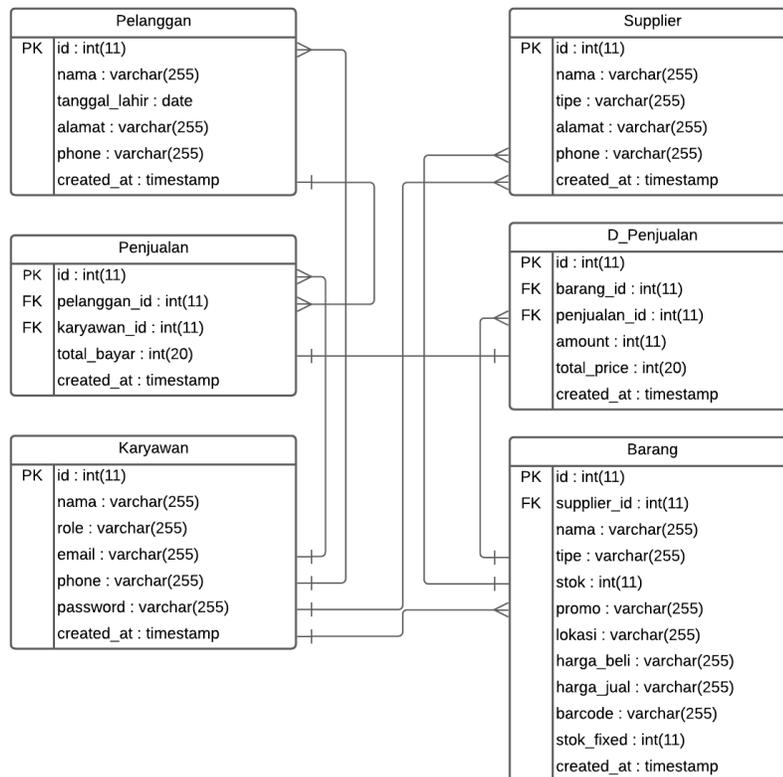


## BAB V HASIL DAN PEMBAHASAN

Berdasarkan hasil kebutuhan yang didapatkan dari tahapan analisis pada Bab IV, kebutuhan yang didapatkan berdasarkan analisis yang didapatkan akan diterapkan dalam perancangan sistem.

### 5.1. *Entity Relationship Diagram (ERD)*



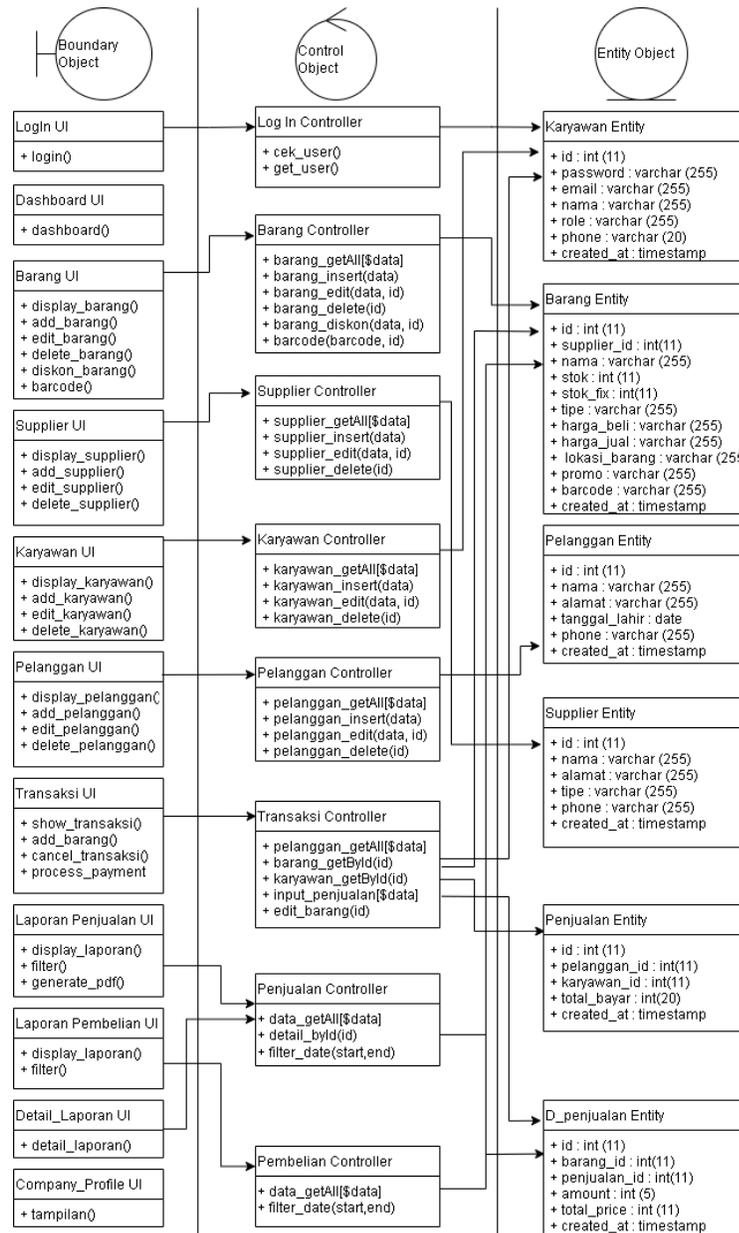
**Gambar 5.1 Entity Relationship Diagram**

*Entity Relationship Diagram* atau ERD yang terdapat pada Gambar 5.1. Digunakan sebagai gambaran struktur data yang diimplementasikan dalam Sistem Informasi Nosa Bike. ERD juga bermanfaat untuk menunjukkan relasi antar data berdasarkan fungsi yang ada.

Berdasarkan Gambar 5.1, dapat dilihat bahwa Tabel User memiliki relasi hampir ke setiap data lainnya kecuali Tabel d\_penjualan, hal ini dikarenakan Tabel User berperan sebagai autentifikasi dimana User Internal harus memasukkan Email dan Password untuk dapat mengakses berbagai informasi yang terdapat di dalam sistem.

Berdasarkan Gambar 5.1, juga dapat dilihat bahwa tabel Penjualan dan D\_Penjualan memiliki FK (*Foreign Key*) yang berperan sebagai data pendukung berupa ID dari tabel lainnya, hal ini digunakan untuk dapat mengakses data pada tabel lainnya.

## 5.2. Class Diagram



Gambar 5.2 Class Diagram

### 5.2.1. *Class Diagram Specific Descriptions*

#### 5.2.1.1. *Specific Design Class Login UI*

Login UI	<<Boundary>>
+ login() Fungsi ini digunakan untuk menampilkan antarmuka autentifikasi untuk User Internal	

#### 5.2.1.2. *Specific Design Class Dashboard UI*

Dashboard UI	<<Boundary>>
+ dashboard() Fungsi ini menampilkan tampilan dashboard ketika fungsi login() berhasil	

#### 5.2.1.3. *Specific Design Class Barang UI*

Barang UI	<<Boundary>>
+ display_barang() Fungsi ini menampilkan semua data barang yang terdapat di sistem	
+ add_barang() Fungsi ini menampilkan form dimana User Internal dapat menambahkan data barang	
+ edit_barang() Fungsi ini menampilkan form dimana User Internal dapat mengubah data barang	
+ delete_barang() Fungsi ini menampilkan konfirmasi untuk menghapus data barang	

<p>+ diskon_barang() Fungsi ini menampilkan form dimana User Internal dapat menambahkan status diskon pada barang</p> <p>+ barcode() Fungsi ini digunakan untuk <i>generate</i> data barang menjadi bentuk barcode</p>
--

#### 5.2.1.4. *Specific Design Class Supplier UI*

Supplier UI	<<Boundary>>
<p>+ display_supplier() Fungsi ini menampilkan semua data supplier yang terdapat di sistem</p> <p>+ add_supplier () Fungsi ini menampilkan form dimana User Internal dapat menambahkan data supplier</p> <p>+ edit_supplier () Fungsi ini menampilkan form dimana User Internal dapat mengubah data supplier</p> <p>+ delete_supplier () Fungsi ini menampilkan konfirmasi untuk menghapus data supplier</p>	

#### 5.2.1.5. *Specific Design Class Karyawan UI*

Karyawan UI	<<Boundary>>
<p>+ display_karyawan() Fungsi ini menampilkan semua data karyawan yang terdapat di sistem</p> <p>+ add_karyawan ()</p>	

Fungsi ini menampilkan form dimana User Internal dapat menambahkan data karyawan

+ edit\_ karyawan ()

Fungsi ini menampilkan form dimana User Internal dapat mengubah data karyawan

+ delete\_ karyawan ()

Fungsi ini menampilkan konfirmasi untuk menghapus data karyawan

#### 5.2.1.6. *Specific Design Class Pelanggan UI*

Pelanggan UI	<<Boundary>>
+ display_pelanggan() Fungsi ini menampilkan semua data pelanggan yang terdapat di sistem	
+ add_ pelanggan () Fungsi ini menampilkan form dimana User Internal dapat menambahkan data pelanggan	
+ edit_ pelanggan () Fungsi ini menampilkan form dimana User Internal dapat mengubah data pelanggan	
+ delete_ pelanggan () Fungsi ini menampilkan konfirmasi untuk menghapus data pelanggan	

#### 5.2.1.7. *Specific Design Class Transaksi UI*

Transaksi UI	<<Boundary>>
+ show_transaksi() Fungsi ini menampilkan tampilan Transaksi	

+ add\_barang()

Fungsi ini menampilkan modal berisi semua data barang yang terdapat di sistem

+ cancel\_transaksi()

Fungsi ini menghapus semua data yang ditambahkan dalam Transaksi

+ process\_payment()

Fungsi ini merupakan fungsi konfirmasi untuk memproses data transaksi

#### 5.2.1.8. *Specific Design Class Laporan Penjualan UI*

Laporan Penjualan UI	<<Boundary>>
+ display_laporan() Fungsi ini menampilkan semua data laporan penjualan	
+ filter() Fungsi ini melakukan sortir data berdasarkan tanggal	
+ generate_pdf() Fungsi ini melakukan <i>generate</i> chart ke bentuk PDF	

#### 5.2.1.9. *Specific Design Class Detail Laporan UI*

Detail_Laporan UI	<<Boundary>>
+ detail_laporan() Fungsi ini menampilkan detail dari data laporan penjualan yang dipilih	

#### 5.2.1.10. *Specific Design Class Laporan Pembelian UI*

Laporan Pembelian UI	<<Boundary>>
+ display_laporan() Fungsi ini menampilkan semua data laporan pembelian	
+ filter() Fungsi ini melakukan sortir data berdasarkan tanggal	

#### 5.2.1.11. *Specific Design Class Company Profile UI*

Company_Profile UI	<<Boundary>>
+ tampilan() Fungsi ini menampilkan tampilan <i>company profile</i>	

#### 5.2.1.12. *Specific Design Class Login Controller*

Login Controller	<<Control>>
+ cek_user() Fungsi ini digunakan untuk menyamakan data yang diinput dengan data karyawan yang ada	
+ get_user() Fungsi ini untuk mengambil dan menyimpan data karyawan sesuai dengan input yang dimasukkan	

#### 5.2.1.13. *Specific Design Class Barang Controller*

Barang Controller	<<Control>>
-------------------	-------------

+ barang_getAll[\$data]	Fungsi ini terjadi ketika User Internal menekan tampilan Barang dimana semua data yang ada di <i>database</i> akan ditampilkan dalam bentuk tabel
+ barang_insert(data)	Fungsi ini digunakan ketika terjadi <i>click</i> pada tombol <i>Add</i> yang akan memproses data yang diinput lalu memasukkan data ke <i>database</i>
+ barang_edit (data, id)	Fungsi ini digunakan ketika terjadi <i>click</i> pada tombol <i>Edit</i> yang akan mengubah data yang diubah lalu memperbaharui data yang ada di <i>database</i>
+ barang_delete (id)	Fungsi ini digunakan ketika terjadi <i>click</i> pada tombol <i>Delete</i> yang akan menghapus data barang yang ada di <i>database</i>
+ barang_diskon (data, id)	Fungsi ini digunakan ketika terjadi <i>click</i> pada tombol "%" yang akan mengubah data harga barang dan status barang yang ada di <i>database</i>
+ barcode (barcode ,id)	Fungsi ini digunakan ketika terjadi <i>click</i> pada tombol <i>barcode</i> yang akan memproses data barang dan mencetak data tersebut dalam bentuk <i>barcode</i> yang dapat dicetak

#### 5.2.1.14. *Specific Design Class Supplier Controller*

Supplier Controller	<<Control>>
+ supplier_getAll[\$data]	Fungsi ini terjadi ketika User Internal menekan tampilan Supplier dimana semua data yang ada di <i>database</i> akan ditampilkan dalam bentuk tabel

+ supplier \_insert(data)

Fungsi ini digunakan ketika terjadi *click* pada tombol *Add* yang akan memproses data yang diinput lalu memasukkan data ke *database*

+ supplier \_edit (data, id)

Fungsi ini digunakan ketika terjadi *click* pada tombol *Edit* yang akan mengubah data yang diubah lalu memperbaharui data yang ada di *database*

+ supplier \_delete (id)

Fungsi ini digunakan ketika terjadi *click* pada tombol *Delete* yang akan menghapus data barang yang ada di *database*

#### 5.2.1.15. *Specific Design Class Karyawan Controller*

Karyawan Controller	<<Control>>
<p>+ karyawan _getAll[\$data]</p> <p>Fungsi ini terjadi ketika User Internal menekan tampilan Karyawan dimana semua data yang ada di <i>database</i> akan ditampilkan dalam bentuk tabel</p> <p>+ karyawan _insert(data)</p> <p>Fungsi ini digunakan ketika terjadi <i>click</i> pada tombol <i>Add</i> yang akan memproses data yang diinput lalu memasukkan data ke <i>database</i></p> <p>+ karyawan _edit (data, id)</p> <p>Fungsi ini digunakan ketika terjadi <i>click</i> pada tombol <i>Edit</i> yang akan mengubah data yang diubah lalu memperbaharui data yang ada di <i>database</i></p> <p>+ karyawan _delete (id)</p> <p>Fungsi ini digunakan ketika terjadi <i>click</i> pada tombol <i>Delete</i> yang akan menghapus data barang yang ada di <i>database</i></p>	

