

**DPPL-xx**

**DESKRIPSI PERANCANGAN PERANGKAT LUNAK**

**Website YA-HOODIE**

untuk:

Ibu Dian Suci


Dipersiapkan oleh:

Muhammad Arya Fikriansyah	1301204066
Satria Aji Permana Siwi	1301204209
Raihan Atsal Hafizh	1301204485
Fachrul Febriana	1301204407
Alif Faidhil Ahmad	1301204141

Program Studi S1 Informatika

Fakultas Informatika, Universitas Telkom

Jl. Telekomunikasi 1, Dayeuhkolot Bandung

	<b>Prodi S1- Informatika Universitas Telkom</b>	<b>Nomor Dokumen</b>		<b>Halaman</b>
		<b><i>DPPL-xx &lt;xx:no grp&gt;</i></b>		<b><i>&lt;#&gt;/&lt;jml #</i></b>
		<b>Revisi</b>	<b><i>&lt;nomor revisi&gt;</i></b>	<b><i>Tgl: &lt;isi tanggal&gt;</i></b>

### DAFTAR PERUBAHAN

Revisi	Deskripsi
<b>A</b>	
<b>B</b>	
<b>C</b>	
<b>D</b>	
<b>E</b>	
<b>F</b>	
<b>G</b>	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperiks a oleh								
Disetuju i oleh								

### Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

## Daftar Isi

<b>Daftar Isi</b>	<b>3</b>
<b>Daftar Tabel</b>	<b>10</b>
<b>Daftar Gambar</b>	<b>11</b>
<b>1. Pendahuluan</b>	<b>12</b>
1.1 Tujuan Penulisan Dokumen	12
1.2 Lingkup Masalah	12
1.3 Definisi dan Istilah	12
1.4 Referensi	13
<b>2 Perancangan Global</b>	<b>14</b>
2.1 Rencana Lingkungan Implementasi	14
2.2 Deskripsi Arsitektur Perangkat Lunak	14
<b>3 Perancangan Rinci</b>	<b>15</b>
3.1 Realisasi Use Case	15
3.1.1 Use Case #1 <Login User>	17
3.1.1.1 Use Case Scenario #1 <Login User>	17
3.1.1.2 UI Design dan Deskripsi Objek UI #1 <Login User>	18
3.1.1.3 Identifikasi Object dan Tipe nya #1 <Login User>	19
3.1.1.4 Sequence Diagram #1 <Login User>	20
3.1.2 Use Case #2 <Login Admin>	21
3.1.2.1 Use Case Scenario #2 <Login Admin>	21
3.1.2.2 UI Design dan Deskripsi Objek UI #2 <Login Admin>	21
3.1.2.3 Identifikasi Object dan Tipe nya #2 <Login Admin>	22
3.1.2.4 Sequence Diagram #2 <Login Admin>	23
3.1.3 Use Case #3 <Login Owner>	24
3.1.3.1 Use Case Scenario #3 <Login Owner>	24
3.1.3.2 UI Design dan Deskripsi Objek UI #3 <Login Owner>	25
3.1.3.3 Identifikasi Object dan Tipe nya #3 <Login Owner>	26
3.1.3.4 Sequence Diagram #3 <Login Owner>	27
3.1.4 Use Case #4 <Register>	28
3.1.4.1 Use Case Scenario #4 <Register>	28
3.1.4.2 UI Design dan Deskripsi Objek UI #4 <Register>	28
3.1.4.3 Identifikasi Object Baru & Tipe Kelas #4 <Register>	30
3.1.4.4 Sequence Diagram #4 <Register>	30
3.1.5 Use Case #5 <Kelola Hoodie>	31
3.1.5.1 Use Case Scenario #5 <Kelola Hoodie>	31
3.1.5.2 UI Design dan Deskripsi Objek UI #5 <Kelola Hoodie>	32
3.1.5.3 Identifikasi Object dan Tipe nya #5 <Kelola Hoodie>	34

