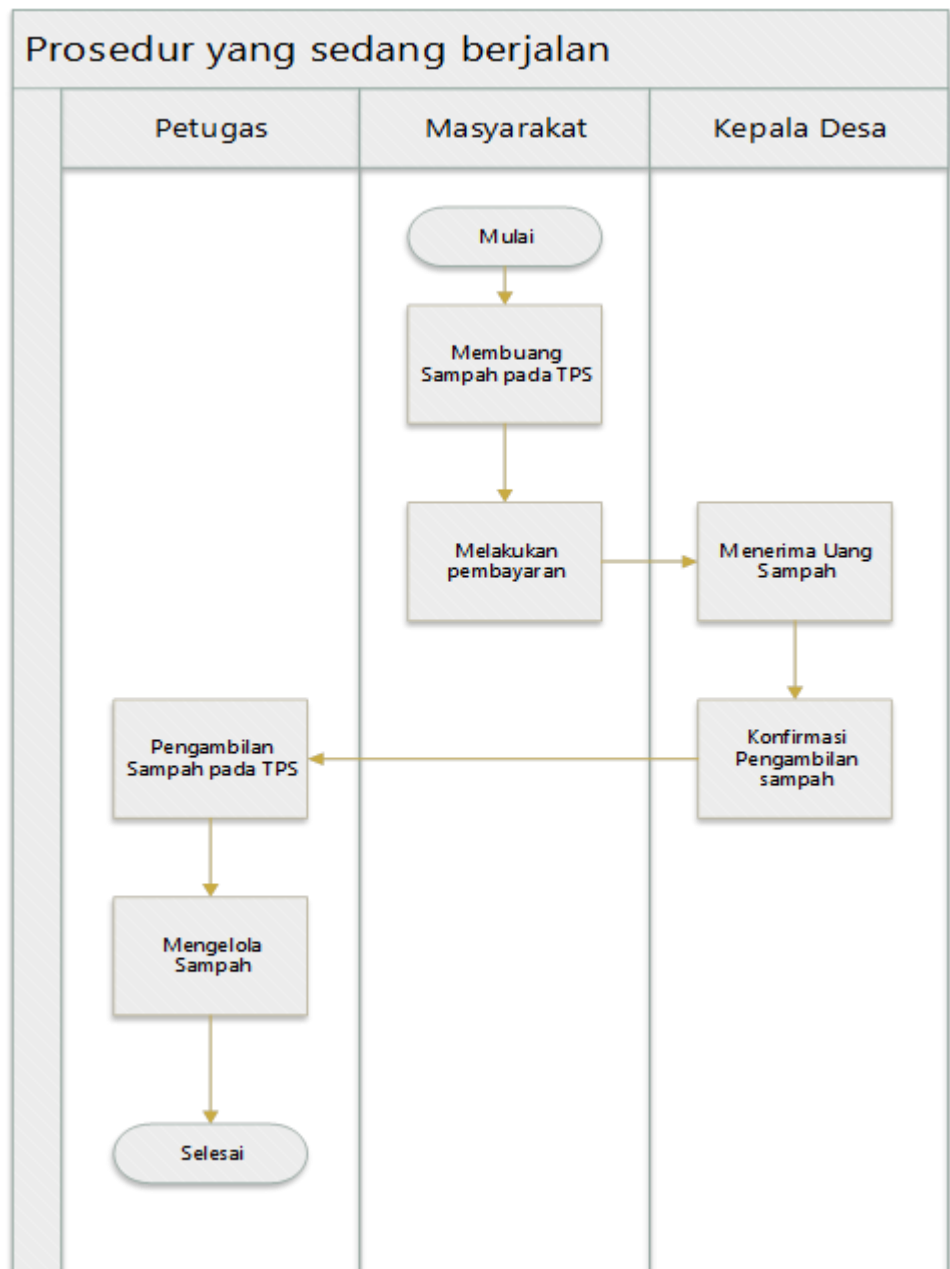
Analisis merupakan langkah awal untuk pengembangan sebuah aplikasi, karena perancangan dan bahkan pengembangan implementasi aplikasi tidak akan berjalan dengan baik tanpa adanya analisa terhadap aplikasi yang akan digunakan. Analisis juga dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi masalah-masalah, kesempatan-kesempatan, hambatan-hambatan yang terjadi serta kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan agar mendapat hasil yang maksimal.

Analisis yang dilakukan terhadap Aplikasi Bank Sampah Menggunakan CodeIgniter ini dibuat menggunakan flowmap dan metode Object Oriented yang memberikan gambaran mengenai proses yang terdapat di dalam aplikasi tersebut.

Sistem ini dibangun dengan menggunakan model MVC atau Model View Controller, dimana MVC merupakan model pengembangan dari Object Oriented Programming, dimana setiap baris kodenya dipisahkan menjadi tiga bagian, yaitu ada pada view sebagai form, lalu controller untuk menyimpan fungsi dan class dan model untuk menyimpan database.

1.1.1 Analisis sistem yang sedang berjalan

Sistem yang berjalan saat ini terdiri dari satu prosedur yaitu alur pemasukan barang yang sedang berjalan.



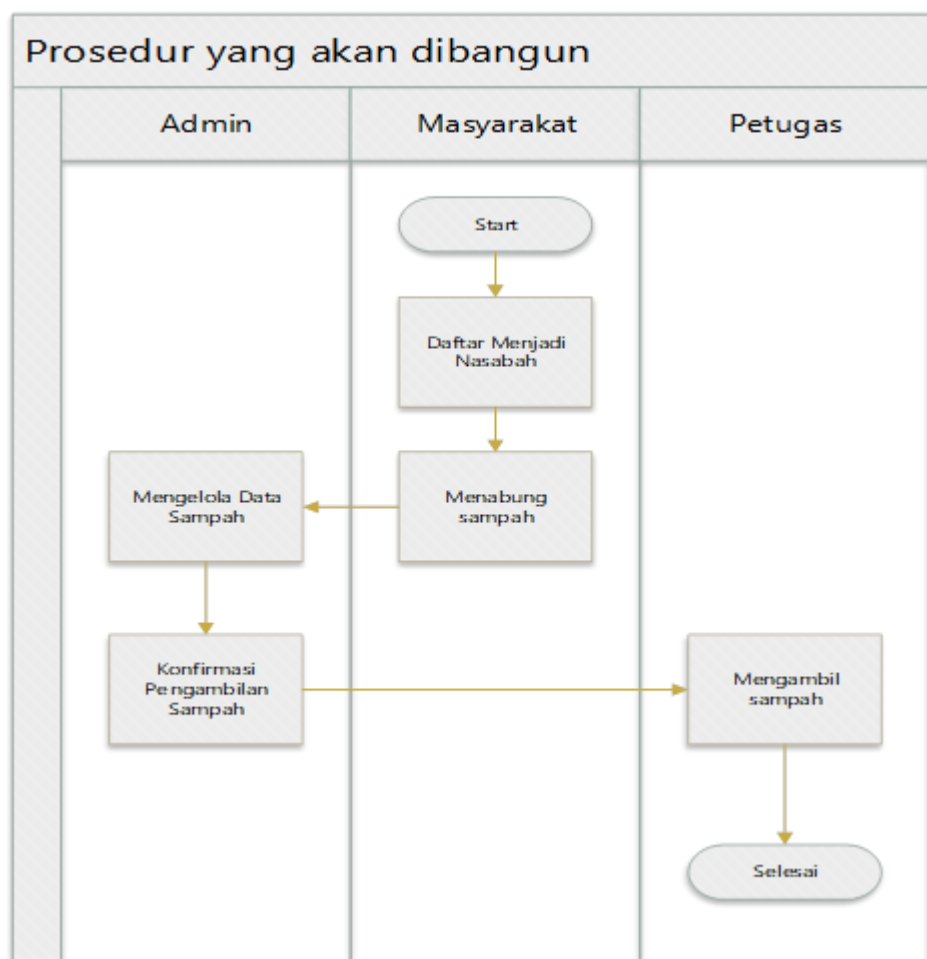
Gambar 3.1 Prosedur yang sedang berjalan

Keterangan :

1. Masyarakat membuang sampah pada TPS yang sudah tersedia di depan rumahnya masing-masing
2. Masyarakat Melakukan pembayaran ke kepala desa setiap sebulan sekali
3. Kepala desa mengkonfirmasi ke pada petugas kebersihan untuk melakukan pengambilan sampah setiap seminggu 2 kali.
4. Petugas kebersihan akan mengambil sampah setiap 2 kali dalam seminggu.
5. Petugas kebersihan akan mengelola sampah, sampah apa saja yang akan di daur ulang dan sampah apa saja yang akan di bakar.

1.1.2 Analisis Sistem yang akan di bangun

Pada analisis sistem yang akan dibangun ini, dibuat beberpa pembaruan dari yang sebelumnya. Pada prosedur ini masyarakat di haruskan mendaftarkan terlebih dahulu untuk menjadi nasabah pada Bank Sampah. Prosedur yang akan dibangun pada Bank Sampah yaitu sebagai berikut :



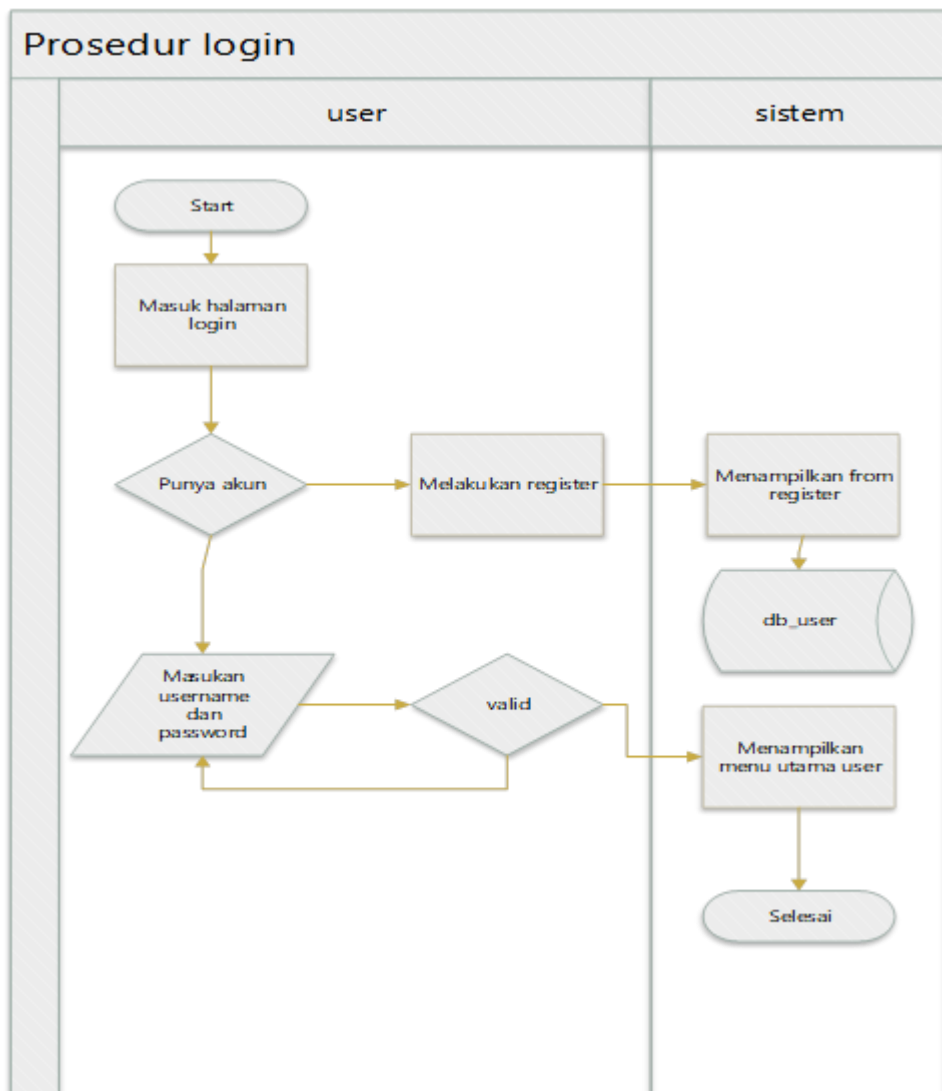
Gambar 3.2 Prosedur yang akan dibangun

Keterangan :

1. Masyarakat harus mendaftarkan diri terlebih dahulu untuk menjadi nasabah pada bank sampah
2. Setelah menjadi nasabah, masyarakat dapat menabung sampah pada bank sampah
3. Jika masyarakat menabung sampah, maka data-data sampah tersebut akan di kelola oleh admin
4. Admin akan mengkonfirmasi kepada petugas untuk melakukan pengambilan sampah pada nasabah yang telah menabung sampah
5. Petugas akan diberi alamat nasabah oleh admin
6. Petugas melakukan pengambilan sampah ke alamat nasabah

1.1.3 Analisis Sistem Login yang akan di bangun

Pada analisis sistem yang akan dibangun ini, dibuat beberapa pembaruan dari yang sebelumnya. Pada prosedur ini masyarakat di haruskan mendaftarkan terlebih dahulu untuk menjadi nasabah pada Bank Sampah. Prosedur login yang akan dibangun pada Bank Sampah yaitu sebagai berikut :



Gambar 3.3 Prosedur Login

Keterangan :

1. User akan masuk ke halaman login
2. Jika sudah memiliki akun maka user akan langsung melakukan login dan masuk ke halaman user
3. Jika belum memiliki akun maka user akan melakukan registrasi terlebih dahulu
4. User disini yaitu nasabah atau petugas

3.1.3.1 Kebutuhan Fungsional (*Functional Requirements*)

Kebutuhan fungsional adalah jenis kebutuhan yang berisi proses-proses apa saja yang nantinya dilakukan oleh sistem. Kebutuhan fungsional juga berisi informasi-informasi apa saja yang harus ada dan dihasilkan oleh sistem.