### 5.2.1.16. *Specific Design Class Pelanggan Controller*

Pelanggan Controller	<<Control>>
<p>+ pelanggan_getAll[\$data] Fungsi ini terjadi ketika User Internal menekan tampilan Pelanggan dimana semua data yang ada di <i>database</i> akan ditampilkan dalam bentuk tabel</p> <p>+ pelanggan_insert(data) Fungsi ini digunakan ketika terjadi <i>click</i> pada tombol <i>Add</i> yang akan memproses data yang diinput lalu memasukkan data ke <i>database</i></p> <p>+ pelanggan_edit (data, id) Fungsi ini digunakan ketika terjadi <i>click</i> pada tombol <i>Edit</i> yang akan mengubah data yang diubah lalu memperbaharui data yang ada di <i>database</i></p> <p>+ pelanggan_delete (id) Fungsi ini digunakan ketika terjadi <i>click</i> pada tombol <i>Delete</i> yang akan menghapus data barang yang ada di <i>database</i></p>	

### 5.2.1.17. *Specific Design Class Transaksi Controller*

Transaksi Controller	<<Control>>
<p>+ pelanggan_getAll[\$data] Fungsi ini terjadi ketika user mengakses menu Transaksi, dimana semua data pelanggan akan diambil</p> <p>+ barang_getById(id) Fungsi ini terjadi ketika user menambahkan data barang ke dalam transaksi</p> <p>+karyawan_getById(id)</p>	

Fungsi ini terjadi ketika user menekan tombol process payment dan data karyawan akan diambil

+ input\_penjualan(data)

Fungsi ini digunakan ketika tombol *Process Payment* ditekan dimana data akan diproses dan dimasukkan ke *database*

+ edit\_barang(id)

Fungsi ini terjadi ketika *Process Payment* ditekan dimana data barang yang dimasukkan ke dalam transaksi akan berkurang sesuai dengan jumlah barang yang dibeli

#### 5.2.1.18. *Specific Design Class Penjualan Controller*

Penjualan Controller	<<Control>>
<p>+ data_getAll[\$data]</p> <p>Fungsi ini terjadi ketika user mengakses menu Laporan Penjualan, dimana semua data transaksi akan diambil</p>	
<p>+ detail_byId(id)</p> <p>Fungsi ini terjadi ketika user menekan tombol Show, dimana sistem akan menampilkan detail transaksi data tersebut</p>	
<p>+ filter_date(start,end)</p> <p>Fungsi ini terjadi ketika user melakukan <i>filter</i> data dengan memasukkan tanggal awal dan tanggal akhir</p>	

#### 5.2.1.19. *Specific Design Class Pembelian Controller*

Pembelian Controller	<<Control>>
<p>+ data_getAll[\$data]</p>	

Fungsi ini terjadi ketika user mengakses menu Laporan Pembelian, dimana semua data barang akan diambil

+ filter\_date(start,end)

Fungsi ini terjadi ketika user melakukan *filter* data dengan memasukkan tanggal awal dan tanggal akhir

#### 5.2.1.20. *Specific Design Class Karyawan Entity*

Karyawan Entity	<<Entity>>
<ul style="list-style-type: none"><li>- id : integer (11) Atribut ini menyimpan identitas unik (<i>Primary Key</i>) pada data Karyawan dan bersifat <i>Auto Increment</i></li><li>- Password : varchar (255) Atribut ini menyimpan password dari data Karyawan</li><li>- Email : varchar (255) Atribut ini menyimpan Email dari data Karyawan</li><li>- Nama : varchar (255) Atribut ini menyimpan Nama dari data Karyawan</li><li>- Role : varchar (255) Atribut ini menyimpan Jabatan dari data Karyawan</li><li>- Phone : varchar (20) Atribut ini menyimpan Nomor Handphone dari data Karyawan</li><li>- Created_at : timestamp Atribut ini menyimpan tanggal data dibuat</li></ul>	
+ cek_user()	

Fungsi ini digunakan untuk mencocokkan data input User Internal dan data Karyawan yang terdapat di *database*

+ get\_user()

Fungsi ini digunakan untuk menyimpan data Karyawan yang sudah melalui proses Login

+ karyawan\_getAll[\$data]

Fungsi ini digunakan untuk mengambil semua data Karyawan yang terdapat di dalam *database* menggunakan *query*

+ karyawan\_insert(data)

Fungsi ini digunakan untuk mengambil *input* dari User Internal lalu memasukkan data Karyawan ke dalam *database* menggunakan *query*

+ karyawan\_edit(data, id)

Fungsi ini digunakan untuk mengubah data Karyawan berdasarkan data yang diubah oleh User Internal, atribut karyawan\_id akan digunakan untuk mengetahui data yang akan diperbaharui dan fungsi ini menggunakan *query* untuk memperbaharui data di *database*

+ karyawan\_delete(id)

Fungsi ini digunakan ketika User Internal ingin menghapus data karyawan, atribut karyawan\_id akan digunakan untuk mengetahui data karyawan yang akan dihapus dari *database* menggunakan *query*

#### 5.2.1.21. *Specific Design Class Barang Entity*

Barang Entity	<<Entity>>
<ul style="list-style-type: none"><li>- id : integer (11) Atribut ini menyimpan identitas unik (<i>Primary Key</i>) pada data Barang dan bersifat <i>Auto Increment</i></li> <li>- Supplier_id : integer(11) Atribut ini menyimpan id dari supplier</li></ul>	

- Nama : varchar (255)  
Atribut ini menyimpan Nama dari data Barang
- Stok : integer (11)  
Atribut ini menyimpan Stok dari data Barang
- Stok\_fix : integer(11)  
Atribut ini menyimpan Stok Tetap dari data Barang
- Tipe : varchar (255)  
Atribut ini menyimpan Tipe dari data Barang
- Harga\_beli : varchar (255)  
Atribut ini menyimpan Harga Beli dari data Barang
- Harga\_jual : varchar (255)  
Atribut ini menyimpan Harga Jual dari data Barang
- Lokasi\_barang : varchar (255)  
Atribut ini menyimpan Lokasi dari data Barang
- Promo : varchar (255)  
Atribut ini menyimpan status Promo dari data Barang
- Barcode : varchar (255)  
Atribut ini menyimpan Barcode dari data Barang
- Created\_at : timestamp  
Atribut ini menyimpan tanggal data dibuat

+ barang\_getAll[\$data]

Fungsi ini digunakan untuk mengambil semua data Barang yang terdapat di dalam *database* menggunakan *query*

+ barang\_insert(data)

Fungsi ini digunakan untuk mengambil *input* dari User Internal lalu memasukkan data Barang ke dalam *database* menggunakan *query*

+ barang\_edit(data, id)

Fungsi ini digunakan untuk mengubah data Barang berdasarkan data yang diubah oleh User Internal, atribut barang\_id akan digunakan untuk mengetahui data yang akan diperbaharui dan fungsi ini menggunakan *query* untuk memperbaharui data di *database*

+ barang\_delete(id)

Fungsi ini digunakan ketika User Internal ingin menghapus data barang, atribut barang\_id akan digunakan untuk mengetahui data barang yang akan dihapus dari *database* menggunakan *query*

+ barang\_diskon(data, id)

Fungsi ini digunakan ketika User Internal mengakses menu Diskon, dimana atribut barang\_id akan digunakan untuk mengetahui data barang yang akan diproses, lalu fungsi ini juga mengubah status dan harga dari barang dari *database* menggunakan *query*

+ barcode(barcode, id)

Fungsi ini digunakan ketika User Internal mengakses menu barcode, dimana atribut barcode dan barang\_id akan di-*generate* menjadi sebuah barcode yang dapat dicetak

+ data\_getAll[\$data]

Fungsi ini digunakan ketika User Internal mengakses menu laporan pembelian, dimana semua data barang akan ditampilkan

+ filter\_date(start,end)

Fungsi ini digunakan ketika User Internal melakukan *filter* berdasarkan tanggal awal dan tanggal akhir, dimana data akan ditampilkan berdasarkan jangka waktu yang dipilih

#### 5.2.1.22. *Specific Design Class Supplier Entity*

Supplier Entity	<<Entity>>
<ul style="list-style-type: none"> <li>- id : integer (11) Atribut ini menyimpan identitas unik (<i>Primary Key</i>) pada data Supplier dan bersifat <i>Auto Increment</i></li> <li>- Nama : varchar (255) Atribut ini menyimpan Nama dari data Supplier</li> <li>- Tipe : varchar (255) Atribut ini menyimpan Tipe dari data Supplier</li> <li>- Alamat : varchar (255) Atribut ini menyimpan Alamat dari data Supplier</li> <li>- Phone : varchar (20) Atribut ini menyimpan Nomor Handphone dari data Supplier</li> <li>- Created_at : timestamp Atribut ini menyimpan tanggal data dibuat</li> </ul>	
<p>+ supplier_getAll[\$data]</p> <p>Fungsi ini digunakan untuk mengambil semua data Supplier yang terdapat di dalam <i>database</i> menggunakan <i>query</i></p>	

+ supplier\_insert(data)

Fungsi ini digunakan untuk mengambil *input* dari User Internal lalu memasukkan data Supplier ke dalam *database* menggunakan *query*

+ supplier\_edit(data, id)

Fungsi ini digunakan untuk mengubah data Supplier berdasarkan data yang diubah oleh User Internal, atribut *supplier\_id* akan digunakan untuk mengetahui data yang akan diperbaharui dan fungsi ini menggunakan *query* untuk memperbaharui data di *database*

+ supplier\_delete(id)

Fungsi ini digunakan ketika User Internal ingin menghapus data supplier, atribut *supplier\_id* akan digunakan untuk mengetahui data supplier yang akan dihapus dari *database* menggunakan *query*

#### 5.2.1.23. *Specific Design Class Pelanggan Entity*

Pelanggan Entity	<<Entity>>
<ul style="list-style-type: none"><li>- id : integer (11) Atribut ini menyimpan identitas unik (<i>Primary Key</i>) pada data Pelanggan dan bersifat <i>Auto Increment</i></li> <li>- Nama : varchar (255) Atribut ini menyimpan Nama dari data Pelanggan</li> <li>- Tanggal_lahir : date Atribut ini menyimpan Tanggal Lahir dari data Pelanggan</li> <li>- Alamat : varchar (255) Atribut ini menyimpan Alamat dari data Pelanggan</li> <li>- Phone : varchar (20) Atribut ini menyimpan Nomor Handphone dari data Pelanggan</li></ul>	

- Created\_at : timestamp  
 Atribut ini menyimpan tanggal data dibuat

+ pelanggan\_getAll[\$data]

Fungsi ini digunakan untuk mengambil semua data Pelanggan yang terdapat di dalam *database* menggunakan *query*

+ pelanggan\_insert(data)

Fungsi ini digunakan untuk mengambil *input* dari User Internal lalu memasukkan data Pelanggan ke dalam *database* menggunakan *query*

+ pelanggan\_edit(data, id)

Fungsi ini digunakan untuk mengubah data Pelanggan berdasarkan data yang diubah oleh User Internal, atribut pelanggan\_id akan digunakan untuk mengetahui data yang akan diperbaharui dan fungsi ini menggunakan *query* untuk memperbaharui data di *database*

+ pelanggan\_delete(id)

Fungsi ini digunakan ketika User Internal ingin menghapus data pelanggan, atribut pelanggan\_id akan digunakan untuk mengetahui data supplier yang akan dihapus dari *database* menggunakan *query*

#### 5.2.1.24. Specific Design Class Penjualan Entity

Penjualan Entity	<<Entity>>
<p>- id : integer (11)          Atribut ini menyimpan identitas unik (<i>Primary Key</i>) pada data Penjualan dan bersifat <i>Auto Increment</i></p> <p>- Pelanggan_id : integer(11)          Atribut ini menyimpan id dari pelanggan</p>	

- Karyawan\_id: integer(11)  
Atribut ini menyimpan id dari karyawan
- Total\_bayar : integer(20)  
Atribut ini menyimpan Total Bayar dari data Penjualan
- Created\_at : timestamp  
Atribut ini menyimpan tanggal data dibuat

+ input\_penjualan(data)

Fungsi ini digunakan untuk memasukkan data transaksi penjualan yang terjadi ke dalam *database* menggunakan *query*

+ data\_getAll[\$data]

Fungsi ini digunakan ketika user mengakses menu laporan penjualan dimana semua data transaksi akan ditampilkan

+ filter\_date(start,end)

Fungsi ini digunakan ketika User Internal melakukan *filter* berdasarkan tanggal awal dan tanggal akhir, dimana data akan ditampilkan berdasarkan jangka waktu yang dipilih

+ detail\_byId(id)

Fungsi ini digunakan ketika user mengakses fitur detail penjualan, dimana data penjualan berdasarkan id akan ditampilkan

#### 5.2.1.25. Specific Design Class D\_Penjualan Entity

D_Penjualan Entity	<<Entity>>
<ul style="list-style-type: none"> <li>- id : integer (11) Atribut ini menyimpan identitas unik (<i>Primary Key</i>) pada data D_Penjualan dan bersifat <i>Auto Increment</i></li> </ul>	

- Barang\_id : integer(11)  
Atribut ini menyimpan id dari barang
- Penjualan\_id: integer(11)  
Atribut ini menyimpan id dari penjualan
- Total\_price : integer(11)  
Atribut ini menyimpan Total Price dari data D\_Penjualan
- Amount: integer(11)  
Atribut ini menyimpan Amount dari data D\_Penjualan
- Created\_at : timestamp  
Atribut ini menyimpan tanggal data dibuat

+ input\_penjualan(data)

Fungsi ini digunakan untuk memasukkan data transaksi penjualan yang terjadi ke dalam *database* menggunakan *query*

### 5.3. Pengujian (*Testing*)

Tabel 5.1 merupakan penjelasan mengenai bagaimana hasil dari setiap fungsi yang terdapat dalam Sistem Nosa Bike beserta catatan yang berisikan *feedback* dan fitur tambahan. Berdasarkan setiap fungsi yang terdapat di dalam Sistem Informasi Nosa Bike, dilakukan tahapan *testing* dengan fungsi untuk mengetahui apakah setiap fungsi dapat dijalankan sesuai dengan tujuan dari fungsi tersebut. Berdasarkan *feedback* dari pemilik Nosa Bike, bahwa sistem yang sebelumnya digunakan (Menggunakan Framework CodeIgniter 3 dan Bootstrap) memiliki masalah yaitu *lagging*. Hal ini menyebabkan beberapa fungsi penting seperti *generate barcode* dan fungsi transaksi beroperasi dalam kecepatan yang lambat. Maka dari itu, peneliti memutuskan untuk membuat ulang sistem menggunakan Framework

Laravel 8 dan Tailwind CSS yang mampu beroperasi lebih ringan dan lebih cepat dari sistem yang digunakan sebelumnya.