3.1.5.4 Diagram #5 <Kelola Hoodie>	34
3.1.6 Use Case #6 <Detail Pembelian >	35
3.1.6.1 Use Case Scenario #6 <Detail Pembelian >	35
3.1.6.2 UI Design dan Deskripsi Objek UI #6 <Detail Pembelian >	35
3.1.6.3 Identifikasi Object dan Tipe nya #6 <Detail Pembelian >	37
3.1.6.4 Sequence Diagram #6 <Detail Pembelian >	37
3.1.7 Use Case #7 <Laporan Penjualan >	38
3.1.7.1 Use Case Scenario #7 <Laporan Penjualan >	38
3.1.7.2 UI Design dan Deskripsi Objek UI #7 <Laporan Penjualan >	39
3.1.7.3 Identifikasi Object dan Tipe nya #7 <Laporan Penjualan >	40
3.1.7.4 Sequence Diagram #7 <Laporan Penjualan >	40
3.1.8 Use Case #8 <Order Hoodie >	41
3.1.8.1 Use Case Scenario #8 <Order Hoodie >	41
3.1.8.2 UI Design dan Deskripsi Objek UI #8 <Order Hoodie >	42
3.1.8.3 Identifikasi Object dan Tipe nya #8 <Order Hoodie >	45
3.1.8.4 Sequence Diagram #8 <Order Hoodie >	45
3.1.9 Use Case #9 <Review Hoodie>	46
3.1.9.1 Use Case Scenario #9 <Review Hoodie>	46
3.1.9.2 UI Design dan Deskripsi Objek UI #9 <Review Hoodie>	47
3.1.9.3 Identifikasi Object dan Tipe nya #9 <Review Hoodie>	49
3.1.9.4 Sequence Diagram #9 <Review Hoodie>	50
3.1.10 Use Case #10 <Statistik Review>	50
3.1.10.1 Use Case Scenario #10 <Statistik Review>	50
3.1.10.2 UI Design dan Deskripsi Objek UI #10 <Statistik Review>	51
3.1.10.3 Identifikasi Object dan Tipe nya #10 <Statistik Review>	52
3.1.10.4 Sequence Diagram #10 <Statistik Review>	53
3.1.11 Use Case #11 <Delete Customer>	53
3.1.11.1 Use Case Scenario#11 <Delete Customer>	53
3.1.11.2 UI Design dan Deskripsi Objek UI #11 <Delete Customer>	54
3.1.11.3 Identifikasi Object dan Tipe nya #11 <Delete Customer>	55
3.1.11.4 Sequence Diagram #11 <Delete Customer>	56
3.1.12 Use Case #12 <Logout User>	56
3.1.12.1 Use Case Scenario#12 <Logout User>	56
3.1.12.2 UI Design dan Deskripsi Objek UI #12 <Logout User>	57
3.1.12.3 Identifikasi Object dan Tipe nya #12 <Logout User>	58
3.1.12.4 Sequence Diagram #12 <Logout User>	59
3.1.13 Use Case #13 <Logout Admin>	59
3.1.13.1 Use Case Scenario#13 <Logout Admin>	59
3.1.13.2 UI Design dan Deskripsi Objek UI #13 <Logout Admin>	60
3.1.13.3 Identifikasi Object dan Tipe nya #13 <Logout Admin>	61

3.1.13.4 Sequence Diagram #13 <Logout Admin>	62
3.1.14 Use Case #14 <Logout Owner>	62
3.1.14.1 Use Case Scenario#14 <Logout Owner>	62
3.1.14.2 UI Design dan Deskripsi Objek UI #14 <Logout Owner>	63
3.1.14.3 Identifikasi Object dan Tipe nya#14 <Logout Owner>	64
3.1.14.4 Sequence Diagram #14 <Logout Owner>	64
3.2 Diagram Kelas Keseluruhan	65
3.3 Perancangan Data / Basis Data	69
3.4 Perancangan Algoritma dan/atau Query	72
<b>4. Matriks Keruntutan (Requirement Traceability Matrix)</b>	<b>85</b>
<b>Lampiran</b>	<b>86</b>