Adapun kebutuhan fungsional yang akan dibuat yaitu:

1. Login
2. Kelola data user
3. Kelola data sampah
4. Menabung sampah
5. Pengambilan sampah

Setiap proses memiliki representasi masing-masing pada sebuah tabel atau data yang terdapat pada *database* yang telah dirancang sebelumnya. Dan setiap proses berhubungan langsung dengan entitas atau *user*.

3.1.3.2 Kebutuhan Non-Fungsional (*Non-Functional Requirement*)

Analisis kebutuhan non fungsional dilakukan untuk mengetahui spesifikasi kebutuhan untuk sistem. Spesifikasi kebutuhan melibatkan analisis perangkat keras/hardware, analisis perangkat lunak/software, analisis pengguna/user.

Adapun kebutuhan non fungsional yang akan dibuat adalah sebagai berikut :

A. Kebutuhan Perangkat Keras

Pembuatan aplikasi ini menggunakan perangkat sebagai berikut :

Tabel 3.1 Kebutuhn perangkat keras

No.	Jenis		Keterangan
1	Processor	:	Intel® core™i3
2	Memory	:	4 GB
3	Monitor	:	LCD 14,1 Inchi
4	Mouse dan keyboard	:	Standard

B. Kebutuhan Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan adalah sebagai berikut :

Tabel 3. 2 Kebutuhan perangkat lunak

No.	Jenis	Keterangan
1.	Sistem Operasi	Windows 10 Pro 64-Bit

2.	Server <i>Database</i>	Xampp 1.8.1
3.	Bahasa Pemrograman	PHP dan Android
4.	Software Pendukung	Visual Studio Code dan Android Studio
5.	Browser	Google Chrome

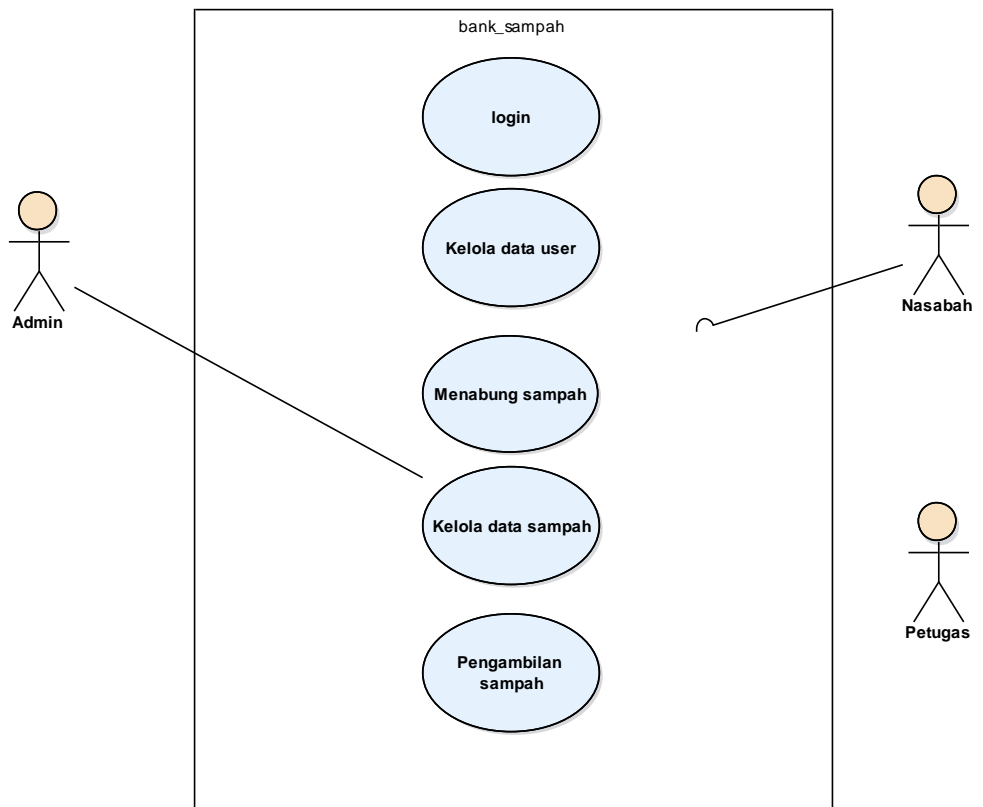
C. Analisis Pengguna/User

Aplikasi yang akan dibuat ini digunakan ketika ingin menabung sampah dan melakukan penjemputan sampah, adapun User yang dilibatkan antara lain : Nasabah (Masyarakat) dan petugas kebersihan.

1.2 Perancangan

1.2.1 Use Case Diagram

Use case diagram mendeskripsikan sebuah interaksi antara satu atau lebih actor dengan sistem informasi yang akan dibuat. Use case diagram digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi



Gambar 3.4 Use Case Diagram

dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

1.2.1.1 Definisi Aktor

Pada bagian ini akan dideskripsikan actor-aktor yang terlibat dalam Aplikasi Bank Sampah

Tabel 3.3 Definisi Aktor

No	Aktor	Deskripsi
1.	Admin	<ul style="list-style-type: none"> - Mengelola data sampah - Mengelola data user - Mengatur jadwal pengambilan sampah
2.	Nasabah	<ul style="list-style-type: none"> - Menabung sampah
3.	Petugas	<ul style="list-style-type: none"> - Kelola data sampah

1.2.1.2 Definisi Use Case

Use case digunakan untuk mengetahui fungsi apa aja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Tabel 3.4 Definisi Use Case

No	Use Case	Deskripsi
1.	Login	Merupakan aktifitas validasi user yang bisa melakukan akses data kedalam sistem.
2.	Kelola data user	Merupakan aktifitas untuk mengelola data sampah,
3	Kelola data sampah	Merupakan aktifitas untuk mengelola data nasabah
4	Menabung sampah	Merupakan aktifitas untuk meminta jadwal pengambilan sampah pada admin
.5.	Pengambilan sampah	Merupakan aktifitas untuk melakukan pengambilan sampah

1.2.1.3 Skenario Use Case

Skenario Use Case mendeskripsikan urutan langkah-langkah dalam proses bisnis, baik yang dilakukan aktor terhadap sistem maupun yang dilakukan oleh sistem terhadap actor.

Tabel 3.5 Scenario Diagram Login

Identifikasi	
Nomor	1
Nama	Login
Tujuan	Merupakan aktifitas validasi yang bisa melakukan akses data ke dalam sistem.
Deskripsi	
Aktor	Admin, Nasabah dan Petugas
Skenario Utama	
Kondisi Awal	form sudah tersedia
Aksi Aktor	Reaksi Sistem
1. Masukan <i>username</i> dan <i>password</i> 2. Jika nasabah dan petugas belum memiliki akun maka harus melakukan registrasi akun terlebih dahulu	3. Mencocokkan data <i>login</i> dengan data Admin, nasabah dan petugas pada basis data 4. Menyimpan data yang telah melakukan <i>register</i> pada basis data
Kondisi Akhir	Jika admin, nasabah dan petugas memasukkan <i>username</i> dan <i>password</i> berhasil, maka akan masuk ke halamannya masing-masing

Tabel 3.6 Scenario Diagram Kelola Data User

Identifikasi	
Nomor	2
Nama	Kelola data user

Tujuan	Merupakan aktifitas untuk mengelola data nasbah		
Deskripsi			
Aktor	Admin		
Skenario Utama			
Kondisi Awal		form sudah tersedia	
Aksi Aktor		Reaksi Sistem	
1. Admin mengelola data role 2. Admin mengelola aktivasi nasbah		3. Sistem menjalankan perintah untuk mengelola data role 4. Sistem menjalankan perintah aktivasi	
Kondisi Akhir		Jika admin telah melakukan aktivasi pada nasabah maka nasabah akan mendapatkan no rekening secara otomatis	

Tabel 3.7 Scenario Diagram Kelola Sampah

Identifikasi	
Nomor	3
Nama	Kelola Data Sampah
Tujuan	Merupakan aktifitas untuk mengelola data sampah
Deskripsi	
Aktor	Admin
Skenario Utama	
Kondisi Awal	form sudah tersedia
Aksi Aktor	Reaksi Sistem
1. Admin memilih perintah (edit, tambah, dan delete)	2. Sistem menjalankan perintah yang dipilih oleh admin
Kondisi Akhir	Jika perintah berhasil dijalankan maka sistem akan menyimpan, mengubah dan menghapus data tergantung perintah apa yang admin jalankan .

Tabel 3.8 Scenario Diagram menabung Sampah

Identifikasi	
Nomor	4

Nama	Menabung Sampah
Tujuan	Merupakan aktifitas untuk meminta jadwal pengambilan sampah
Deskripsi	
Aktor	Nasabah
Skenario Utama	
Kondisi Awal	form sudah tersedia
Aksi Aktor	Reaksi Sistem
1. Nasabah mengatur tanggal pengambilan sampah	1. Sistem menjalankan perintah yang dipilih oleh nasabah
Kondisi Akhir	Jika perintah berhasil dijalankan maka tanggal yang telah ditetapkan oleh nasabah akan tersimpan pada baasis data dan data tersebut akan di tampilkan pada halaman pengambilan sampah .

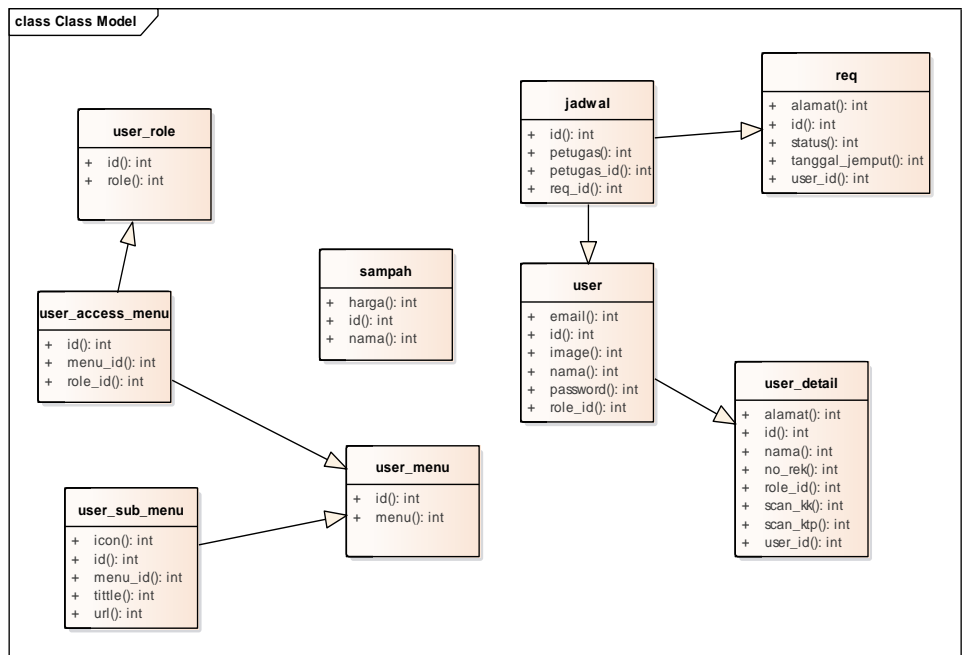
Tabel 3.9 Scenario Diagram Pengambilan Sampah

Identifikasi	
Nomor	5
Nama	Pengambilan Sampah
Tujuan	Merupakan aktifitas untuk mengelola pengambilan sampah
Deskripsi	
Aktor	Admin dan petugas
Skenario Utama	

Kondisi Awal	form sudah tersedia
Aksi Aktor	Reaksi Sistem
1. Admin mengatur pembagian pengambilan sampah pada petugas 2. Petugas menginputkan data sampah	3. Sistem menjalankan perintah yang dipilih oleh Admin 4. Sistem akan menjalankan perintah yang dipilih oleh petugas
Kondisi Akhir	Jika perintah yang dipilih oleh admin maka sistem akan menampilkan data pengambilan sampah pada petugas dan jika perintah yang telah dijalankan oleh petugas maka sistem akan menyimpan data.