**Tabel 5.1 Pengujian Fungsi Sistem Informasi Nosa Bike**

No	Fungsi	Tujuan Fungsi	Hasil	Catatan	
1	<i>Log In</i>	Pengguna harus dapat melewati fungsi Autentifikasi dengan memasukkan Email dan Password	Berhasil	<ul style="list-style-type: none"> <li>• Peneliti menambahkan fitur <i>Log In</i> berdasarkan jabatan jika dibutuhkan</li> </ul>	
2	<i>Dashboard</i>	Tampilan <i>Dashboard</i> harus dapat menampilkan beberapa data transaksi bulan ini	Berhasil	-	
3	Pengelolaan Data Barang	<i>Display Data</i> Barang	Tampilan <i>Display</i> Barang harus dapat menampilkan data barang yang sudah ada di <i>database</i>	Berhasil	-
		<i>Add Data</i> Barang	Pengguna harus dapat memasukkan data barang yang diinginkan ke dalam <i>database</i>	Berhasil	<ul style="list-style-type: none"> <li>• Berdasarkan permintaan pemilik Nosa Bike, pengguna dapat menambahkan tipe barang sendiri</li> </ul>
		<i>Edit Data</i> Barang	Pengguna harus dapat mengubah data barang yang diinginkan	Berhasil	-
		<i>Delete Data</i> Barang	Pengguna harus dapat menghapus data barang yang diinginkan	Berhasil	<ul style="list-style-type: none"> <li>• Peneliti menerapkan fitur <i>soft delete</i> untuk mencegah barang terhapus secara tidak sengaja</li> </ul>

No	Fungsi	Tujuan Fungsi	Hasil	Catatan	
		Diskon Data Barang	Pengguna harus dapat menambahkan status diskon dan merubah harga barang yang diinginkan	Berhasil	-
		<i>Generate Data</i> Barang	Pengguna harus dapat <i>generate</i> barcode barang yang diinginkan	Berhasil	<ul style="list-style-type: none"> <li>Berdasarkan kebutuhan, barcode memiliki ukuran 3x8 cm</li> </ul>
4	Penyimpanan Data Karyawan	<i>Display Data</i> Karyawan	Tampilan <i>Display</i> Karyawan harus dapat menampilkan data karyawan yang sudah ada di <i>database</i>	Berhasil	-
		<i>Add Data</i> Karyawan	Pengguna harus dapat memasukkan data karyawan yang diinginkan ke dalam <i>database</i>	Berhasil	-
		<i>Edit Data</i> Karyawan	Pengguna harus dapat mengubah data karyawan yang diinginkan	Berhasil	-
		<i>Delete Data</i> Karyawan	Pengguna harus dapat menghapus data karyawan yang diinginkan	Berhasil	-
5	Penyimpanan Data Supplier	<i>Display Data</i> Supplier	Tampilan <i>Display</i> Supplier harus dapat menampilkan data supplier yang sudah ada di <i>database</i>	Berhasil	-

No	Fungsi	Tujuan Fungsi	Hasil	Catatan	
		<i>Add Data Supplier</i>	Pengguna harus dapat memasukkan data supplier yang diinginkan ke dalam <i>database</i>	Berhasil	-
		<i>Edit Data Supplier</i>	Pengguna harus dapat mengubah data supplier yang diinginkan	Berhasil	-
		<i>Delete Data Supplier</i>	Pengguna harus dapat menghapus data supplier yang diinginkan	Berhasil	-
6	Penyimpanan Data Pelanggan	<i>Display Data Pelanggan</i>	Tampilan <i>Display</i> Pelanggan harus dapat menampilkan data pelanggan yang sudah ada di <i>database</i>	Berhasil	-
		<i>Add Data Pelanggan</i>	Pengguna harus dapat memasukkan data pelanggan yang diinginkan ke dalam <i>database</i>	Berhasil	-
		<i>Edit Data Pelanggan</i>	Pengguna harus dapat mengubah data pelanggan yang diinginkan	Berhasil	-
		<i>Delete Data Pelanggan</i>	Pengguna harus dapat menghapus data pelanggan yang diinginkan	Berhasil	-
7	Transaksi	<i>Add Barang</i>	Pengguna harus dapat menambahkan barang, serta menggunakan alat <i>scanner</i> untuk menambahkan barang yang diinginkan. Pengguna juga harus dapat	Berhasil	<ul style="list-style-type: none"> <li>Berdasarkan kebutuhan, penambahan data barang juga harus dapat</li> </ul>

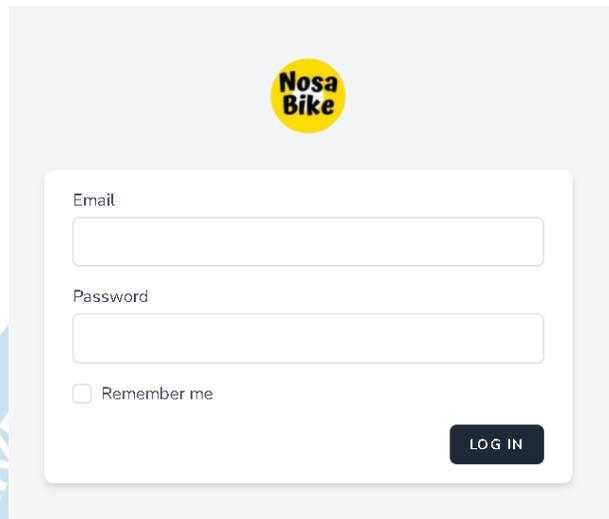
No	Fungsi	Tujuan Fungsi	Hasil	Catatan	
		mengubah harga dan jumlah barang yang dipilih		mengubah harga produk yang dipilih	
	<i>Show</i> Transaksi	Tampilan <i>Show</i> Transaksi harus dapat menampilkan data barang yang sudah melalui fungsi <i>Add</i> Barang	Berhasil	-	
	<i>Cancel</i> Transaksi	Pengguna harus dapat menekan tombol " <i>Cancel</i> " untuk menghapus semua barang yang sudah ditambahkan	Berhasil	-	
	<i>Process Payment</i>	Sistem harus dapat mengolah data yang dimasukkan pengguna ke dalam <i>database</i> ketika pengguna menekan tombol <i>Process Payment</i>	Berhasil	-	
	<i>Edit</i> Stok Barang	Sistem harus dapat mengubah jumlah stok barang di <i>database</i> sesuai dengan barang yang dipilih	Berhasil	-	
8	Laporan Penjualan	<i>Display</i> Laporan Penjualan	Tampilan <i>Display</i> Laporan Penjualan harus dapat menampilkan semua data transaksi	Berhasil	-
	<i>Detail</i> Laporan Penjualan	Sistem harus dapat menampilkan <i>Detail</i> Penjualan yang dipilih oleh pengguna	Berhasil	-	

No	Fungsi	Tujuan Fungsi	Hasil	Catatan	
		<i>Generate PDF Chart</i>	Sistem harus dapat melakukan <i>Generate</i> grafik dalam bentuk PDF ketika pengguna menekan tombol " <i>Generate PDF</i> "	Berhasil	-
		<i>Filter Penjualan</i>	Sistem harus dapat menampilkan data penjualan sesuai dengan tanggal yang dipilih pengguna	Berhasil	-
9	Laporan Pembelian	<i>Display Laporan Pembelian</i>	Tampilan <i>Display Laporan Pembelian</i> harus dapat menampilkan semua pemasukkan data barang	Berhasil	-
		<i>Filter Pembelian</i>	Sistem harus dapat menampilkan data pembelian barang sesuai dengan tanggal yang dipilih pengguna	Berhasil	-
10	<i>Company Profile</i>	<i>Display Company Profile</i>	Tampilan <i>Display Company Profile</i> harus dapat digunakan oleh setiap pengguna dari berbagai perangkat	Berhasil	-

## 5.4. Perancangan Antarmuka

### 5.3.1. Antarmuka *Login* UI

#### 5.3.1.1. Antarmuka *Login*()



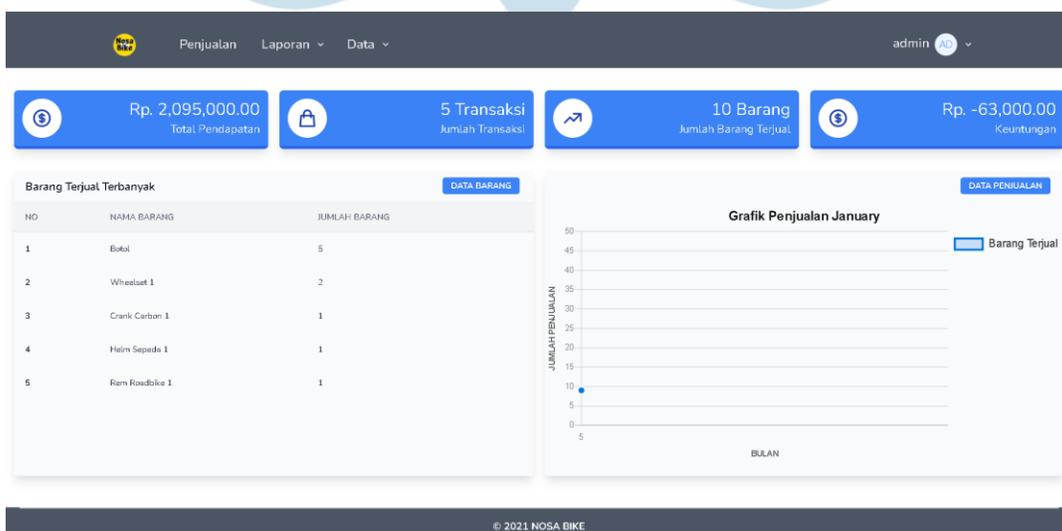
The login interface for Nosa Bike includes a logo at the top center. Below it is a white rectangular form containing the following elements: an 'Email' label above a text input field, a 'Password' label above another text input field, a checkbox labeled 'Remember me', and a dark blue button with the text 'LOG IN' in white.

**Gambar 5.3** *Login* UI

Gambar 5.3 merupakan tampilan halaman *Log In*. Halaman ini merupakan halaman autentifikasi untuk User Internal dapat memasuki sistem Nosa Bike.

### 5.3.2. Antarmuka *Dashboard* UI

#### 5.3.2.1. Antarmuka *Dashboard*()

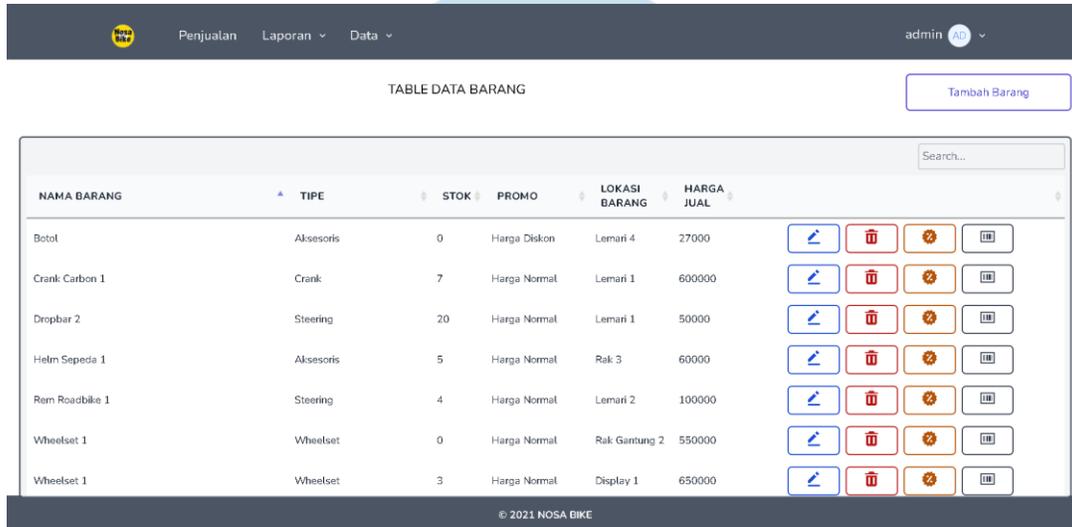


**Gambar 5.4** *Dashboard* UI

Gambar 5.4 merupakan tampilan halaman *Dashboard* . Halaman ini merupakan halaman ketika user melewati proses autentifikasi, dimana user akan ditampilkan ringkasan data penjualan pada bulan sistem diakses.

### 5.3.3. Antarmuka Barang UI

#### 5.3.3.1. Antarmuka *Display\_Barang()*



The screenshot shows a web application interface for displaying goods. At the top, there is a navigation bar with a logo, menu items 'Penjualan', 'Laporan', and 'Data', and a user profile 'admin'. Below the navigation bar, the title 'TABLE DATA BARANG' is centered, with a 'Tambah Barang' button on the right. The main content area features a table with a search bar at the top right. The table has the following columns: NAMA BARANG, TYPE, STOK, PROMO, LOKASI BARANG, and HARGA JUAL. Each row represents an item with its details and a set of action buttons (edit, delete, refresh, and print).

NAMA BARANG	TYPE	STOK	PROMO	LOKASI BARANG	HARGA JUAL	
Botol	Aksesoris	0	Harga Diskon	Lemari 4	27000	[Edit] [Delete] [Refresh] [Print]
Crank Carbon 1	Crank	7	Harga Normal	Lemari 1	600000	[Edit] [Delete] [Refresh] [Print]
Dropbar 2	Steering	20	Harga Normal	Lemari 1	50000	[Edit] [Delete] [Refresh] [Print]
Helm Sepeda 1	Aksesoris	5	Harga Normal	Rak 3	60000	[Edit] [Delete] [Refresh] [Print]
Rem Roadbike 1	Steering	4	Harga Normal	Lemari 2	100000	[Edit] [Delete] [Refresh] [Print]
Wheelset 1	Wheelset	0	Harga Normal	Rak Gantung 2	550000	[Edit] [Delete] [Refresh] [Print]
Wheelset 1	Wheelset	3	Harga Normal	Display 1	650000	[Edit] [Delete] [Refresh] [Print]

© 2021 NOSA BIKE

**Gambar 5.5 *Display* Barang UI**

Gambar 5.5 merupakan tampilan halaman *Display* Barang. Halaman ini merupakan halaman ketika User Internal mengakses Barang, dimana semua data akan ditampilkan dalam bentuk tabel.