## Daftar Tabel

Tabel 1. Definisi dan Istilah	14
Tabel 2. Rencana Lingkungan Implementasi	15
Tabel 3. Deskripsi setiap use case	18
Tabel 4. Primary Flow	18
Tabel 5. UI Design dan Deskripsi Objek UI	19
Tabel 6. Page Login	20
Tabel 7. Object Perancangan	20
Tabel 8. Primary Flow	22
Tabel 9. Deskripsi Objek UI	23
Tabel 10. Page Login	23
Tabel 11. Object Perancangan	24
Tabel 12. Primary Flow	25
Tabel 13. Deskripsi Objek UI	26
Tabel 14. Page Login	27
Tabel 15. Identifikasi Object dan Tipe	27
Tabel 16. Primary Flow	29
Tabel 17. Deskripsi Objek UI	30
Tabel 18. Page Registrasi	31
Tabel 19. Object Perancangan	31
Tabel 20. Use Case Description, Primary Flow	32
Tabel 21. UI Design dan Deskripsi Objek UI #5 <Kelola Hoodie>	33
Tabel 22. Page Kelola Hoodie	35
Tabel 23. OBJECT PERANCANGAN	35
Tabel 24. primary flow	36
Tabel 25. Deskripsi Objek UI	36
Tabel 26. Page Detail Pembelian	37
Tabel 27. OBJECT PERANCANGAN	38
Tabel 28. Primary Flow, Use Case Description	39
Tabel 29. Tabel Deskripsi Objek UI	40
Tabel 30. Page Laporan Penjualan	41
Tabel 31. OBJECT PERANCANGAN	41
Tabel 32. Use Case Description, Primary Flow	42
Tabel 33. Deskripsi Objek UI	43
Tabel 34. Page Order Hoodie	45
Tabel 35. Page Checkout	45
Tabel 36. Use Case Description, Primary Flow	47
Tabel 37. Deskripsi Objek UI	48
Tabel 38. Page Review	50
Tabel 39. Page addReview	50

Tabel 40. Object Perancangan	50
Tabel 41. Use Case Description, Primary Flow	52
Tabel 42. Deskripsi Objek UI	52
Tabel 43. Page Statistik Review	53
Tabel 44. Object Perancangan	53
Tabel 45. Primary Flow	55
Tabel 46. Deskripsi Objek UI	56
Tabel 47. Page Delete Customer	56
Tabel 48. Object Perancangan	57
Tabel 49. Primary Flow	58
Tabel 50. Deskripsi Objek UI	58
Tabel 51. Page Logout User	59
Tabel 52. OBJECT PERANCANGAN	60
Tabel 53. Primary Flow	61
Tabel 54. Deskripsi Objek UI	61
Tabel 55. Page Logout Admin	62
Tabel 56. OBJECT PERANCANGAN	62
Tabel 57. Primary Flow	63
Tabel 58. Deskripsi Objek UI	64
Tabel 59. OBJECT PERANCANGAN	65
Tabel 60 TABEL KELAS : User	67
Tabel 61. TABEL KELAS : Admin	67
Tabel 62. TABEL KELAS : Owner	67
Tabel 63. TABEL KELAS : Pembelian	67
Tabel 64. TABEL KELAS : Hoodie	68
Tabel 65. TABEL KELAS : Pembelian_produk	68
Tabel 66. TABEL KELAS : laporan	69
Tabel 67. TABEL KELAS : reviewHoodie	69
Tabel 68. TABEL KELAS : ViewReviewHoodie	69
Tabel 69.Matriks Keruntutan (Requirement Traceability Matrix)	87