1.2.2 Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas – kelas yang akan dibuat untuk membangun sistem.

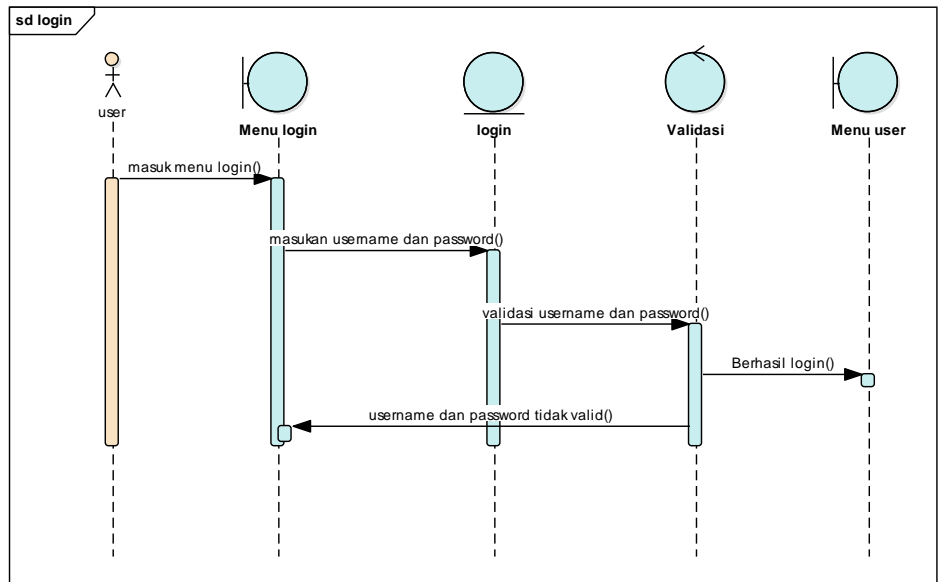


Gambar 3.5 Class Diagram

1.2.3 Sequence Diagram

Sequence diagram menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek.

1.2.3.1 Squence Diagram Login

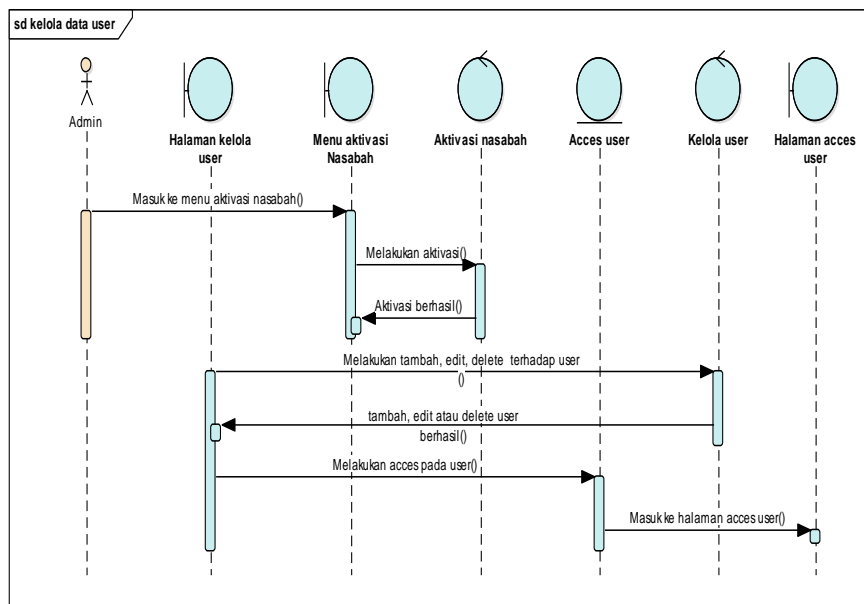


Gambar 3.6 Sequence Diagram Kelola Login

Deskripsi Squence Login :

Pada *Sequence Diagram* ini menjelaskan proses *login*. Admin, nasabah, dan petugas mulai memasukkan *username* dan *password* pada menu *login*. Kemudian *database* akan mengecek *validasi*, jika *username* dan *password* benar maka akan masuk ke sistem dan tampil halaman awal sistem. Jika *username* dan *password* salah maka kembali memasukkan *username* dan *password*.

1.2.3.2 Squence Diagram Kelola User

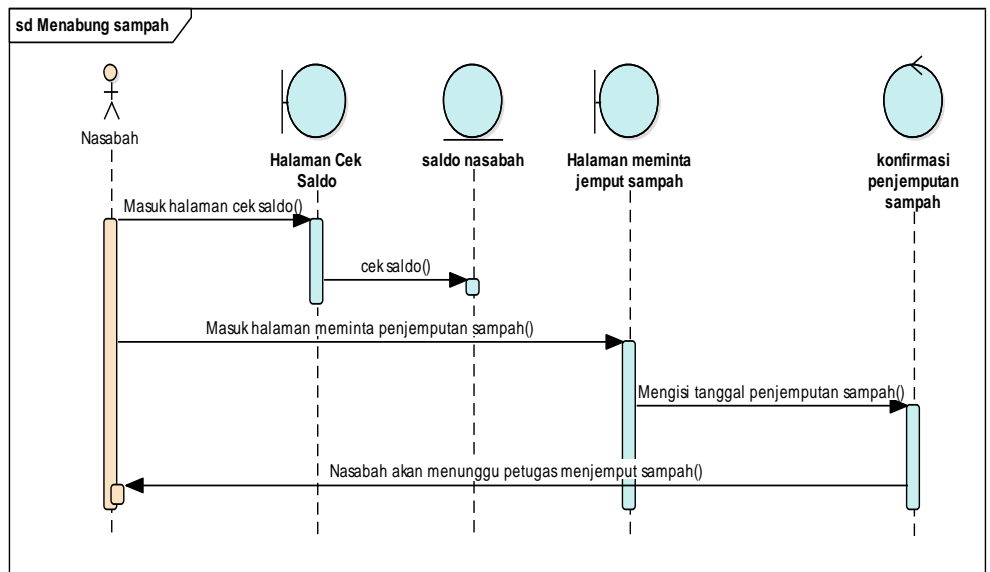


Gambar 3.7 Squence Diagram Kelola User

Deskripsi Squence Kelola User:

Pada *Sequence Diagram* ini menjelaskan admni yang melakukan proses pengelolaan data nasabah memilih menu aktivasi nasabah lalu sistem akan memproses dan memberikan no.rekening kepada nasabah. Selanjutnya nasabah akan mendapatkan no.rekening dan dapat melakukan proses menabung sampah pada Aplikasi Bank Sampah.

1.2.3.3 Sequence Diagram Menabung Sampah

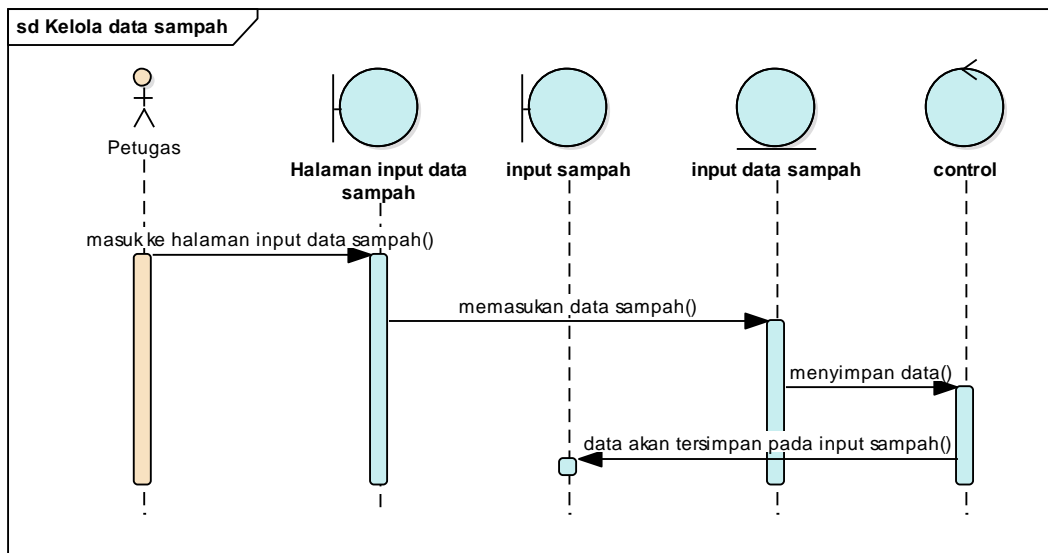


Gambar 3.8 Sequence Diagram Menabung Sampah

Deskripsi Sequence Menabung Sampah :

Pada *Sequence Diagram* ini menjelaskan proses menabung sampah yang dilakukan oleh nasabah, disini nasabah akan mengatur tanggal untuk pengambilan sampah, maka sistem akan menyimpan tanggal tersebut dan akan di tampilkan pada halaman admin di menu jadwal pengambilan sampah.

1.2.3.4 Squence Diagram Kelola Data Sampah

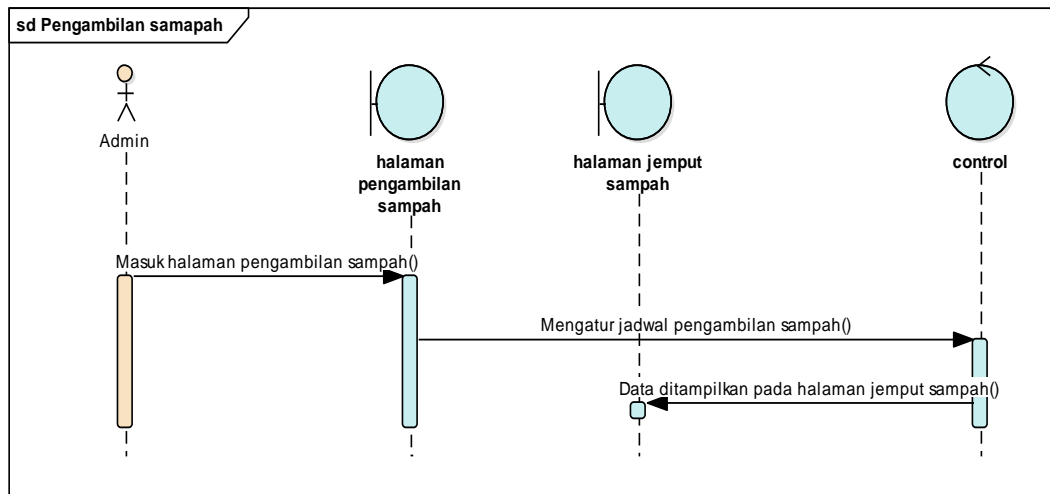


Gambar 3.9 Squence Diagram Kelola Data Sampah

Deskripsi Squence Kelola Data Sampah :

Pada *Sequence Diagram* ini menjelaskan proses penginputan data sampah oleh petugas kebersihan setelah melakukan pengambilan sampah pada nasabah, sistem akan menyimpan data yang telah di inputkan oleh petugas.