### 5.3.3.2. Antarmuka *Add\_Barang()*

The screenshot shows the 'TAMBAH DATA BARANG' form. At the top left is a 'Kembali' button. The form fields are: 'Nama Barang' (empty), 'Pilih Supplier' (dropdown menu showing 'Tidak Ada'), 'Tipe' (empty), 'Stok Barang' (empty), 'Lokasi Barang' (empty), and 'Harga Beli Barang' (empty). The footer contains '© 2021 NOSA BIKE'.

**Gambar 5.6 Add Barang UI**

Gambar 5.6 merupakan halaman *Add Barang*. Halaman ini merupakan tampilan ketika User Internal menekan tombol *Tambah Barang*, pada form ini dapat dilakukan *input* data *Barang* ke dalam *database* dengan memasukkan data *barang* dan menekan tombol *Submit*.

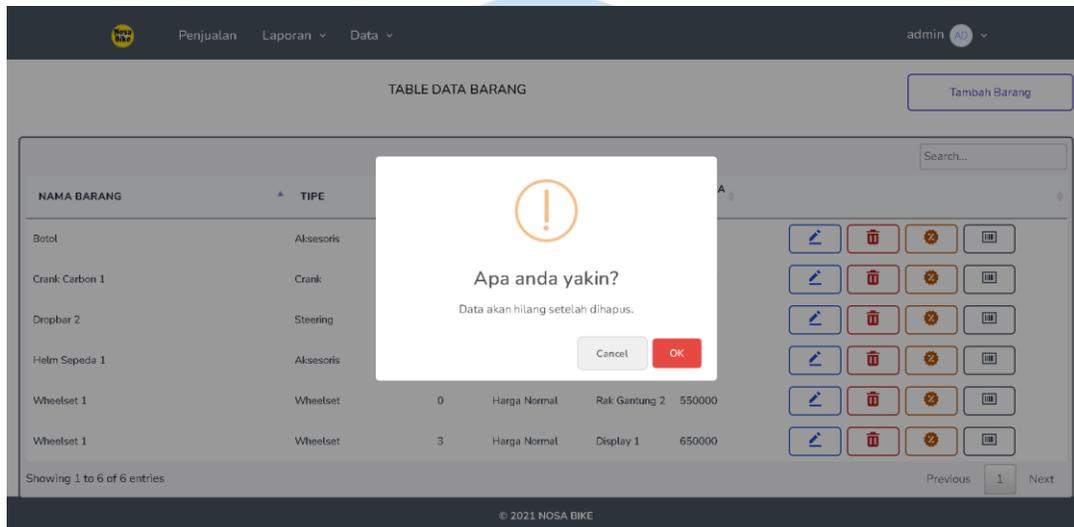
### 5.3.3.3. Antarmuka *Edit\_Barang()*

The screenshot shows the 'EDIT DATA BARANG' form. At the top left is a 'Kembali' button. The form fields are: 'Nama Barang' (Botol), 'Tipe' (Aksesoris), 'Stok Barang' (0), 'Lokasi Barang' (Lemari 4), 'Harga Jual Barang' (27000), and 'Harga Beli Barang' (20000). The footer contains '© 2021 NOSA BIKE'.

**Gambar 5.7 Edit Barang UI**

Gambar 5.7 merupakan halaman *Edit* Barang. Halaman ini merupakan tampilan ketika User Internal menekan tombol dengan *icon edit* (Berwarna Biru), pada form ini dapat dilakukan *edit* data Barang yang terdapat didalam *database* ketika menekan tombol *Update*.

#### 5.3.3.4. Antarmuka *Delete\_Barang()*



**Gambar 5.8 Delete Barang UI**

Gambar 5.8 merupakan halaman *Delete* Barang . Halaman ini merupakan *pop-up* ketika User Internal menekan tombol dengan *icon delete* (Berwarna Merah), pada form ini dapat dilakukan *delete* data Barang yang terdapat didalam *database* ketika menekan tombol *OK*.

### 5.3.3.5. Antarmuka Diskon\_Barang()

Penjualan Laporan Data admin AD

— Kembali

DISKON BARANG

Nama Barang  
Botol

Harga Barang  
27000

Harga Beli Barang  
20000

Masukkan Jumlah Diskon  
0

Harga Setelah Diskon  
0

© 2021 NOSA BIKE

**Gambar 5.9 Diskon Barang UI**

Gambar 5.9 merupakan halaman Diskon Barang. Halaman ini merupakan tampilan ketika User Internal menekan tombol dengan *icon %* (Berwarna Kuning), pada form ini dapat dilakukan diskon data Barang yang terdapat didalam *database*, hal ini dilakukan dengan memasukkan jumlah diskon yang diinginkan, lalu tombol *Simpan* akan tampil setelah user memasukkan jumlah diskon.

### 5.3.3.6. Antarmuka Barcode()

Penjualan Laporan Data admin AD

— Kembali

BARCODE BARANG

Nama Barang  
Botol

Harga Barang  
27000

Botol...  
Rp. 27.000

Cetak Barcode

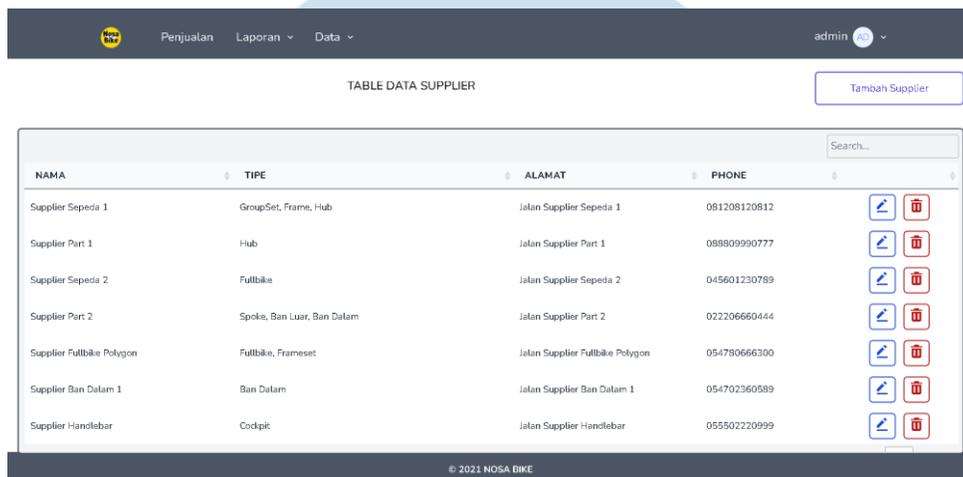
© 2021 NOSA BIKE

**Gambar 5.10 Barcode UI**

Gambar 5.10 merupakan halaman Barcode Barang . Halaman ini merupakan tampilan ketika User Internal menekan tombol dengan *icon barcode* (Berwarna Abu-Abu), pada form ini dapat dilakukan *generate* barcode data Barang yang terdapat didalam *database* ketika menekan tombol *Cetak Barcode*.

### 5.3.4. Antarmuka Supplier UI

#### 5.3.4.1. Antarmuka *Display\_Supplier()*



NAMA	TIPE	ALAMAT	PHONE	
Supplier Sepeda 1	GroupSet, Frame, Hub	Jalan Supplier Sepeda 1	081208120812	 
Supplier Part 1	Hub	Jalan Supplier Part 1	088809990777	 
Supplier Sepeda 2	Fullbike	Jalan Supplier Sepeda 2	045601230789	 
Supplier Part 2	Spoke, Ban Luar, Ban Dalam	Jalan Supplier Part 2	022206660444	 
Supplier Fullbike Polygon	Fullbike, Frameset	Jalan Supplier Fullbike Polygon	054780666300	 
Supplier Ban Dalam 1	Ban Dalam	Jalan Supplier Ban Dalam 1	054702360589	 
Supplier Handlebar	Codipit	Jalan Supplier Handlebar	055502220999	 

**Gambar 5.11 *Display Supplier* UI**

Gambar 5.11 merupakan tampilan halaman *Display Supplier*. Halaman ini merupakan halaman ketika User Internal mengakses Supplier, dimana semua data akan ditampilkan dalam bentuk tabel.

### 5.3.4.2. Antarmuka *Add\_Supplier()*

Penjualan Laporan Data admin

← Kembali TAMBAH DATA SUPPLIER

Nama Supplier:  
...

Alamat:  
...

Tipe:  
...

Phone:  
...

Submit

© 2021 NOSA BIKE

**Gambar 5.12 Add Supplier UI**

Gambar 5.12 merupakan halaman *Add Supplier*. Halaman ini merupakan tampilan ketika User Internal menekan tombol Tambah Supplier, pada form ini dapat dilakukan *input* data Supplier ke dalam *database* dengan memasukkan data barang dan menekan tombol *Submit*.

### 5.3.4.3. Antarmuka *Edit\_Supplier()*

Penjualan Laporan Data admin

← Kembali EDIT DATA SUPPLIER

Nama Supplier :  
Supplier Sepeda 1

Alamat Supplier :  
Jalan Supplier Sepeda 1

Tipe :  
GroupSet, Frame, Hub

Nomor HP Supplier :  
081208120812

Update

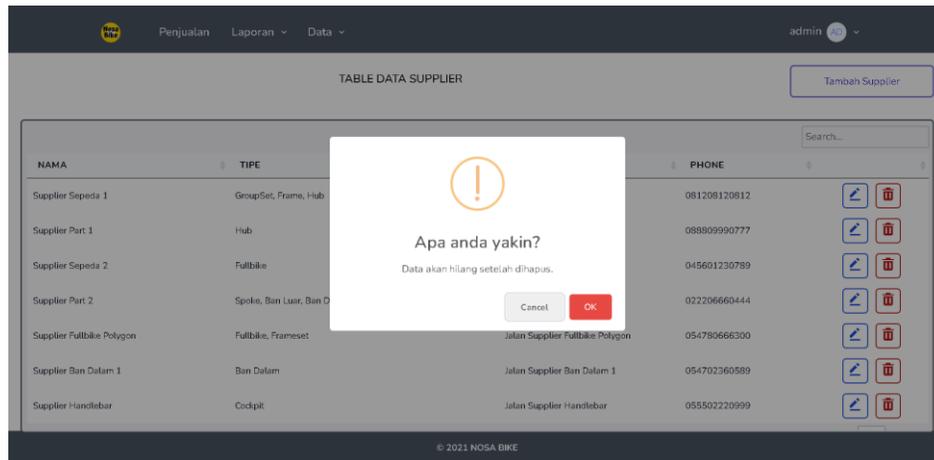
© 2021 NOSA BIKE

**Gambar 5.13 Edit Supplier UI**

Gambar 5.13 merupakan halaman *Edit Supplier*. Halaman ini merupakan tampilan ketika User Internal menekan tombol dengan *icon edit* (Berwarna Biru), pada

form ini dapat dilakukan *edit* data Supplier yang terdapat didalam *database* ketika menekan tombol *Update*.

### 5.3.4.4. Antarmuka *Delete\_Supplier()*

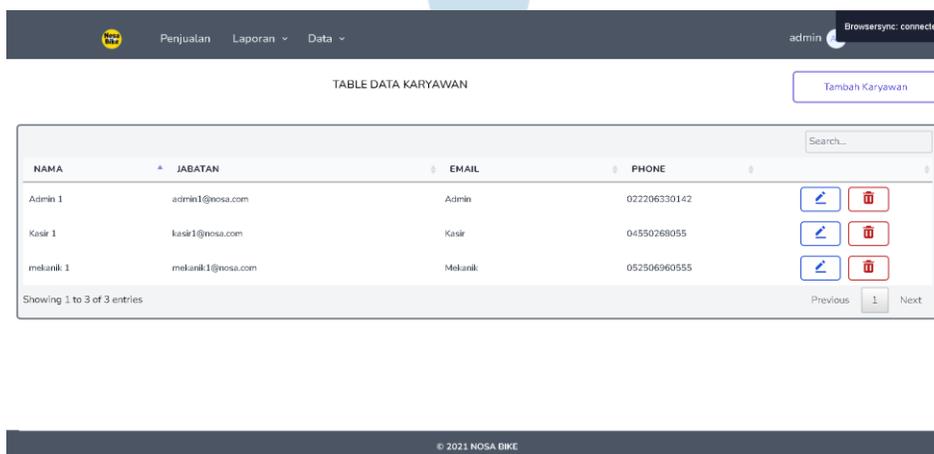


**Gambar 5.14 Delete Supplier UI**

Gambar 5.14 merupakan halaman *Delete Supplier* . Halaman ini merupakan *pop-up* ketika User Internal menekan tombol dengan *icon delete* (Berwarna Merah), pada form ini dapat dilakukan *delete* data Supplier yang terdapat didalam *database* ketika menekan tombol *OK*.

### 5.3.5. Antarmuka Karyawan UI

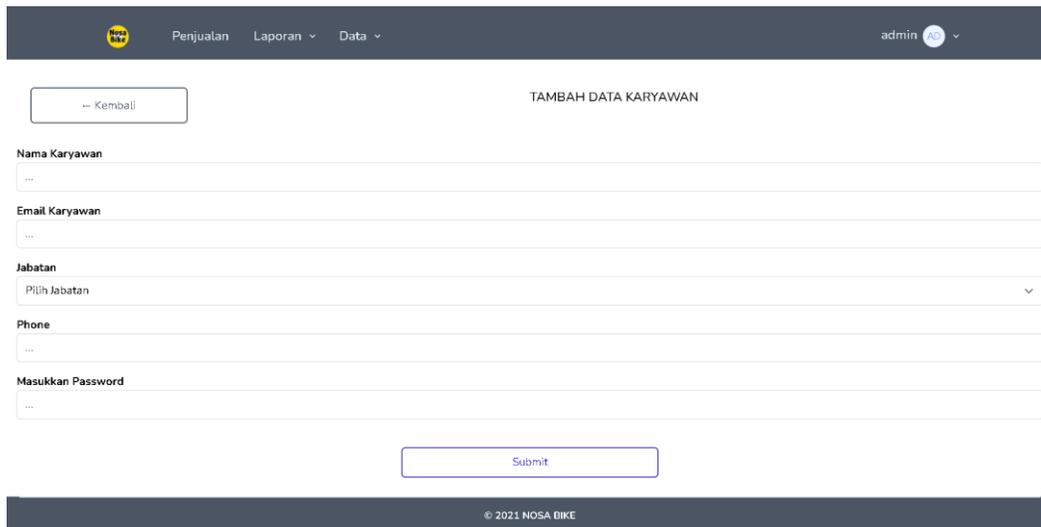
#### 5.3.5.1. Antarmuka *Display\_Karyawan()*



**Gambar 5.15 Display Karyawan UI**

Gambar 5.15 merupakan tampilan halaman *Display* Karyawan. Halaman ini merupakan halaman ketika User Internal mengakses Karyawan, dimana semua data akan ditampilkan dalam bentuk tabel.