1.2.3.5 Squence Diagram Pengambilan Sampah



Gambar 3.10 Sequence Diagram Pengambilan Sampah

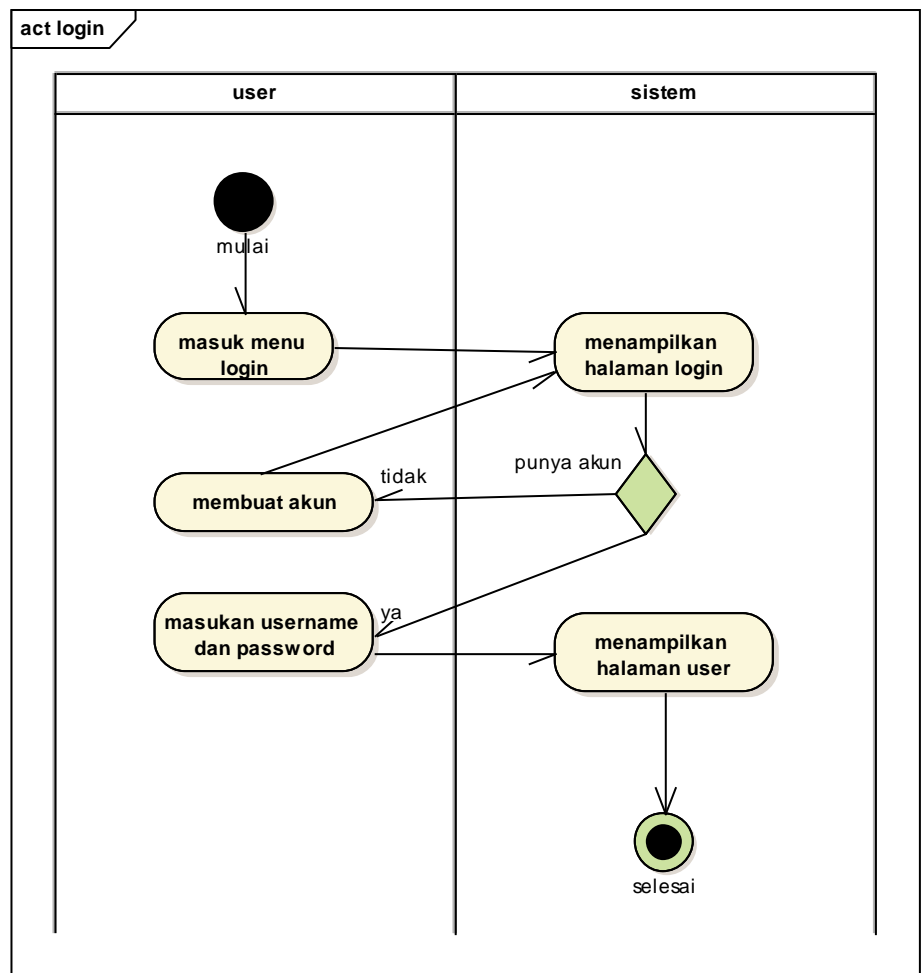
Deskripsi Sequence Diagram Pengambilan Sampah:

Pada *Sequence Diagram* ini menjelaskan proses pengambilan sampah yang akan di atur oleh admin, disini admin akan mengatur petugas mana yang akan melakukan pengambilan sampah pada nasabah. Sistem akan menyimpan data penjemputan sampah dengan status diambil dan belum diambil.

1.2.4 Activity Diagram

Activity Diagram adalah sebuah diagram alur kerja yang menjelaskan berbagai kegiatan pengguna (atau sistem), orang yang melakukan masing-masing aktivitas, dan aliran sekuensial dari aktivitas-aktivitas tersebut.

1.2.4.1 Activity Diagram Login



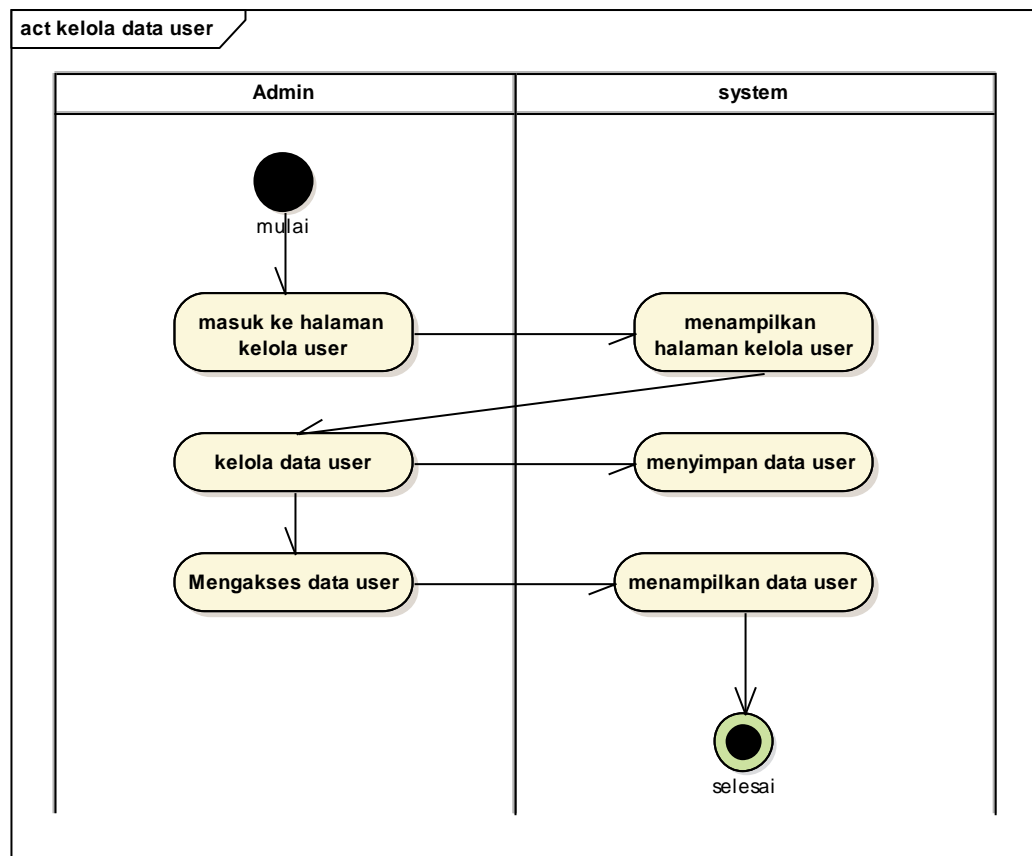
Gambar 3.11 Activity Diagram Kelola Login

Deskripsi Activity Diagram Login:

Pada *Activity Diagram* ini menjelaskan tentang proses login. Dimulai dari admin, pemilik dan peminjam menginput *username* dan *password*, kemudian sistem akan memvalidasi *user*. Jika *valid* maka sistem akan menampilkan menu utama.

Jika tidak *valid*, maka admin, pemilik dan peminjam kembali menginput *username* dan *password*.

1.2.4.2 Activity Diagram Kelola Data User

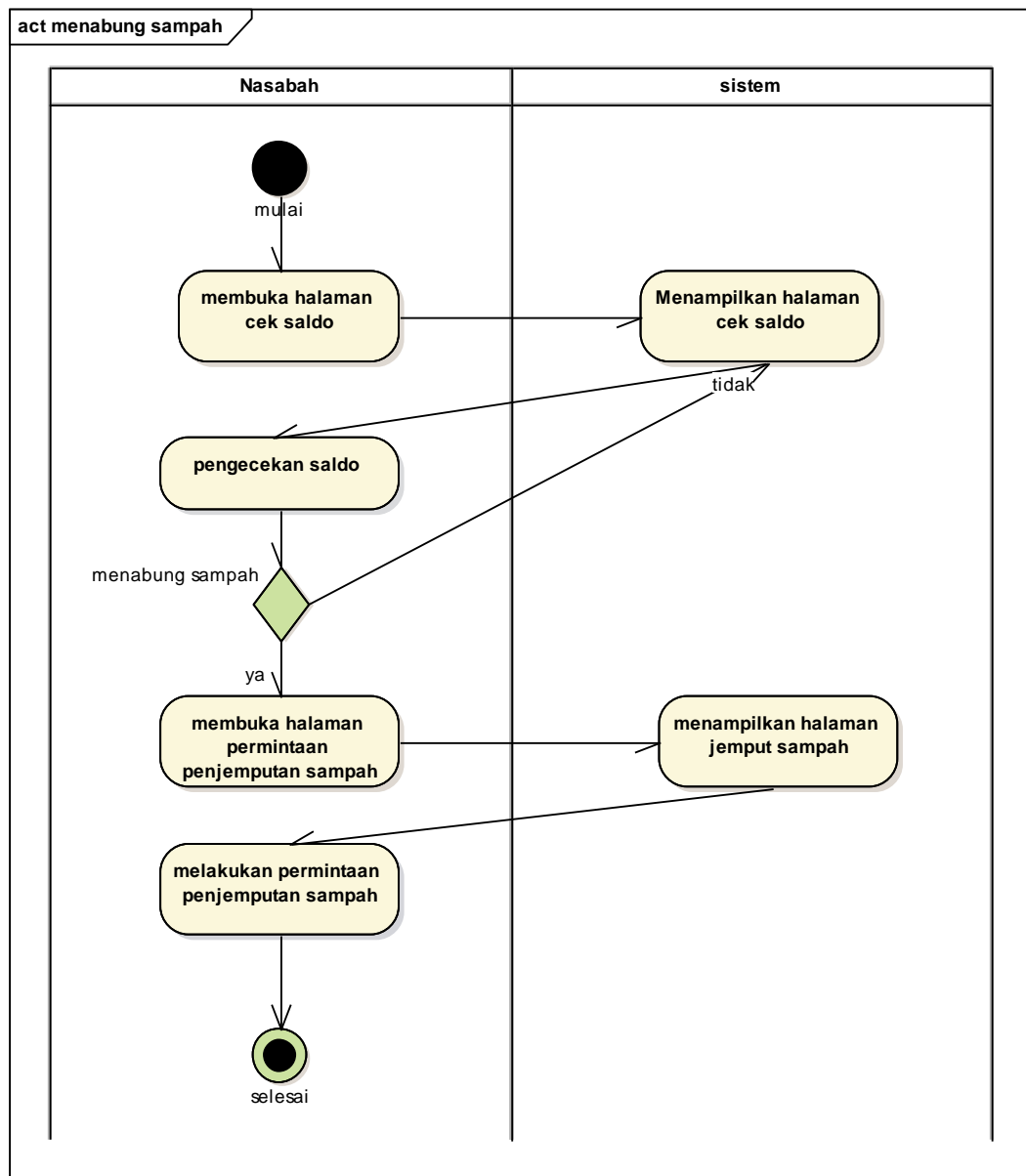


Gambar 3.12 Activity Diagram kelola Data User

Deskripsi Activity Diagram Kelola Data User :

Pada *Activity Diagram* ini menjelaskan admni yang melakukan proses pengelolaan data nasabah memilih menu aktivasi nasabah lalu sistem akan memproses dan memberikan no.rekening kepada nasabah. Selanjutnya nasabah akan mendapatkan no.rekening dan dapat melakukan proses menabung sampah pada Aplikasi Bank Sampah.

1.2.4.3 Activity Diagram Menabung Sampah

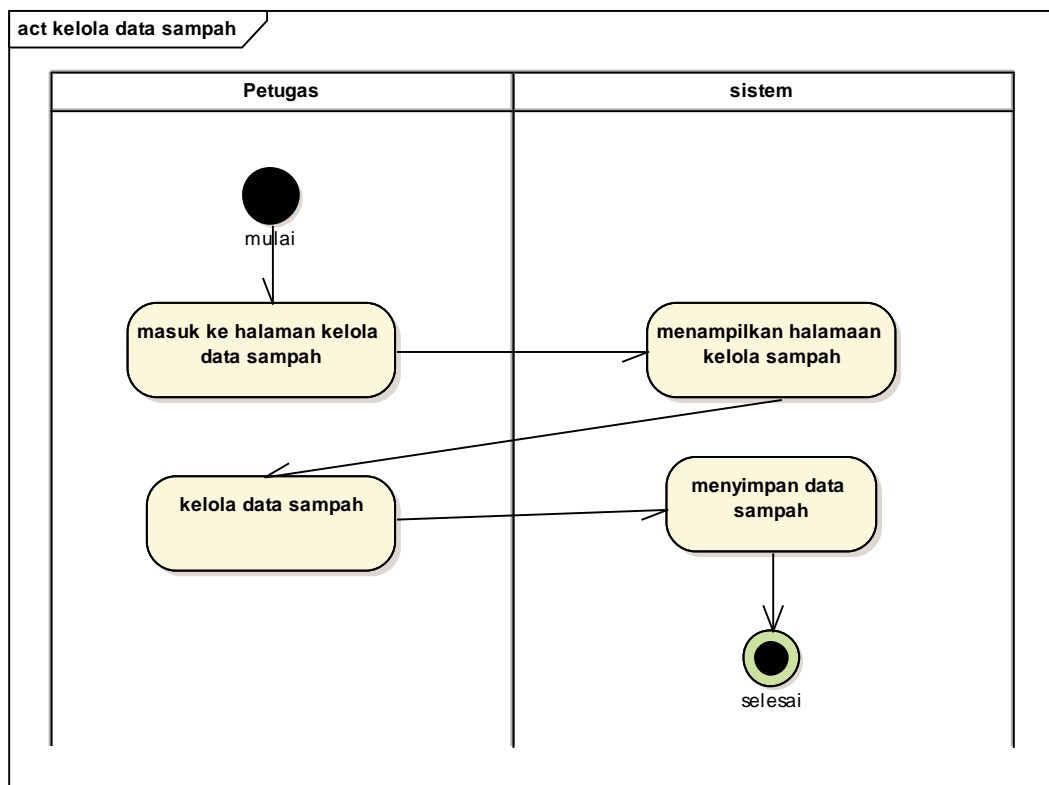


Gambar 3.13 Activity Diagram Menabung Sampah

Deskripsi Activity Diagram Menabung Sampah:

Pada *Activity Diagram* ini menjelaskan proses menabung sampah yang dilakukan oleh nasabah, disini nasabah akan mengatur tanggal untuk pengambilan sampah, maka sistem akan menyimpan tanggal tersebut dan akan di tampilkan pada halaman admin di menu jadwal pengambilan sampah.

1.2.4.4 Activity Diagram Kelola Data Sampah

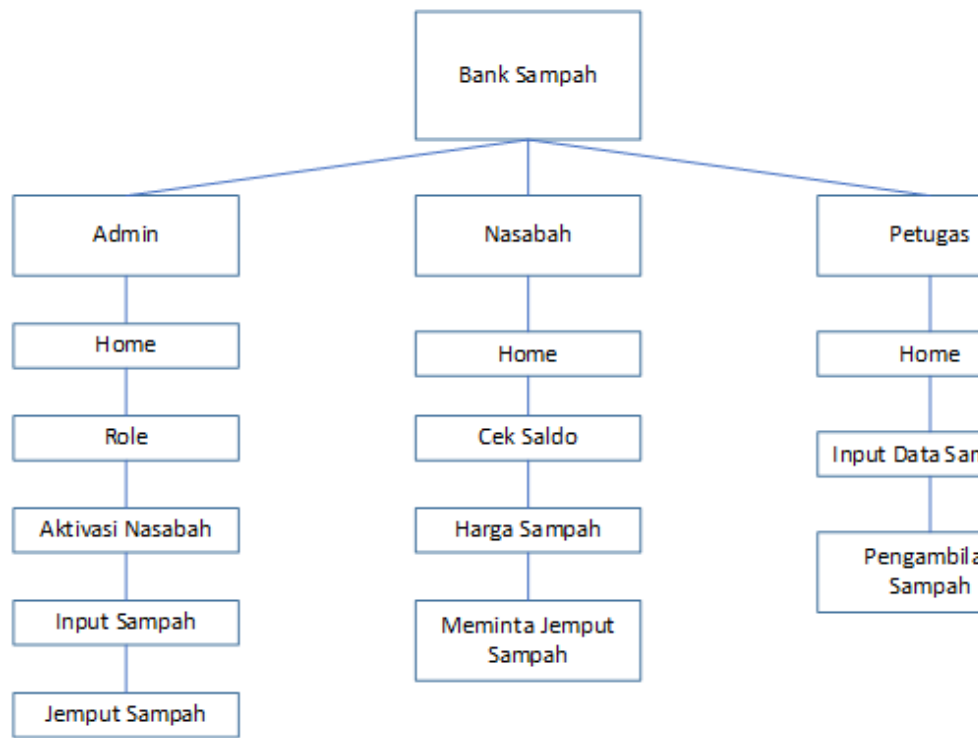


Gambar 3.14 Activity Diagram kelola Data Sampah

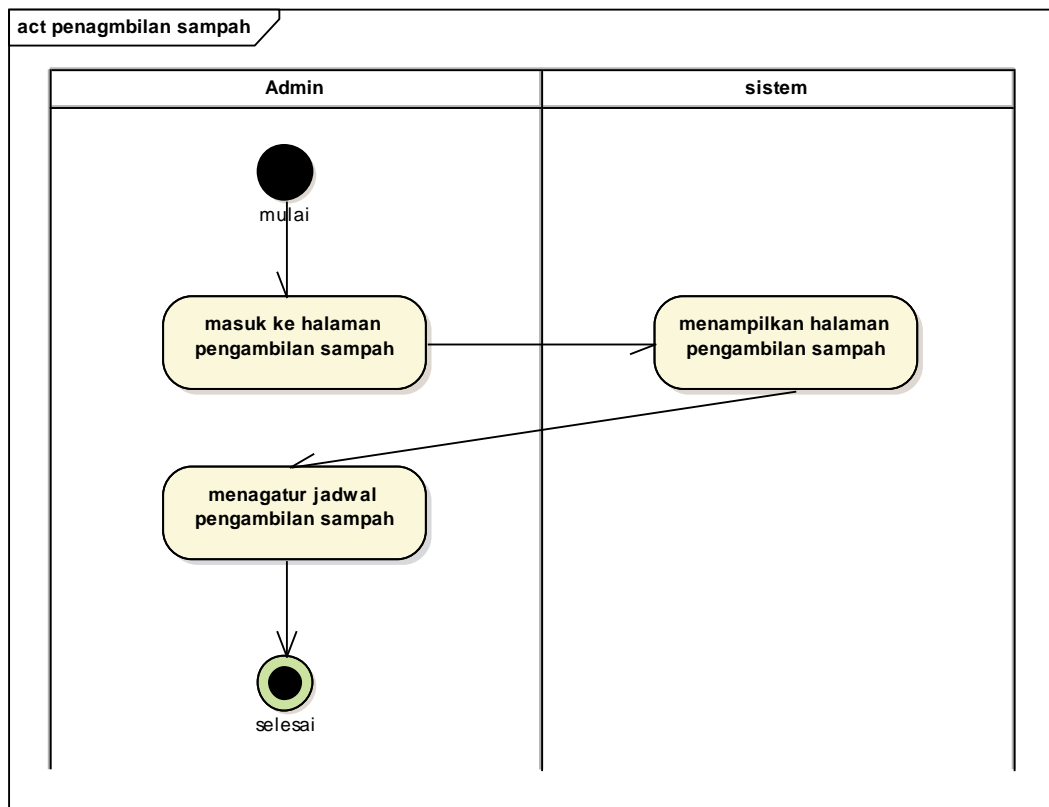
Deskripsi Activity Diagram Kelola Data Sampah:

Pada *Activity Diagram* ini menjelaskan proses penginputan data sampah oleh petugas kebersihan setelah melakukan

pengambilan sampah pada nasabah, sistem akan menyimpan data yang telah di inputkan oleh petugas.



1.2.4.5 Activity Diagram Pengambilan Sampah

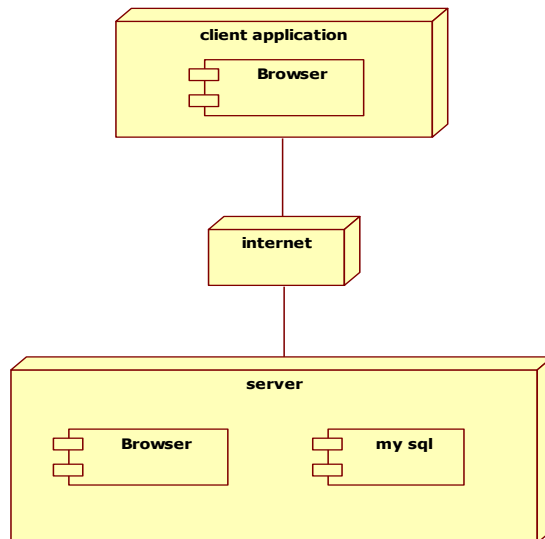


Gambar 3.15 Activity Diagram Pengambilan Sampah

Deskripsi Activity Diagram Pengambilan Sampah:

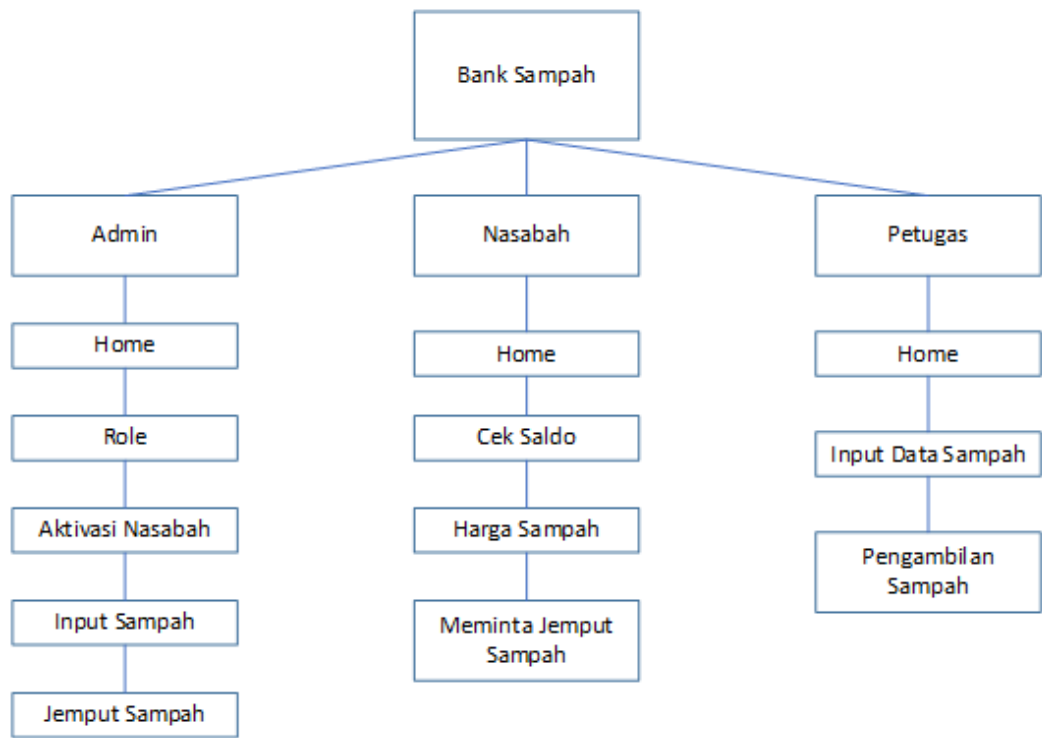
Pada *Sequence Diagram* ini menjelaskan proses pengambilan sampah yang akan di atur oleh admin, disini admin akan mengatur petugas mana yang akan melakukan pengambilan sampah pada nasabah. Sistem akan menyimpan data penjemputan sampah dengan status diambil dan belum diambil.

1.2.5 *Deployment Diagram*



Gambar 3.16 Deployment Diagram

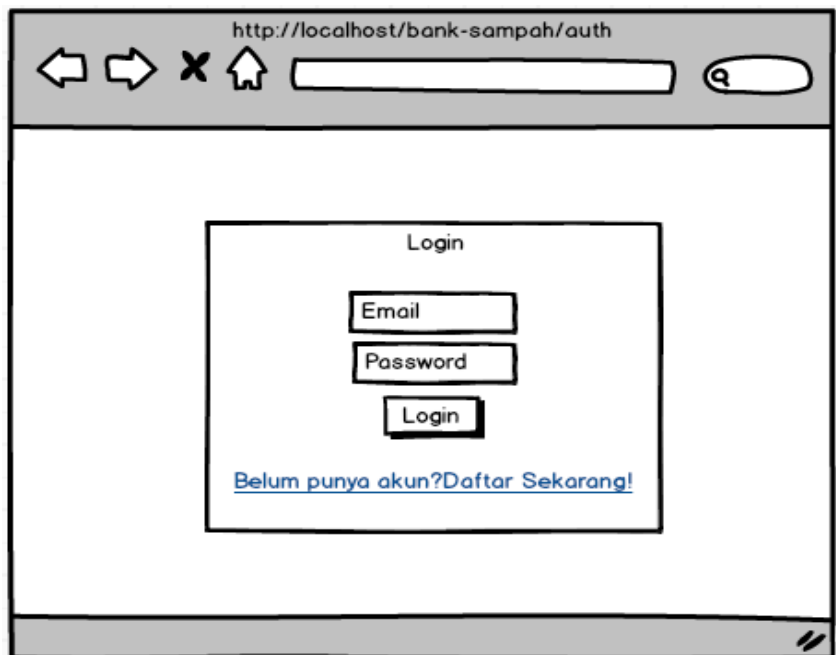
1.2.6 **Struktur Diagram Menu**



Gambar 3.17 Struktur Diagram Menu

1.2.7 Perancangan Antar Muka

1.2.7.1 *User Interface Login*



Gambar 3.18 User Interface Login

1.2.7.2 *User Interface Registrasi*

The image shows a web browser window with the address bar displaying `http://localhost/bank-sampah/auth/registration`. The browser's navigation bar includes back, forward, stop, and home buttons, along with a search bar. The main content area features a registration form titled "Daftar Akun Sebagai :". This form contains two radio buttons for "Nasabah" and "Pegawai", followed by input fields for "Nama Lengkap", "Alamat", and "Alamat email". There are also input fields for "Password" and "Ulangi Pas", and a "Daftar" button at the bottom. A link labeled "Sudah punya akun?Silahkan Login!" is positioned below the form. The browser's status bar at the bottom right shows a double-slash icon.

http://localhost/bank-sampah/auth/registration

Daftar Akun Sebagai :