### 5.3.5.2. Antarmuka *Add\_Karyawan()*



The screenshot shows a web application interface for adding a new employee. At the top, there is a dark navigation bar with a logo on the left, menu items 'Penjualan', 'Laporan', and 'Data' in the center, and a user profile 'admin' on the right. Below the navigation bar, there is a button labeled '-- Kembali' and a title 'TAMBAH DATA KARYAWAN'. The form consists of several input fields: 'Nama Karyawan', 'Email Karyawan', 'Jabatan' (a dropdown menu with 'Pilih Jabatan' selected), 'Phone', and 'Masukkan Password'. A 'Submit' button is located at the bottom of the form. At the very bottom of the page, there is a dark footer with the text '© 2021 NOSA BIKE'.

**Gambar 5.16 Add Karyawan UI**

Gambar 5.16 merupakan halaman *Add* Karyawan. Halaman ini merupakan tampilan ketika User Internal menekan tombol Tambah Karyawan, pada form ini dapat dilakukan *input* data Karyawan ke dalam *database* dengan memasukkan data barang dan menekan tombol *Submit*.

### 5.3.5.3. Antarmuka *Edit\_Karyawan()*

Penjualan Laporan Data admin

← Kembali EDIT DATA KARYAWAN

Nama Karyawan :  
Admin 1

Email karyawan :  
admin1@nosa.com

Jabatan Karyawan :  
Admin

Nomor HP Karyawan :  
022206330142

Update

© 2021 NOSA BIKE

**Gambar 5.17 Edit Karyawan UI**

Gambar 5.17 merupakan halaman *Edit Karyawan*. Halaman ini merupakan tampilan ketika User Internal menekan tombol dengan *icon edit* (Berwarna Biru), pada form ini dapat dilakukan *edit* data Karyawan yang terdapat didalam *database* ketika menekan tombol *Update*.

### 5.3.5.4. Antarmuka *Delete\_Karyawan()*

Penjualan Laporan Data admin

TABLE DATA KARYAWAN Tambah Karyawan

NAMA	JABATAN	PHONE
Admin 1	admin1@nosa.com	022206330142
Kasir 1	kasir1@nosa.com	04550269055
mekanik 1	mekanik1@nosa.com	052506960555

Showing 1 to 3 of 3 entries

Previous 1 Next

! Apa anda yakin?  
Data akan hilang setelah dihapus.  
Cancel OK

© 2021 NOSA BIKE

**Gambar 5.18 Delete Karyawan UI**

Gambar 5.18 merupakan halaman *Delete* Karyawan . Halaman ini merupakan *pop-up* ketika User Internal menekan tombol dengan *icon delete* (Berwarna Merah), pada form ini dapat dilakukan *delete* data Karyawan yang terdapat didalam *database* ketika menekan tombol *OK*.

### 5.3.6. Antarmuka Pelanggan UI

#### 5.3.6.1. Antarmuka *Display\_Pelanggan()*

NAMA	ALAMAT	TANGGAL LAHIR	PHONE	
Pelanggan 1	Jalan Pelanggan 1	2022-01-04	05440789022	 
Pelanggan 2	Jatan Pelanggan 2	2022-01-16	025803690147	 
Pelanggan 3	Jalan Pelanggan 3	2021-11-18	047805698022	 
Pelanggan 4	Jatan Pelanggan 4	2021-10-14	05660220888	 

**Gambar 5.19 *Display Pelanggan* UI**

Gambar 5.19 merupakan tampilan halaman *Display Pelanggan*. Halaman ini merupakan halaman ketika User Internal mengakses Pelanggan, dimana semua data akan ditampilkan dalam bentuk tabel.

### 5.3.6.2. Antarmuka *Add\_Pelanggan()*

Penjualan Laporan Data admin

— Kembali TAMBAH DATA PELANGGAN

Nama Pelanggan:  
...

Alamat:  
...

Tanggal Lahir:  
dd/mm/yyyy

Phone:  
...

Submit

© 2021 NOSA BIKE

**Gambar 5.20 Add Pelanggan UI**

Gambar 5.20 merupakan halaman *Add Pelanggan*. Halaman ini merupakan tampilan ketika User Internal menekan tombol *Tambah Pelanggan*, pada form ini dapat dilakukan *input* data *Pelanggan* ke dalam *database* dengan memasukkan data barang dan menekan tombol *Submit*.

### 5.3.6.3. Antarmuka *Edit\_Pelanggan()*

Penjualan Laporan Data admin Browsersync: connected

— Kembali EDIT DATA PELANGGAN

Nama Pelanggan :  
Pelanggan 1

Alamat Pelanggan :  
Jalan Pelanggan 1

Tanggal Lahir :  
04/01/2022

Nomor HP Pelanggan :  
05440788022

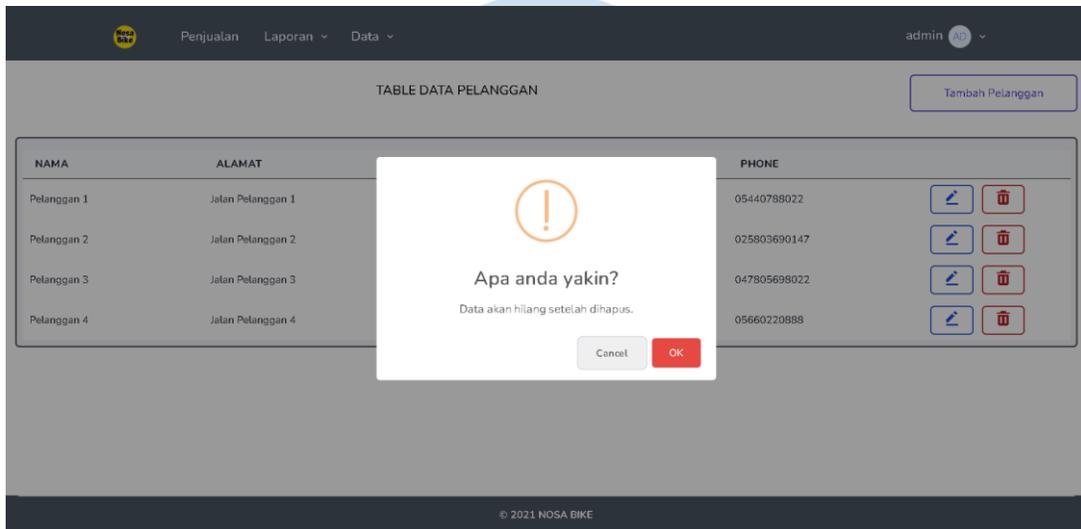
Update

© 2021 NOSA BIKE

**Gambar 5.21 Edit Pelanggan UI**

Gambar 5.21 merupakan halaman *Edit* Pelanggan. Halaman ini merupakan tampilan ketika User Internal menekan tombol dengan *icon edit* (Berwarna Biru), pada form ini dapat dilakukan *edit* data Pelanggan yang terdapat didalam *database* ketika menekan tombol *Update*.

#### 5.3.6.4. Antarmuka *Delete\_Pelanggan()*



**Gambar 5.22 *Delete Pelanggan* UI**

Gambar 5.22 merupakan halaman *Delete* Pelanggan. Halaman ini merupakan *pop-up* ketika User Internal menekan tombol dengan *icon delete* (Berwarna Merah), pada form ini dapat dilakukan *delete* data Pelanggan yang terdapat didalam *database* ketika menekan tombol *OK*.

### 5.3.7. Antarmuka Transaksi UI

#### 5.3.7.1. Antarmuka *Add\_Barang()*

Data Barang ×

Barcode ▲	Nama Barang ↕	Tipe Barang ↕	Jumlah Barang ↕	Harga Barang ↕	
21370	Wheelset 1	Wheelset	<input type="text" value="1"/>	<input type="text" value="650000"/>	<input type="button" value="Add To Cart"/>
52318	Dropbar 2	Steering	<input type="text" value="1"/>	<input type="text" value="50000"/>	<input type="button" value="Add To Cart"/>
98375	Helm Sepeda 1	Aksesoris	<input type="text" value="1"/>	<input type="text" value="60000"/>	<input type="button" value="Add To Cart"/>

Showing 1 to 3 of 3 entriesPrevious  Next

**Gambar 5.23 Add Barang UI**

Gambar 5.23 merupakan halaman *Add Barang*. Halaman ini merupakan *pop-up* ketika User Internal menekan tombol dengan +, pada form ini dapat dilakukan adalah menambahkan data barang yang akan diproses. User juga dapat mengubah harga dan jumlah stok yang ingin diproses, serta penggunaan alat bantu *Scanner* dapat digunakan untuk mempermudah pencarian barang

#### 5.3.7.2. Antarmuka *Show\_Transaksi()*

Penjualan Laporan Dataadmin

TRANSAKSI PENJUALAN BARANG

NAMA BARANG ▲	JUMLAH BARANG ↕	HARGA ↕	ACTION ↕
No data available in table			
Total	Rp. 0	Jumlah Barang	0 Barang

Showing 0 to 0 of 0 entriesPrevious Next

Tambah Barang

Pilih Pelanggan

Masukkan Jumlah Uang

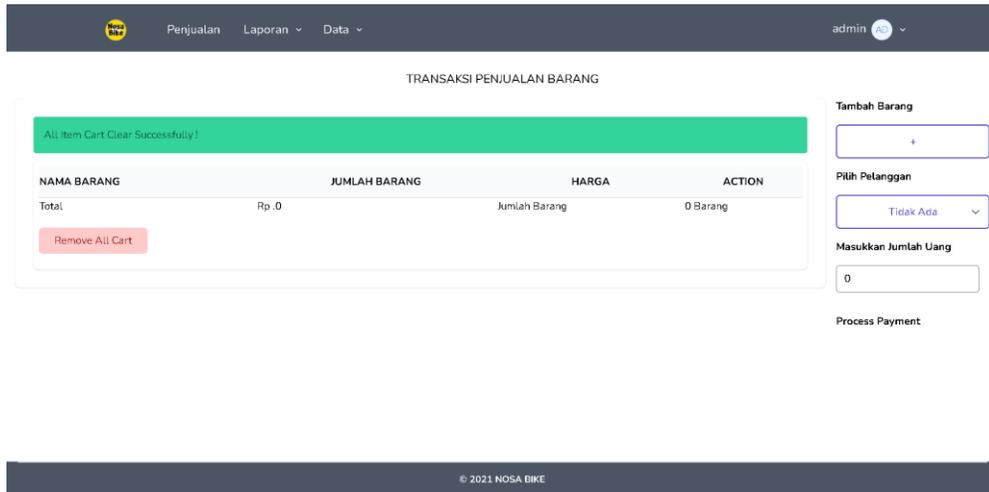
Process Payment

© 2021 NOSA BIKE

**Gambar 5.24 Show Transaksi UI**

Gambar 5.24 merupakan tampilan halaman *Show* Transaksi. Halaman ini merupakan halaman ketika User Internal mengakses Transaksi, dimana semua data pelanggan akan diambil dan tabel tempat menyimpan barang dalam keadaan kosong

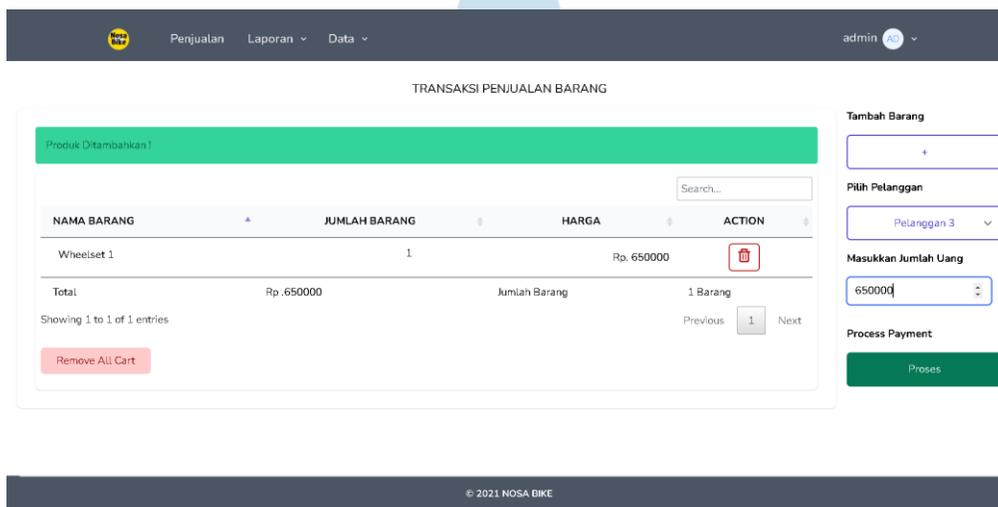
### 5.3.7.3. Antarmuka *Cancel\_Transaksi()*



**Gambar 5.25 *Cancel* Transaksi UI**

Gambar 5.25 merupakan tampilan halaman *Cancel* Transaksi. Halaman ini merupakan halaman ketika User Internal menekan tombol "Remove All Cart", dimana semua data barang yang sudah dimasukkan akan dihapus dari daftar barang

### 5.3.7.4. Antarmuka *Process Payment()*

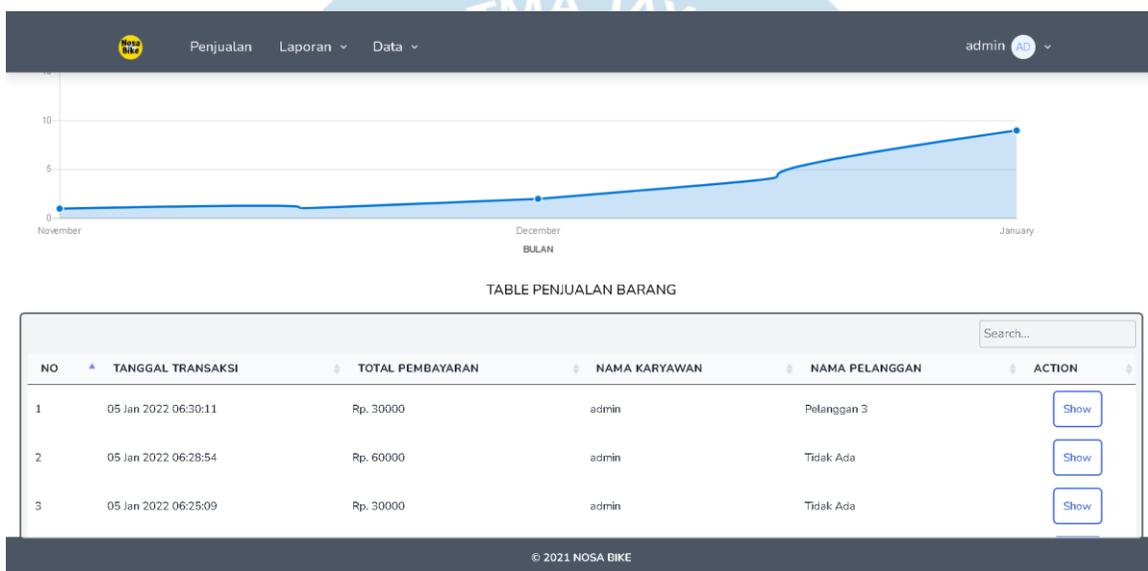


**Gambar 5.26 *Process Payment* UI**

Gambar 5.26 merupakan tampilan halaman *Process Payment*. Halaman ini merupakan halaman ketika User Internal sudah menambahkan barang dan memasukkan jumlah uang dengan nominal sama dengan atau lebih dari total pembayaran, maka tombol "Proses" akan tampil yang akan memproses data transaksi untuk disimpan ke *database*

### 5.3.8. Antarmuka Laporan Penjualan UI

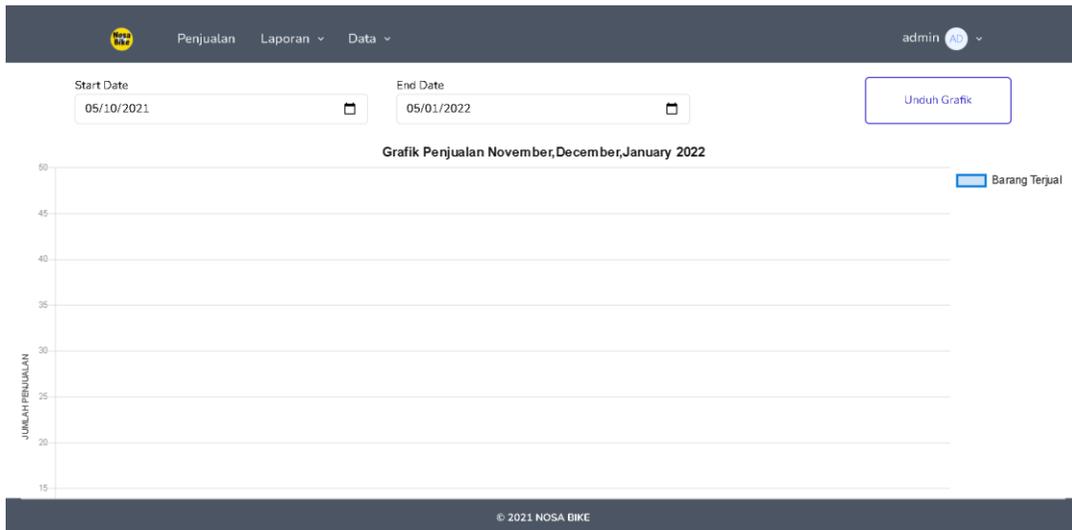
#### 5.3.8.1. Antarmuka *Display\_Laporan()*



**Gambar 5.27 Display Laporan Penjualan UI**

Gambar 5.27 merupakan tampilan halaman *Display* Laporan Penjualan. Halaman ini merupakan halaman ketika User Internal mengakses Laporan Penjualan, dimana semua data transaksi akan diambil dan ditampilkan dalam bentuk tabel dan grafik garis

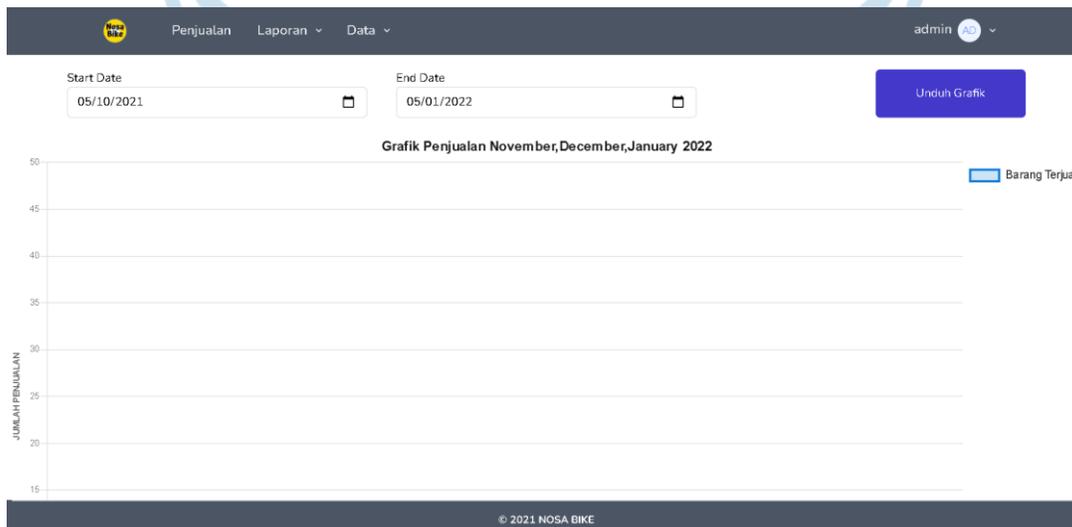
### 5.3.8.2. Antarmuka *Filter()*



**Gambar 5.28 *Filter Penjualan UI***

Gambar 5.28 merupakan tampilan halaman *Filter Penjualan*, dimana semua data transaksi akan diambil berdasarkan tanggal *start* dan *end* sehingga data yang ditampilkan hanya data yang dimasukkan ke *database* dalam jangka waktu yang dipilih

### 5.3.8.3. Antarmuka *Generate\_PDF()*



**Gambar 5.29 *Generate PDF UI***

Gambar 5.29 merupakan tampilan halaman *Generate PDF*, dimana grafik garis akan dicetak dalam format PDF

## 5.3.9. Antarmuka Detail Laporan UI

### 5.3.9.1. Antarmuka Detail()

DETAIL PENJUALAN

NO	NAMA BARANG	JUMLAH BARANG	HARGA
1	Wheelset 1	2	Rp. 550,000.00

Showing 1 to 1 of 1 entries

© 2021 NOSA BIKE

**Gambar 5.30 Detail Laporan UI**

Gambar 5.30 merupakan tampilan halaman Detail Laporan, dimana ketika user menekan tombol "Show" pada tampilan *Display* Laporan, maka user akan ditampilkan detail dari transaksi tersebut sesuai data yang dipilih

## 5.3.10. Antarmuka Laporan Pembelian UI

### 5.3.10.1. Antarmuka *Display*\_Laporan()

TABLE PEMBELIAN BARANG

Start Date: 05/10/2021 End Date: 05/01/2022

NO	TANGGAL TRANSAKSI	NAMA PRODUK	JUMLAH PRODUK	JUMLAH PRODUK SAAT INI	NAMA SUPPLIER
1	05 Jan 2022 06:13:30	Wheelset 1	2	5	Supplier Ban Dalam 1
2	05 Jan 2022 06:05:33	Botol	0	2	Supplier Part 1
3	05 Jan 2022 06:04:56	Helm Sepeda 1	5	6	Supplier Part 2
4	05 Jan 2022 06:02:36	Wheelset 1	0	5	Supplier Sepeda 2

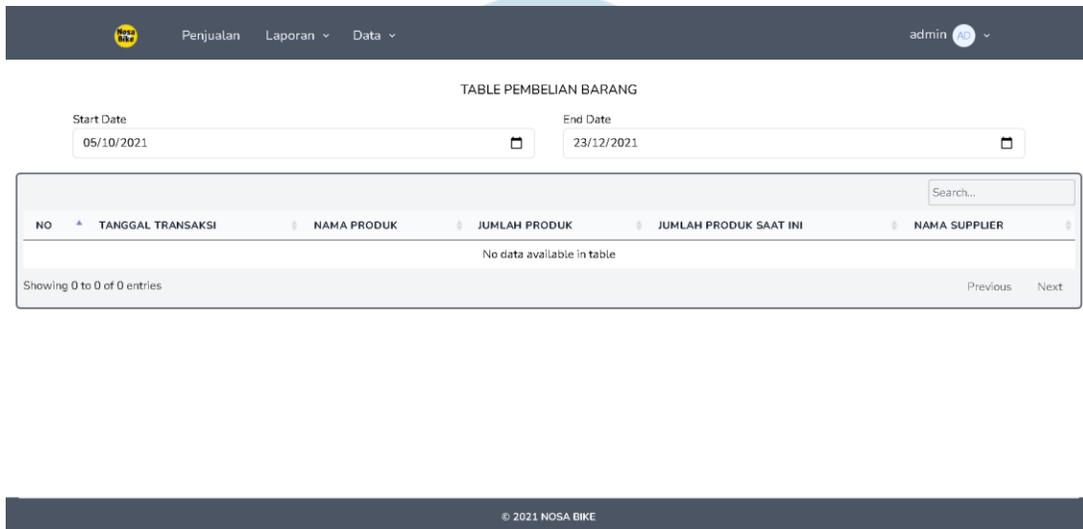
Showing 1 to 4 of 4 entries

© 2021 NOSA BIKE

### Gambar 5.31 *Display* Laporan Pembelian UI

Gambar 5.31 merupakan tampilan halaman *Display* Laporan Pembelian. Halaman ini merupakan halaman ketika User Internal mengakses Laporan Pembelian, dimana semua data transaksi akan diambil dan ditampilkan dalam bentuk tabel

#### 5.3.10.2. Antarmuka *Filter*()

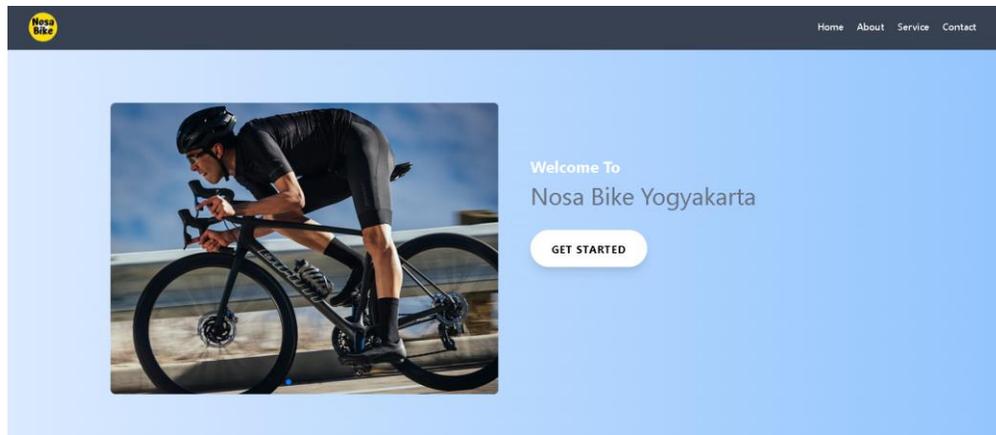


### Gambar 5.32 *Filter* Pembelian UI

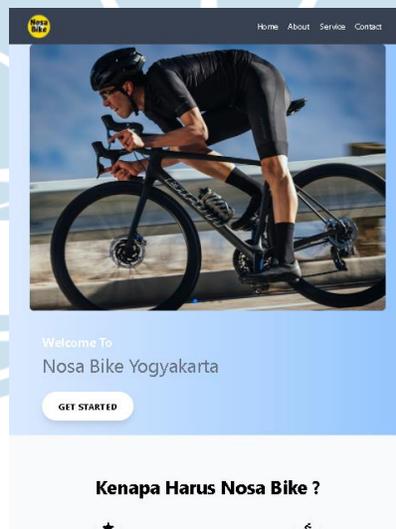
Gambar 5.32 merupakan tampilan halaman *Filter* Pembelian, dimana semua data barang akan diambil berdasarkan tanggal *start* dan *end* sehingga data yang ditampilkan hanya data yang dimasukkan ke *database* dalam jangka waktu yang dipilih

### 5.3.11. Antarmuka *Company Profile* UI

#### 5.3.11.1. Antarmuka Tampilan()



Gambar 5.33 *Company Profile* UI



Gambar 5.34 *Company Profile* UI (Mobile)

Gambar 5.33 dan 5,34 merupakan halaman tampilan *Company Profile*. Halaman ini merupakan halaman utama yang dapat diakses semua user dan tidak memerlukan autentifikasi. Halaman *Company Profile* sendiri bersifat *Single-Page Application* (SPA) dimana semua tampilan hanya disajikan dalam 1 tampilan tanpa mengurangi fungsi atau informasi yang disampaikan. Tampilan ini juga bersifat *Responsive* sehingga pengguna dari berbagai perangkat dapat mengakses tampilan tanpa merusak konten dari *Company Profile* tersebut

## **BAB VI**

### **KESIMPULAN DAN SARAN**

#### **6.1 Kesimpulan**

Pembahasan penelitian ini dilakukan untuk merancang sistem informasi berbasis website yang dapat digunakan oleh Nosa Bike. Pengembangan sistem dilakukan dengan mengimplementasi metode *Rapid Application Development* dengan tujuan dapat mempercepat proses pembuatan sistem hingga dapat digunakan. Berdasarkan analisis yang dilakukan, terdapat 32 fungsi yang dibutuhkan untuk mendukung proses bisnis dari Nosa Bike. Fungsi-fungsi tersebut didapatkan dari observasi peneliti dan wawancara secara langsung dengan pemilik Nosa Bike. Hasil dari perancangan pada penelitian ini dibagi menjadi 3 yaitu ERD (*Entity Relationship Diagram*), *Class Diagram*, dan Perancangan Antarmuka.