☐ Nasabah ☐ Pegawai

Nama Lengkap

Alamat

Alamat email

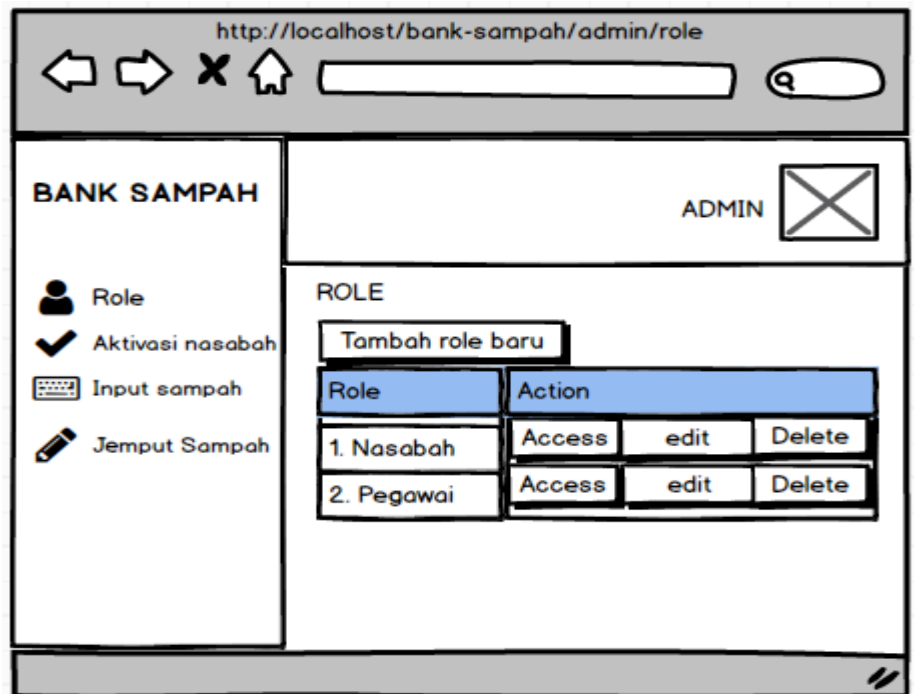
Password Ulangi Pas

Daftar

[Sudah punya akun?Silahkan Login!](#)

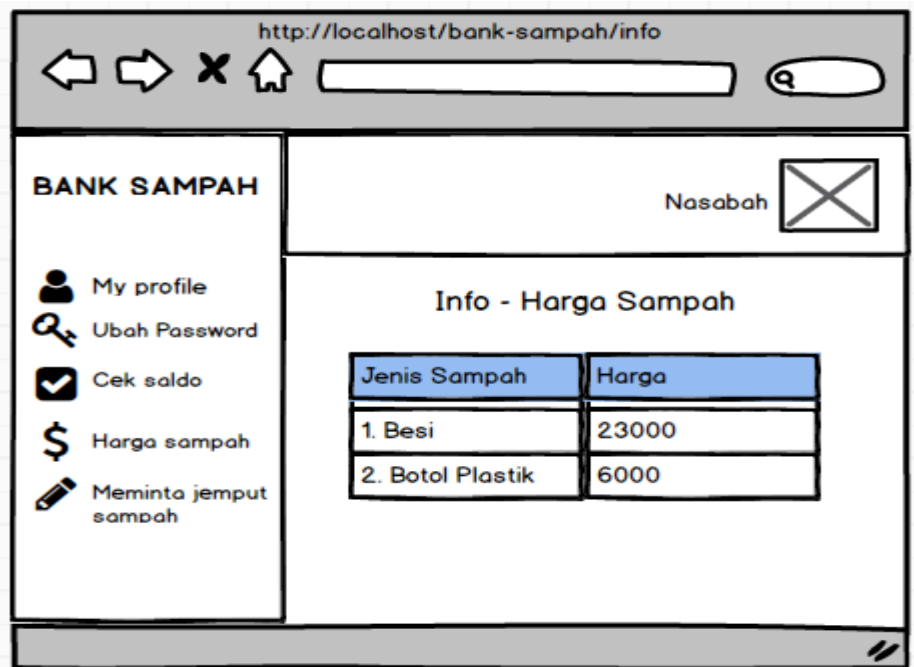
Gambar 3.19 User Interface Registrasi

1.2.7.3 User Interface Kelola User



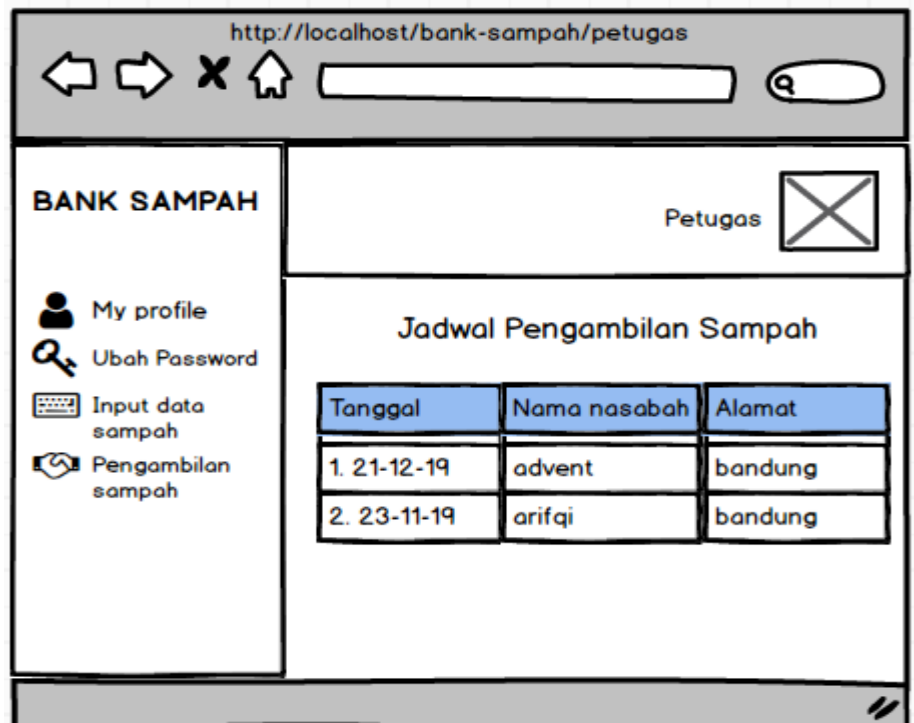
Gambar 3.20 User Interface Kelola User

1.2.7.4 User Interface Harga Sampah



Gambar 3.21 User Interface Harga Sampah

1.2.7.5 User Interface Jadwal Pengambilan Sampah



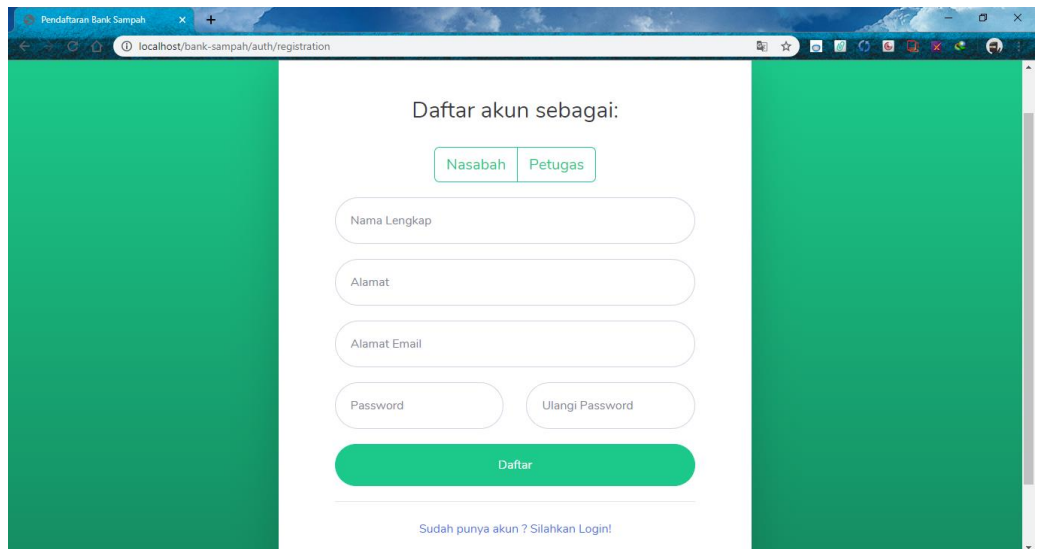
Gambar 3.22 User Interface Pengambilan Sampah

1.3 Pembahasan Hasil Implementasi

1.3.1 Tampilan Antarmuka

Berdasarkan perancangan yang telah dibuat untuk desain antar muka pada bab sebelumnya, didapatkan hasil yang sejalan. Berikut hasil cuplikan halaman-halaman antar muka dari aplikasi yang dibangun :

1.3.1.1 Tampilan Page Registrasi



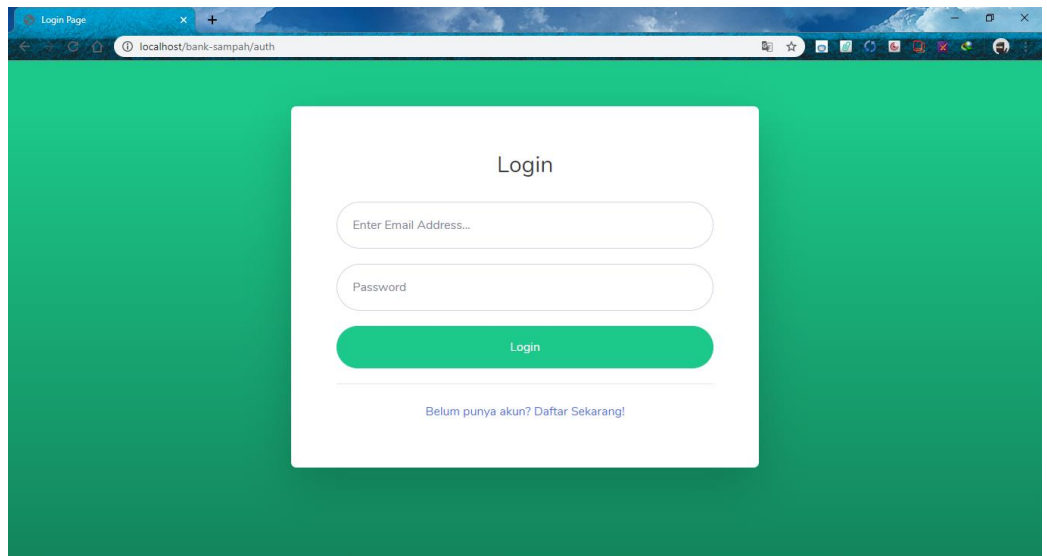
Gambar 4.1 Tampil Page Registrasi

Petunjuk Pemakaian :

1. Input data registrasi dengan benar, kemudian pilih daftar
2. Apabila registrasi berhasil, maka akan masuk ke halaman login.
3. Ada 2 pilihan untuk register, yaitu menjadi nasabah atau petugas

1.3.1.2 Tampilan Page Login Pada Admin

b. Tampilan Page Login Admin

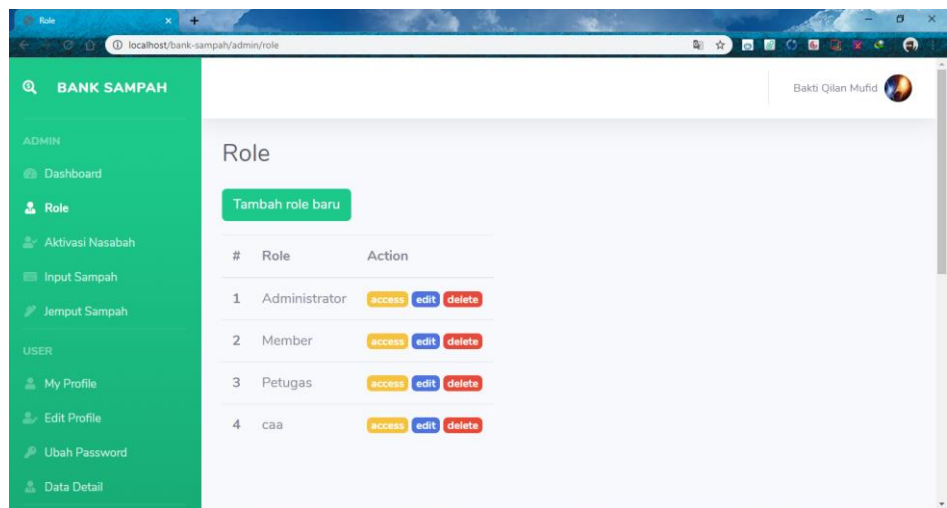


Gambar 4.2 Tampilan Page Login

Petunjuk Pemakaian :

1. Isikan *Username* dan *Password* kemudian klik *Login*.
2. Apabila proses login berhasil maka akan masuk ke *form* halaman utama yang sesuai dengan hak akses yang dimiliki.

c. Tampilan Page Role

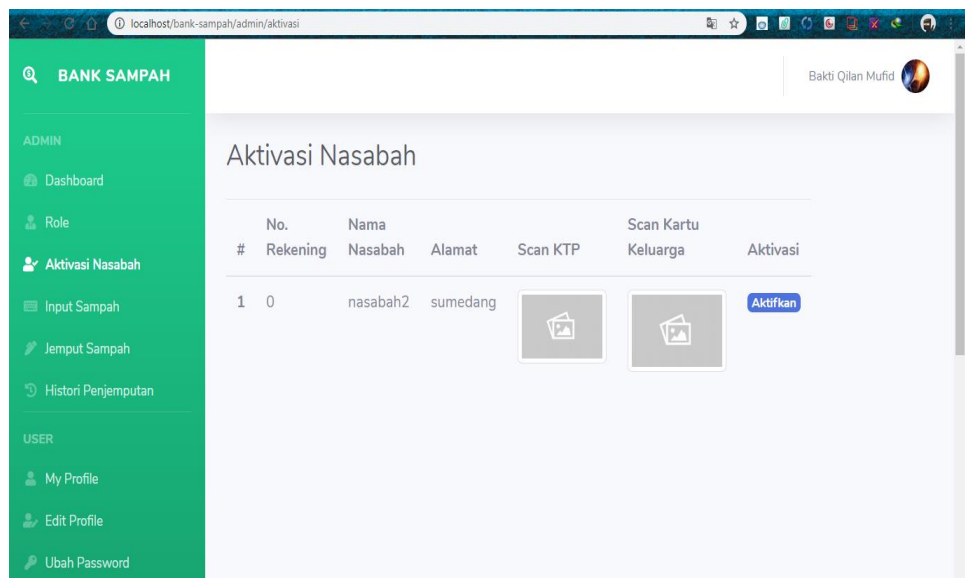


Gambar 4.3 Tampilan Page Role

Petunjuk Pemakaian :

3. Admin mengelola data role
4. Admin dapat melakukan tambah, access, edit dan delete pada role.

d. Tampilan Page Aktivasi Nasabah

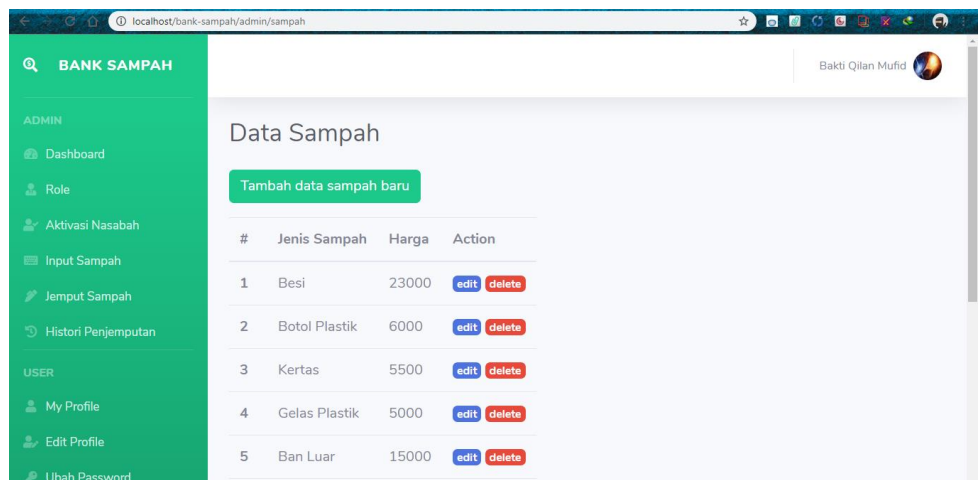


Gambar 4.4 Tampilan Page Aktivasi Nasabah

Petunjuk Pemakaian :

1. Admin akan melakukan proses aktivasi bila nasabah telah menyelesaikan registrasi akun.
2. Apabila proses aktivasi berhasil maka nasabah akan secara otomatis mendapatkan no.rekening.

e. Tampilan Page Input Sampah



Gambar 4.5 Tampilan Page Input Sampah

Petunjuk Pemakaian :

1. Admin akan melakukan proses tambah, edit dan delete data sampah.
2. Apabila proses tambah berhasil maka akan membuat data sampah yang baru
3. Apabila proses edit berhasil maka akan mengubah data sampah
4. Apabila proses delete berhasil maka akan menghapus data sampah

f. Tampilan Page Jemput Sampah



Permintaan Jemput Sampah dari Nasabah

#	Nama Nasabah	Alamat	Tanggal Jemput	Status	Atur Petugas
1	Nasabah pertama	jl. sarimanah 45	2019-12-31	Belum diambil	atur
2	Nasabah pertama	jl. sarimanah 45	2020-01-17	Belum diambil	atur

Gambar 4.6 Tampilan Page Jemput Sampah

Petunjuk Pemakaian :

1. Admin akan mengatur petugas sampah dalam melakukan pengambilan sampah pada nasabah, sebelumnya nasabah telah menetapkan tanggal pengambilan sampahnya.
2. Apabila proses atur berhasil maka admin akan memilih petugas ke berapa yang akan melakukan pengambilan sampah tersebut, sehingga akan masuk ke halaman pengambilan sampah ke petugas yang telah ditentukan.

g. Tampilan Page Histori Penjemputan

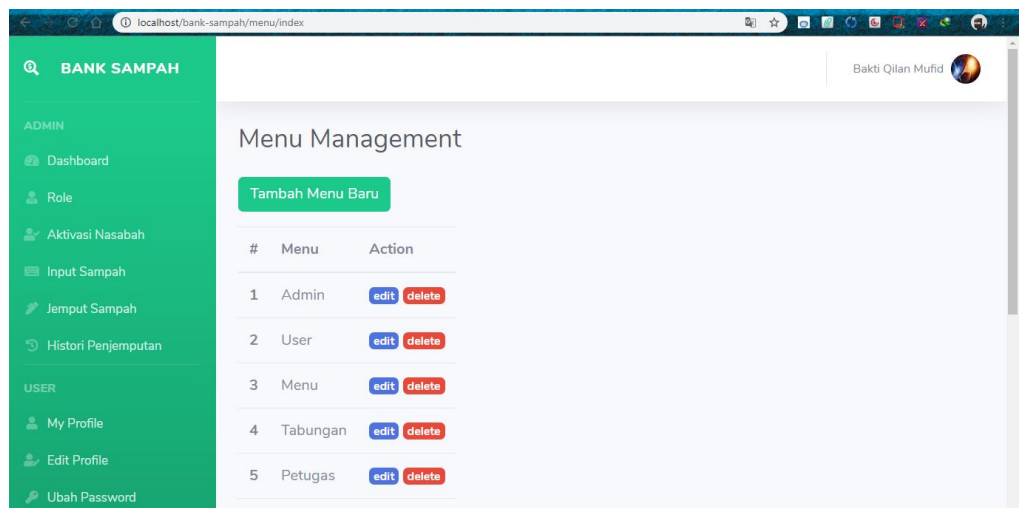
#	Nama Nasabah	Alamat	Tanggal Jemput	Status
1	Nasabah pertama	Jl. sarimamah 45	2019-12-28	diambil
2	Nasabah kedua	Jl. cijerokaso 50	2019-12-31	diambil
3	Nasabah kedua	Jl. cijerokaso 50	2019-12-31	diambil
4	Nasabah kedua	Jl. cijerokaso 50	2019-12-26	diambil
5	nasabah1	bandung	2020-01-09	diambil

Gambar 4.7 Tampilan Page Histori Penjemputan Sampah

Petunjuk Pemakaian :

1. Admin akan melihat dan mengecek menu histori penjemputan sampah .
2. Jika sampah pada nasabah telah di ambil oleh petugas maka status pada histori penjemputan sampah diambil.

h. Tampilan Page Menu Management

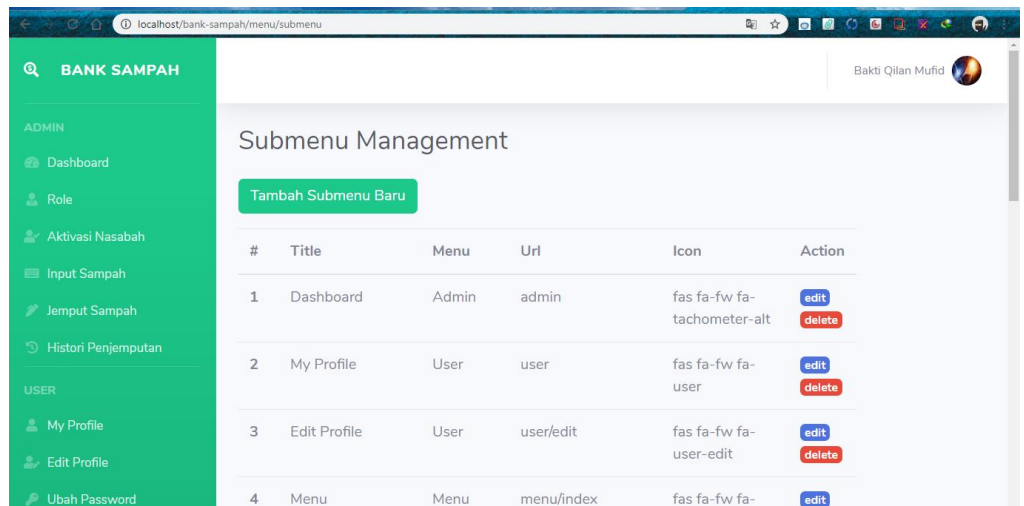


Gambar 4.8 Tampilan Page Menu Management

Petunjuk Pemakaian :

1. Admin akan melakukan proses tambah, edit dan delete pada menu management.
2. Apabila proses tambah berhasil maka akan membuat menu management yang baru
3. Apabila proses edit berhasil maka akan mengubah menu management.
4. Apabila proses delete berhasil maka akan menghapus menu management.

i. Tampilan Page Sub Menu Management

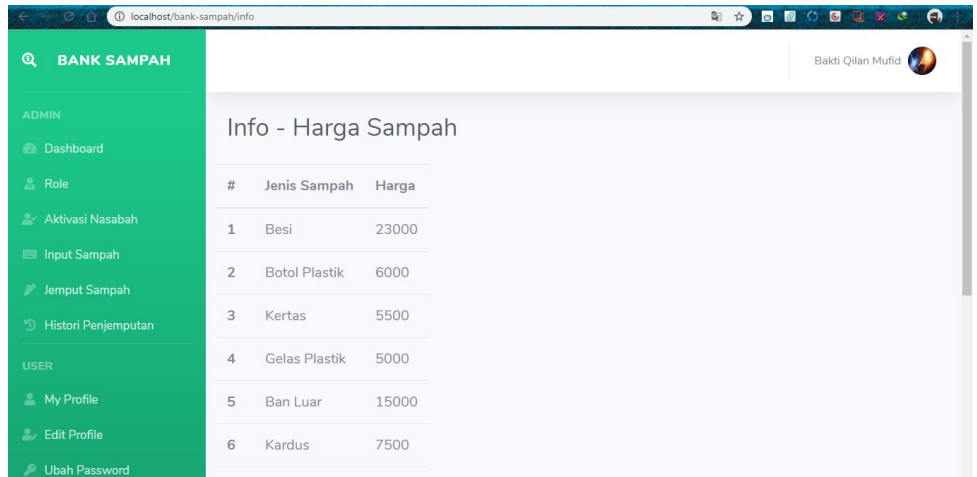


Gambar 4.9 Tampilan Page Sub Menu Management

Petunjuk Pemakaian :

1. Admin akan melakukan proses tambah, edit dan delete pada sub menu management.
2. Apabila proses tambah berhasil maka akan membuat sub menu management yang baru
3. Apabila proses edit berhasil maka akan mengubah sub menu management.
4. Apabila proses delete berhasil maka akan menghapus sub menu management.