#### **6.2 Saran**

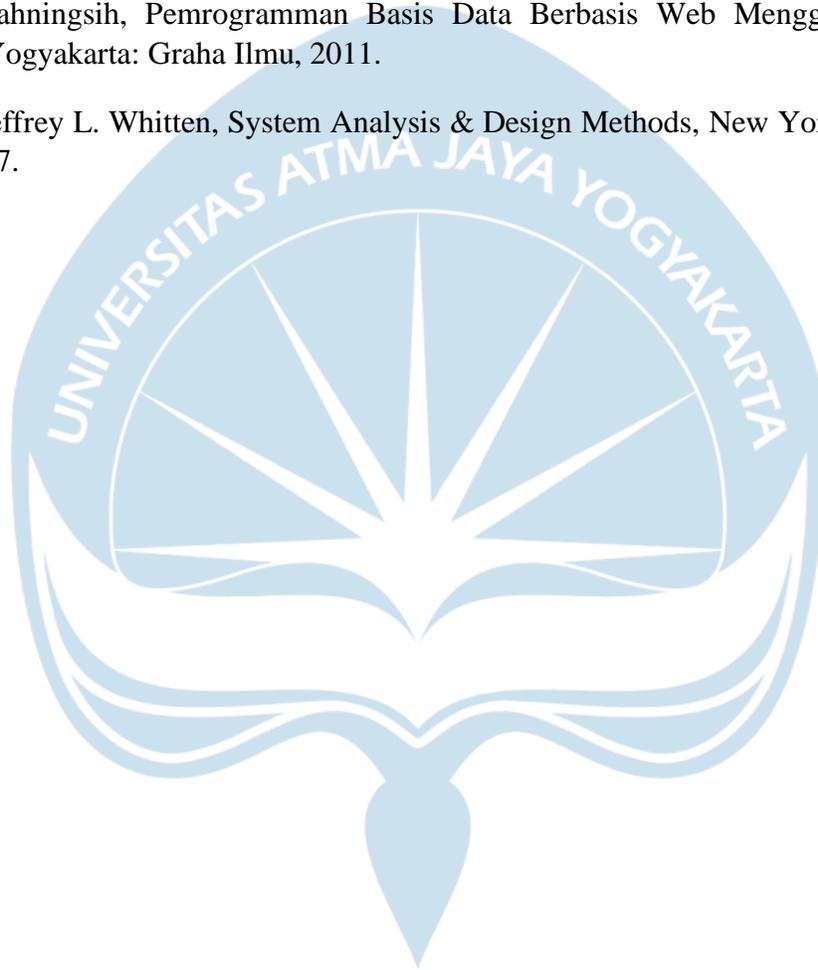
Berdasarkan hasil yang didapatkan dari penelitian ini, diharapkan untuk penelitian dimasa mendatang yang memiliki topik yang sama dapat memberikan peningkatan seperti:

1. Penelitian terhadap studi kasus website Nosa Bike masih belum terbentuk dengan tampilan mobile-friendly sehingga penelitian lanjutan dapat dilakukan.
2. Penggunaan *Rapid Application Development* lebih banyak digunakan sebagai metode pengembangan sistem.
3. Penelitian lanjutan berupa uji coba keefektifan sistem terhadap proses bisnis Nosa Bike.

## DAFTAR PUSTAKA

- [1] E. M. Rogers, Diffusion of Innovations, London: The Free Press, 1983.
- [2] B. Oetomo, E-ducation: Konsep Teknologi dan Aplikasi Internet Pendidikan, Yogyakarta: Andi, 2002.
- [3] N. M. D. Wulansari, Didiklah Anak Sesuai Zamannya, Jakarta: Transmedia Pustaka, 2017.
- [4] D. P. Yulia Djahir, Sistem Informasi Manajemen, Yogyakarta: CV. Budi Utama, 2015.
- [5] A. F. K. Sibero, Kitab Suci Web Programming, Yogyakarta: Mediakom, 2011.
- [6] A. M. Rudianto, Pemrograman Web Dinamis Menggunakan PHP dan MySQL, Yogyakarta: C.V ANDI OFFSET, 2011.
- [7] L. Sunarya, Diktat Mata Kuliah Desain Karakter dan Modeling, Tangerang, 2010.
- [8] A. A. P. N. N. Eva Rahmawati, "Rancang Bangun Sistem Informasi Berbasis Web Pada SMK MADANI DEPOK Menggunakan Framework Codeigniter," *Jurnal Rekayasa Perangkat Lunak*, vol. 2, no. 1, 2021.
- [9] H. E. Meidyana Permata Putri, "Implementasi Metode Rapid Application Development Pada Website Service Guide "Waterfall Tour South Sumatera"," *Jurnal SISFOKOM*, vol. 07, no. 02, 2018.
- [10] S. H. A. Andi Christian, "Rancang Bangun Website Sekolah Dengan Menggunakan Framework Bootstrap (Studi Kasus SMP Negeri 6 Prabumulih)," *Jurnal SISFOKOM*, vol. 7, no. 1, 2018.
- [11] R. Triyanto, "Rancang Bangun Aplikasi Penjualan Berbasis Website (Studi Kasus : Toko Waroeng Bola)," *Jurnal Sistem Informasi dan Sains Teknologi*, vol. 2, 2020.
- [12] A. D. Fuad, "Analisis Kebutuhan dan Perancangan Sistem Pengelolaan Web & App PT.XYZ Berbasis Website," *Universitas Atma Jaya Yogyakarta*, vol. 1, no. 1, p. 201, 2021.
- [13] A. R. S. M. Sukanto, Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek, Bandung: Informatika. Winarno, Wing Wahyu (2006), 2013.
- [14] J. Hutahaean, Konsep Sistem Informasi, Yogyakarta: DEEPUBLISH, 2014.
- [15] H. B. Bakti, Mahir Membuat Website dengan Adobe Dreamweaver CS6, CS5 dan JQuery, Yogyakarta: C. V. Andi Offset, 2015.

- [16] A. Supriyanto, Web dengan HTML dan XML, 2007: Graha Ilmu, Yogyakarta.
- [17] Sutarman, Membangun Aplikasi Web PHP dan MySQL, Yogyakarta: PENERBIT ANDI, 2003.
- [18] A. Kadir, Dasar Pemrograman Web Dinamis Menggunakan PHP, Yogyakarta: PENERBIT ANDI, 2008.
- [19] D. Kustiyahningsih, Pemrograman Basis Data Berbasis Web Menggunakan PHP & MySQL, Yogyakarta: Graha Ilmu, 2011.
- [20] L. D. B. Jeffrey L. Whitten, System Analysis & Design Methods, New York: McGraw-Hill Irwin, 2007.



## LAMPIRAN

Monrev90\_YuliusWilsonGunawan

### ORIGINALITY REPORT

<b>15%</b>	<b>14%</b>	<b>4%</b>	<b>6%</b>
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

### PRIMARY SOURCES

<b>1</b>	<b>e-journal.uajy.ac.id</b> Internet Source	<b>3%</b>
<b>2</b>	<b>Submitted to Universitas Atma Jaya Yogyakarta</b> Student Paper	<b>2%</b>
<b>3</b>	<b>www.scribd.com</b> Internet Source	<b>2%</b>
<b>4</b>	<b>repository.dinamika.ac.id</b> Internet Source	<b>1%</b>
<b>5</b>	<b>repository.bsi.ac.id</b> Internet Source	<b>1%</b>
<b>6</b>	<b>eprints.uty.ac.id</b> Internet Source	<b>&lt;1%</b>
<b>7</b>	<b>docplayer.info</b> Internet Source	<b>&lt;1%</b>
<b>8</b>	<b>dppm.uii.ac.id</b> Internet Source	<b>&lt;1%</b>
<b>9</b>	<b>Submitted to UIN Maulana Malik Ibrahim Malang</b>	<b>&lt;1%</b>



## Transkrip Wawancara Nosa Bike

### Lampiran 1 Kondisi Toko Nosa Bike



### Lampiran 2 Transkrip Wawancara

Nama : Arifin Putra Surya

Tanggal : 2 Agustus 2021

- Wilson : "Halo bosku, ini yang tentang kemarin mau buatin sistem buat bantu toko, sekalian mau jadiin topik skripsiku kalau boleh bosku."
- Arifin : "Boleh-boleh sekalian nanti rencana mau punya sistem baru kalau nanti jadi pindah toko, gapapa sekalian bantu-bantu toko juga lah."
- Wilson : "Jadi sistem nya yang mau dipake kira-kira mau dibuat gimana bosku? Cuma untuk olah data atau ada yang lain juga kira-kira?"
- Arifin : "Yang utama kan bantu buat stok barang, karena ni barang padet banget jadi kadang ga ketahuan ada berapa stok sama kalau bisa ada fungsi mirip kasir lah jadi agak mudah nanti."
- Wilson : "Ini mumpung waktu kuliah ada coba buat sistem mirip gitu, ini mumpung bawa laptop mau sekalian tak liatin kalau mau. (Memperlihatkan Sistem)"
- Arifin : "Boleh sih kalau buat gini, udah agak lengkap juga ya isinya?"
- Wilson : "Iya bosku ini sebenarnya bisa langsung pake tinggal ubah sesuai nanti butuhnya apa aja."
- Arifin : "Yowes itu paling nanti coba buat-buat dulu aja, yang utama kan buat olah barang sama penjualan barang, jadi paling dicoba buat dulu aja."

- Wilson : "Kalau tampilannya mau dibuat gimana bosku, ini kan kalau pake yang ini tampilannya biasa aja, kalau mau ada namanya pake Template Bootstrap (Memperlihatkan contoh) jadi lebih bagus."
- Arifin : "Gitu aja ga masalah sih, kan cuma buat karyawan juga yang lihat, biasa-biasa aja ga masalah."
- Wilson : "Oke bosku nanti tak coba garap dulu, kira-kira mau dipake kapan bosku kalau sudah ada sistem? Biar susun jadwal biar kelar."
- Arifin : "Belum tau sih mau kapan, masih cari kontrakan buat toko baru, palingan nanti kalau sudah ketemu kontrakan tak kabarin."
- Wilson : "Siap bosku, nanti kebutuhannya aku chat lewat WA aja ya biar ga lupa."
- Arifin : "Siap kabarin aja nanti butuh apa aja."

Tanggal : 9 Agustus 2021

- Wilson : "Halo bosku, ini udah jadi beberapa udah jadi, buat barang sama penjualannya."
- Arifin : "Oke udah bagus ini, kemarin aku lupa itu kalau bisa nanti buat ada tempat buat simpan data pelanggan, supplier, sama karyawan juga. Buat simpen aja biar enak carinya nanti."
- Wilson : "Siap bosku, ini mau sekalian buatin Login juga biar aman?"
- Arifin : "Nah boleh juga, nanti jadi cuma karyawan aja yang bisa buka kan?"
- Wilson : "Iya bosku cuma yang terdaftar aja bisa masuk sistemnya."
- Arifin : "Mantap boleh lah, itu sama kepikiran nanti di toko baru mau pake barcode buat data barangnya, jadi nanti bisa tinggal scan barcode."
- Wilson : "Bisa sih kalau mau, barcodenya nanti kita cetak sendiri artinya?"
- Arifin : "Iya jadi kita punya barcode sendiri, nanti tinggal scan aja kalau orang mau beli."
- Wilson : "Siap bosku nanti tak coba cari-cari caranya, ini sama nanti mau ditambahin apa lagi kira-kira bosku?"
- Arifin : "Diskon barang boleh juga kalau bisa, jadi nanti tinggal masuk mau diskon berapa nanti berubah harga sama ada semacam keterangan barang diskon gitu."
- Wilson : "Oke bosku bisa nanti tak coba garap, nanti kalau sudah selesai nanti aku tak kabarin lagi bosku."
- Arifin : "Sama kalau bisa buatin tampilan Nosa yang baru gimana? Yang lama udah ga diurus lagi masih pake foto toko lama"
- Wilson : "Ooohh bisa-bisa nanti kucoba cariin tampilan yang bagus, paling nanti aku minta foto-foto Nosa aja."
- Arifin : "Nanti ada temenku yang pinter foto, nanti aku minta tolong dia fotoin aja."
- Wilson : "Siap bosku, nanti kalau sudah kelar aku tak kabarin lagi."
- Arifin : "Siap."

Tanggal : 16 Agustus 2021

- Wilson : "Halo bosku, ini udah kelar bosku, barcodenya sama tampilan depannya sudah rapi bosku. (Memperlihatkan Sistem)"
- Arifin : "Ooh ini yang barcodenya tampilin nama barang sama harga aja kalau bisa, tampilan depan udah cocok ini, nanti tinggal kasih foto sudah bagus sama paling kontak tampilan depannya tambahin online store aja."
- Wilson : "Tambahin semua bosku? Tokopedia, Shopee?"
- Arifin : "Iya sama Facebook Instagram juga boleh."
- Wilson : "Siap bosku artinya tinggal atur-atur ini aja bosku ga ada yang lain?"
- Arifin : "Sejauh ini aman paling nanti kalau aku kepikir fungsi baru tak kabarin lagi. Ini karena udah dapet kontrakan jadi minggu depan sudah pindah ke toko baru."
- Wilson : "Siap bosku."

Tanggal : 22 Agustus 2021

- Wilson : "Halo bosku, ini sudah kelar semua bosku. Ini boleh dilihat-lihat dulu. (Memperlihatkan Sistem)"
- Arifin : "Ini fungsi barcodenya masih kegedean ukurannya, takutnya nanti kebesaran. Paling nanti kecilin dikit aja."
- Wilson : "Mau dibuat ukuran berapa kira-kira bosku?"
- Arifin : "Mungkin 3x8 Cm, sekitar itu bisa. Takutnya kalo barang kecil susah ditempel."
- Wilson : "Siap bosku ini langsung tak coba ganti ukurannya."
- Arifin : "Ini nanti sekalian temen minggu depan dateng mau fotoin toko, nanti fotonya tak kirimin. Itu kalau mau sekalian coba-coba cari hostingan juga kalau bisa."
- Wilson : "Kira-kira mau beli harga berapa hostingan web nya bos?"
- Arifin : "Nanti kucoba tanya temenku dulu, dia kemarin sempat jualan hosting, nanti tak kabarin kalau harga cocok."
- Wilson : "Siap bosku. Nanti alatnya enakan gimana bosku, buat scan sama printnya?"
- Arifin : "Aku sempat cek online cuma harganya beda jauh-jauh semua. Nanti bisa minta tolong sekalian cariin yang cocok alatnya?"
- Wilson : "Siap bosku nanti sekalian kucariin."
- Arifin : "Siap."

Tanggal : 3 September 2021

- Wilson : "Halo bosku, ini sistemnya sudah selesai, itu soal hostingan aku udah tanya temenku, harganya kisaran 300 – 400 ribuan setahun."
- Arifin : "Temenku juga bilang gitu, jadi paling nanti tak coba kabarin lagi deh."
- Wilson : "Ini sistemnya kalau mau dipake bisa di localhost dulu, jadi ga perlu hosting kalau mau."
- Arifin : "Gapapa nanti sekalian hosting aja, biar ga 2 kali kerja."
- Wilson : "Siap bosku nanti boleh minta tolong foto bareng buat bukti penyerahan sistemnya bosku?"
- Arifin : "Siap boleh."

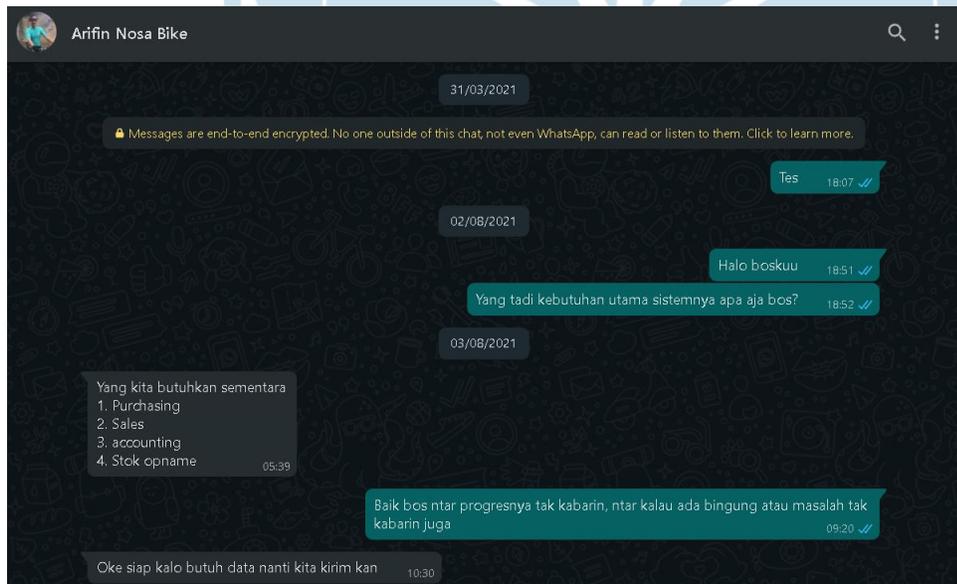
Tanggal : 18 September 2021

- Wilson : "Halo bosku, ini tadi sempat coba liat-liat alat printer barcode sama scanner-nya, kisaran harga buat printer-nya 750 ribuan bosku."
- Arifin : "Itu tapi ukurannya bisa diatur sebelum di print?"
- Wilson : "Bisa bosku ini nanti pake aplikasi bawaan printernya, jadi sebelum di print bisa diatur dulu."
- Arifin : "Oke nanti aku coba liatin, sama ini website-nya bisa diubah dikit di bagian input barang-barangnya? Ini kemarin pas coba kan cuma bisa pilihan yang udah ada aja, ini bisa dibuat bisa ketik sendiri aja?"
- Wilson : "Bisa bosku, ini nanti aku ubahin nanti langsung tak kasih pake. Kalau yang lain kira-kira gimana bosku? Lancar atau kira-kira butuh ubahan?"
- Arifin : "Mungkin agak lamban atau internet ya? Ini pas kemarin dicoba pake masukkin data barang dia agak ngelag, atau mungkin karena datanya kebanyakan jadi ngelag?"
- Wilson : "Nggak bosku, kalau soal data ini harusnya kuat kalau isinya banyak, kalau internet masalahnya palingan nanti cuma dibagian laporan bagian grafik, sama download barcode sama grafik-grafik aja karena butuh internet."
- Arifin : "Ini nanti kira-kira bisa dibenerin dulu? Atau enakan gimana?"
- Wilson : "Kucoba cek dulu aja ya bosku, kalau misal ternyata masalahnya minor atau kecil-kecil nanti bisa kubenerin, kalau semisal ternyata memang masalah dari sistem yang kubuat, nanti kucoba cariin solusi baru."
- Arifin : "Oke nanti kalau bisa paling penting di barcode dulu aja ya, biar bisa ngeprint barcode."
- Wilson : "Siap bosku, kucoba kabarin nanti kalau ternyata kubuat in baru sistemnya gapapa bosku?"
- Arifin : "Gapapa nanti kabarin aja."

Tanggal : 22 September 2021

- Wilson : "Halo bosku, ini aku ada cek sistemnya. Jadi setelah kucek memang dari sistemnya yang kemungkinan bikin ngelag, jadi ini sementara kubenerin dulu biar bisa dipake, terus aku nanti coba buatin sistem baru pake yang lebih ringan, gimana bosku?"
- Arifin : "Boleh yang penting nanti bisa dipake."
- Wilson : "Oke bosku, ini aku rencananya bakal kelar sebulan, nanti kan aku tapi udah di Jakarta bosku, jadi nanti aku coba urusin online aja gapapa ya bosku?"
- Arifin : "Oke, ini kan nanti juga mau ada bisnis sama Metafora, nanti kalau bisa website nya digabung aja biar 1 hosting."
- Wilson : "Boleh bosku, nanti boleh dikabarin aja bosku, ini aku ntar coba buatin sistemnya sesuai yang lama, palingan ntar kutambahin beberapa fitur yang bisa bantu juga."
- Arifin : "Oke boleh ntar kabarin aja."
- Wilson : "Siap bosku."

### Lampiran 3 Hasil Wawancara Kebutuhan Pertama



[18:51, 02/08/2021] : Halo boskuu

Wilson Gunawan

[18:52, 02/08/2021] : Yang tadi kebutuhan utama sistemnya apa aja bos?

Wilson Gunawan

[05:39, 03/08/2021] : Yang kita butuhkan sementara

Arifin Nosa Bike

1. Purchasing

2. Sales

3. accounting

4. Stok opname

[09:20, 03/08/2021] : Baik bos ntar progresnya tak kabarin, ntar kalau ada bingung atau masalah tak kabarin juga

Wilson Gunawan

[10:30, 03/08/2021] : Oke siap kalo butuh data nanti kita kirim kan

Arifin Nosa Bike

[11:50, 03/08/2021] : nanti paling data barang" yang di toko sama data supplier, tapi datanya nanti dikirim gpp

Wilson Gunawan

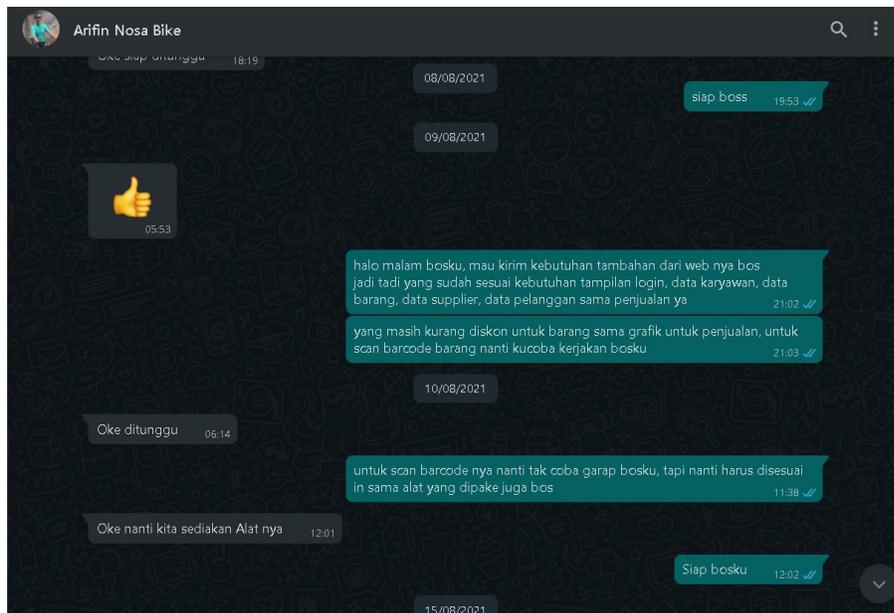
[11:54, 03/08/2021] : Oke siap

Arifin Nosa Bike

[12:07, 03/08/2021] : siapp

Wilson Gunawan

Lampiran 4 Hasil Wawancara Kebutuhan Kedua



- [21:02, 09/08/2021] : halo malam bosku, mau kirim kebutuhan tambahan dari web nya bos jadi  
Wilson Gunawan jadi tadi yang sudah sesuai kebutuhan tampilan login, data karyawan, data barang, data supplier, data pelanggan sama penjualannya
- [21:03, 09/08/2021] : yang masih kurang diskon untuk barang sama grafik untuk penjualan,  
Wilson Gunawan untuk scan barcode barang nanti kucoba kerjakan bosku
- [06:14, 10/08/2021] : Oke ditunggu  
Arifin Nosa Bike
- [11:38, 10/08/2021] : untuk scan barcode nya nanti tak coba garap bosku, tapi nanti harus  
Wilson Gunawan disesuaikan in sama alat yang dipake juga bos
- [12:01, 10/08/2021] : Oke nanti kita sediakan Alat nya  
Arifin Nosa Bike
- [12:02, 10/08/2021] : Siap bosku  
Wilson Gunawan

#### Lampiran 5 Hasil Wawancara Kebutuhan Ketiga

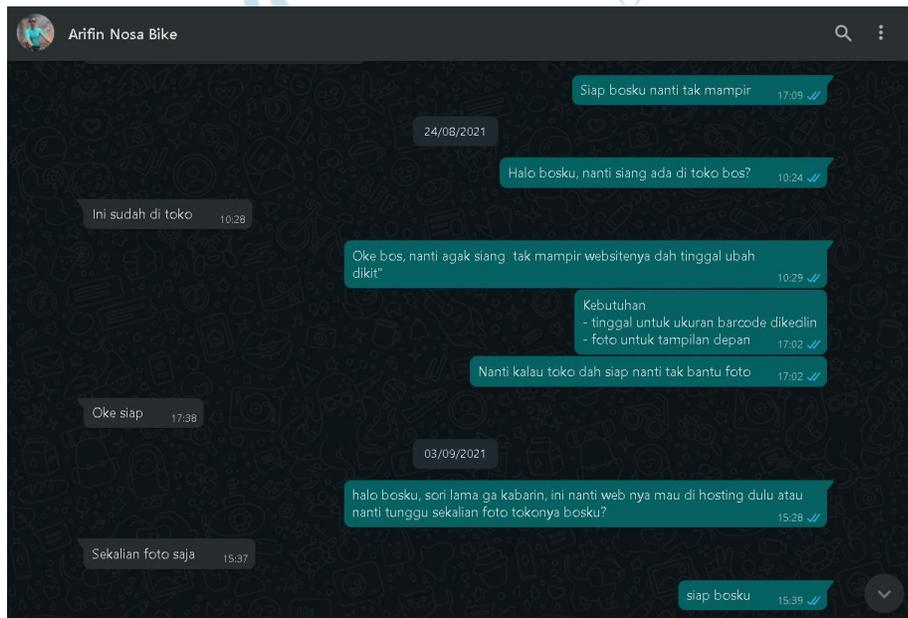


[17:36, 16/08/2021] : Halo bosku, jadi tadi websitenya tinggal:  
Wilson Gunawan

- barcode dan keterangan
- kontak di tampilan depan ditambah
- foto di website tunggu toko baru
- barcode bisa di cetak dan di scan

[17:47, 16/08/2021] : Oke  
Arifin Nosa Bike

#### Lampiran 6 Hasil Wawancara Kebutuhan Keempat



[10:24, 24/08/2021] : Halo bosku, nanti siang ada di toko bos?  
Wilson Gunawan

[10:28, 24/08/2021] : Ini sudah di toko

Arifin Nosa Bike

[10:29, 24/08/2021] : Oke bos, nanti agak siang tak mampir websitenya dah tinggal ubah dikit"

Wilson Gunawan

[17:02, 24/08/2021] : Kebutuhan

Wilson Gunawan - tinggal untuk ukuran barcode dikecilin  
- foto untuk tampilan depan

[17:02, 24/08/2021] : Nanti kalau toko dah siap nanti tak bantu foto

Wilson Gunawan

[17:38, 24/08/2021] : Oke siap

Arifin Nosa Bike

#### Lampiran 7 Bukti Penyerahan Sistem



## TABEL REVISI

No	Tugas Revisi	Halaman Revisi
1.	Perubahan Logo UAJY	Mengganti logo UAJY dengan logo UAJY terbaru (Halaman Cover)
2.	Perubahan Format Penamaan Gambar	Mengubah format gambar sesuai bab (Halaman 4-91)
3.	Perubahan Format Penamaan Tabel	Mengubah format tabel sesuai bab (Halaman 7-67)
4.	Perubahan Tahapan Penelitian	Menambahkan gambar metode penelitian serta menambahkan penjelasan tahapan penelitian (Halaman 13-16)
5.	Penambahan <i>Flowchart</i> Sistem	Menambahkan bagian <i>Flowchart</i> dari Sistem Nosa Bike (Halaman 20-21)
6.	Penambahan Tabel <i>Testing</i>	Menambahkan tabel hasil <i>testing</i> dari setiap fungsi yang digunakan (Halaman 67-71)
7.	Penambahan Lampiran	Menambahkan ringkasan reka ulang percakapan antar peneliti dan pemilik dari Nosa Bike (Halaman 99-100)