j. Tampilan Page Harga Sampah



#	Jenis Sampah	Harga
1	Besi	23000
2	Botol Plastik	6000
3	Kertas	5500
4	Gelas Plastik	5000
5	Ban Luar	15000
6	Kardus	7500

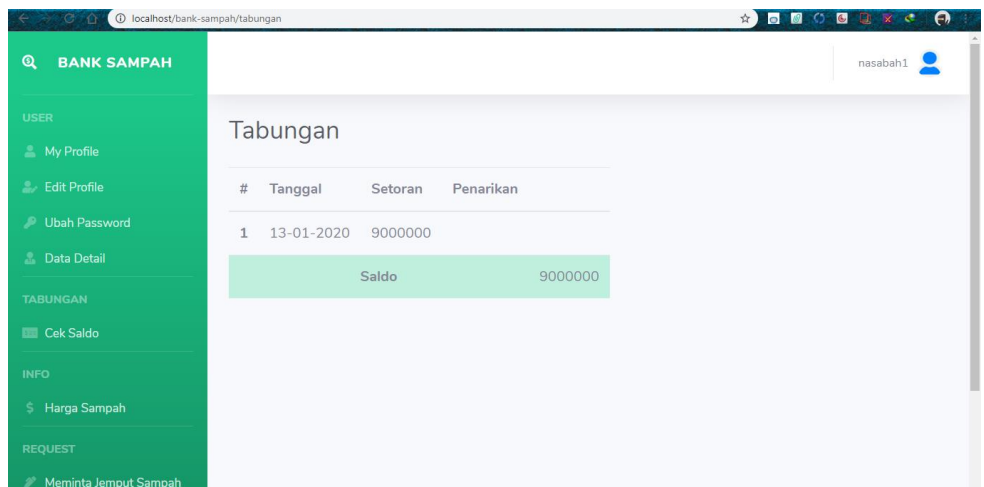
Gambar 4.10 Tampilan Page Harga Sampah

Petunjuk Pemakaian :

1. Admin melihat dan mengecek menu info harga sampah.

1.3.1.3 Tampilan Page Login pada Nasabah

a. Tampilan Page Cek Saldo



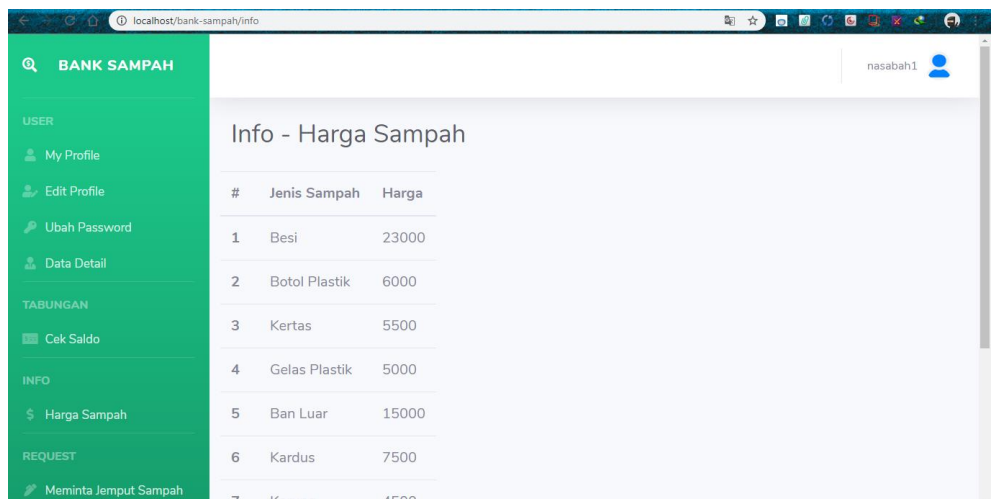
#	Tanggal	Setoran	Penarikan
1	13-01-2020	9000000	
Saldo		9000000	

Gambar 4.11 Tampilan Page Cek Saldo

Petunjuk Pemakaian :

1. Ketika nasabah sudah di aktivasi oleh admin maka nasabah dapat melihat saldo sekaligus mendapatkan no.rekening.
2. Apabila nasabah sudah mendapatkan no rekening maka nasabah dapat melakukan proses menabung sampah.

b. Tampilan Page Harga Sampah



The screenshot shows a web application titled 'BANK SAMPAH'. The left sidebar is green and contains the following menu items: USER (My Profile, Edit Profile, Ubah Password, Data Detail), TABUNGAN (Cek Saldo), INFO (Harga Sampah), and REQUEST (Meminta Jemput Sampah). The main content area is titled 'Info - Harga Sampah' and displays a table with 7 rows of waste types and their prices.

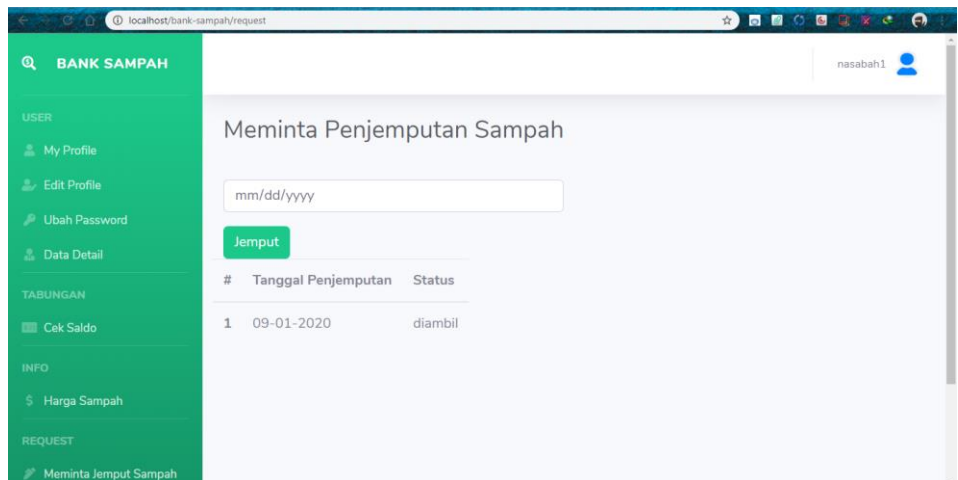
#	Jenis Sampah	Harga
1	Besi	23000
2	Botol Plastik	6000
3	Kertas	5500
4	Gelas Plastik	5000
5	Ban Luar	15000
6	Kardus	7500
7	Karung	4500

Gambar 4.12 Tampilan Page Harga Sampah

Petunjuk Pemakaian :

1. Selain Admin dan petugas, nasabah juga dapat melihat dan mengecek info harga sampah

c. Tampilan Page Meminta Jemput Sampah



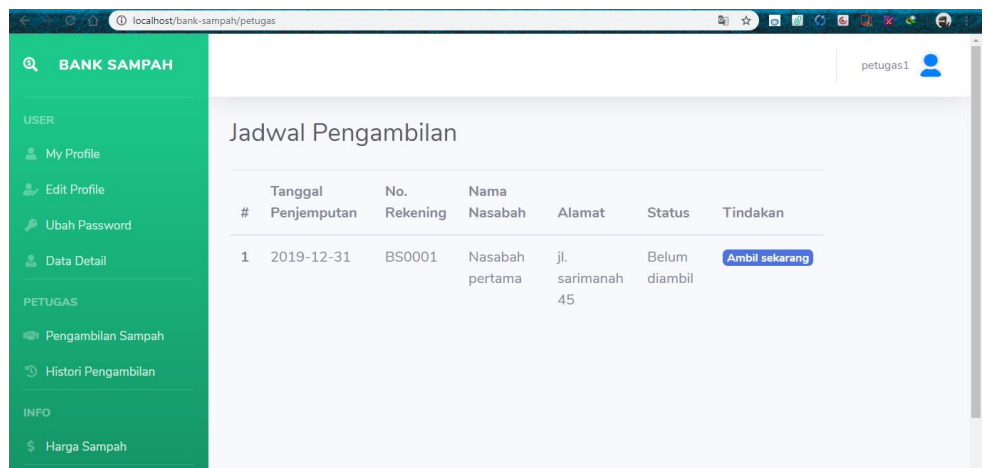
Gambar 4.13 Tampilan Page Meminta Penjemputan Sampah

Petunjuk Pemakaian :

1. Nasabah mengatur tanggal penjemputan sampah maka tanggal tersebut akan tersimpan pada halaman jemput sampah di page admin.
2. Jika sampah pada nasabah telah di ambil oleh petugas maka statusnya akan berubah menjadi diambil.

1.3.1.4 Tampilan Page Login Petugas

a. Tampilan Page Pengambilan Sampah

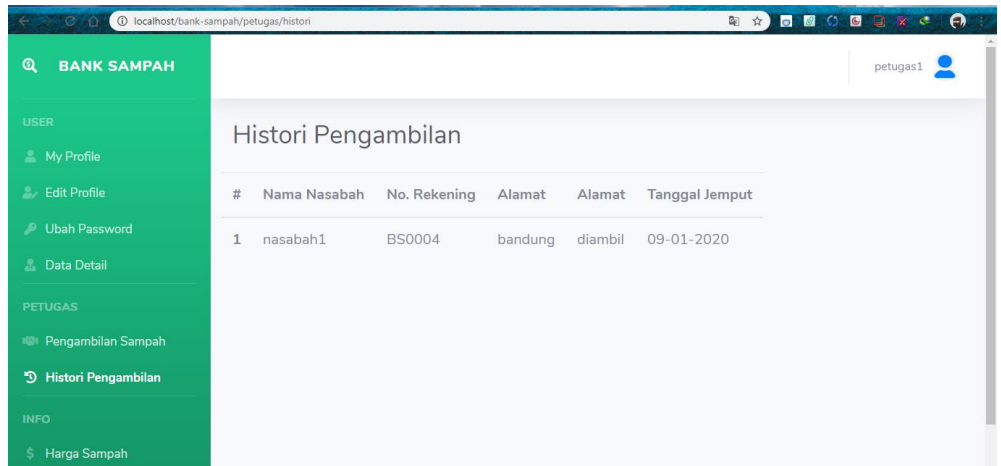


Gambar 4.14 Tampilan Page Pengambilan Sampah

Petunjuk Pemakaian :

1. Ketika Admin mengatur petugas mana yang akan melakukan pengambilan sampah jika petugas 1 yang dipilih untuk melakukan pengambilan sampah maka akan muncul pada halaman pengambilan sampah pada akun petugas 1.
2. Apabila proses pengambilan sampah berhasil maka petugas akan melakukan input data-data sampah yang akan diambilnya.

b. Tampilan Page Histori Pengambilan



The screenshot shows a web application interface for 'BANK SAMPAH'. The left sidebar is green and contains navigation links under three categories: USER (My Profile, Edit Profile, Ubah Password, Data Detail), PETUGAS (Pengambilan Sampah, Histori Pengambilan), and INFO (Harga Sampah). The 'Histori Pengambilan' link is selected. The main content area is titled 'Histori Pengambilan' and displays a table with the following data:

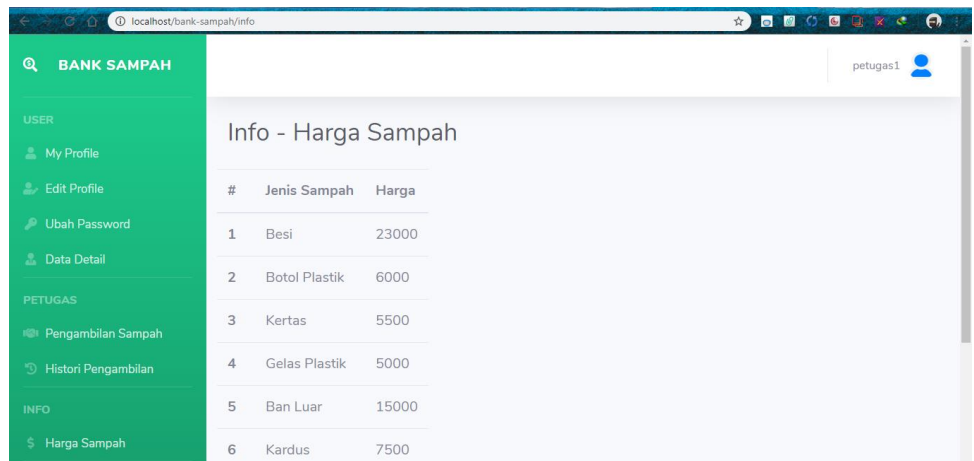
#	Nama Nasabah	No. Rekening	Alamat	Alamat	Tanggal Jemput
1	nasabah1	BS0004	bandung	diambil	09-01-2020

Gambar 4.15 Tampilan Page Histori Pengambilan

Petunjuk Pemakaian :

1. Petugas akan mengecek dan melihat histori pengambilan sampah yang telah petugas tersebut lakukan .
2. Jika proses penginputan berhasil maka akan muncul pada halaman histori pengambilan, bahwa sampah tersebut telah diambil.

c. Tampilan Sampah Harga Sampah



Gambar 4.16 Tampilan Page Harga Sampah

Petunjuk Pemakaian :

1. Selain admin dan nasabah, petugas juga memiliki menu harga sampah .
2. Menu tersebut memudahkan petugas dalam melakukan proses input data sampah.