## Daftar Gambar

Gambar 1. Tampilan Arsitektur Perangkat Lunak	16
Gambar 2. Tampilan login User website YA-HOODIE	20
Gambar 3. Tampilan login Admin website YA-HOODIE	24
Gambar 4. Tampilan Sequence Diagram Login Admin	25
Gambar 5. Tampilan login Owner website YA-HOODIE	27
Gambar 6. Tampilan Sequence Diagram Login Owner	29
Gambar 7. Tampilan Register website YA-HOODIE	31
Gambar 8. Tampilan Sequence Diagram Register	32
Gambar 9. Tampilan Kelola Hoodie website YA-HOODIE	34
Gambar 10. Tampilan Add Data hoodie pada Admin website YA-HOODIE	35
Gambar 11. Tampilan Sequence Diagram Kelola Hoodie	36
Gambar 12. Tampilan Detail Pembelian pada Admin website YA-HOODIE	38
Gambar 13. Tampilan Sequence Diagram Detail Pembelian	39
Gambar 14. Tampilan Laporan Penjualan pada Admin website YA-HOODIE	41
Gambar 15. Tampilan Sequence Diagram Laporan Penjualan	42
Gambar 16. Tampilan Order Hoodie pada website YA-HOODIE	45
Gambar 17. Tampilan Order Hoodie (Checkout) pada website YA-HOODIE	45
Gambar 18. Tampilan Bukti Pembelian pada website YA-HOODIE	46
Gambar 19. Tampilan Sequence Diagram Order Hoodie	47
Gambar 20. Tampilan Review (testimonial) pada website YA-HOODIE	50
Gambar 21. Tampilan addReview pada website YA-HOODIE	50
Gambar 22. Tampilan Sequence Diagram Review Hoodie	52
Gambar 23. Tampilan statistik review pada website YA-HOODIE	54
Gambar 24. Tampilan Sequence Diagram Statistik Review	55
Gambar 25. Tampilan Delete Customer pada website YA-HOODIE	57
Gambar 26. Tampilan Sequence Diagram Delete Customer	58
Gambar 27. Tampilan Logout User pada website YA-HOODIE	60
Gambar 28. Tampilan Sequence Diagram Logout User	61
Gambar 29. Tampilan Logout Admin pada website YA-HOODIE	63
Gambar 30. Tampilan Sequence Diagram Logout Admin	64
Gambar 31. Tampilan Logout Owner pada website YA-HOODIE	66
Gambar 32. Tampilan Sequence Diagram Logout Owner	67
Gambar 33. Tampilan Diagram Class Keseluruhan	68
Gambar 34. Tampilan Rancangan	72
Gambar 35. Tampilan Tabel Admin	72
Gambar 36. Tampilan Tabel Hoodie	73
Gambar 37. Tampilan Tabel Owner	73
Gambar 38. Tampilan Tabel Pembelian	73
Gambar 39. Tampilan Tabel Pembelian_produk	74

Gambar 40. Tampilan Tabel Review Hoodie

74

Gambar 41. Tampilan Tabel User

74

## 1. Pendahuluan

### 1.1 Tujuan Penulisan Dokumen

Dokumen ini berisi tentang Deskripsi Perancangan Perangkat Lunak ( DPPL ) untuk Ya-Hoodie. Tujuan penulisan dokumen ini untuk mendeskripsikan mengenai perangkat lunak yang akan dibangun. Dokumen ini digunakan sebagai acuan pada proses pengembangan perangkat lunak dan evaluasi pada saat pengembangan maupun diakhir pengembangan perangkat lunak.

Pengguna dari dokumen ini adalah tim pengembang dari perangkat lunak dan stakeholders yang terlibat dalam sistem ini. Dengan disusunnya dokumen DPPL ini, diharapkan pembangunan perangkat lunak akan lebih terkonsep dan tidak menimbulkan ambiguitas pada saat pembangunannya

### 1.2 Lingkup Masalah

Ya-Hoodie merupakan sebuah perangkat lunak berbasis *web* dimana nantinya dapat memudahkan proses pembelian serta kustomisasi hoodie sesuai dengan yang diinginkan oleh para calon customer.

### 1.3 Definisi dan Istilah

Berikut adalah daftar definisi dan istilah penting yang digunakan dalam dokumen DPPL ini:

Istilah, Akronim dan Singkatan	Penjelasan
SKPL	Spesifikasi Kebutuhan Perangkat Lunak Merupakan dokumen hasil analisis yang berisi spesifikasi kebutuhan user.
DPPL	Deskripsi Perancangan Perangkat Lunak. Merupakan deskripsi perancangan perangkat lunak yang akan dikembangkan
IEEE	Institute of Electrical and Electronics Engineers, merupakan standar internasional untuk pengembangan dan rancangan perangkat lunak.
DFD	Data Flow Diagram Diagram yang menggambarkan aliran data dan proses yang terjadi di dalam sistem.
User	Orang yang menggunakan website atau bisa dibilang orang yang merupakan menjadi pembeli hoodie
Owner	Orang yang merupakan pemilik dari Ya-Hoodie
Admin	Orang yang mengatur serta mengorganisir semua fitur yang ada dalam sistem aplikasi Ya-Hoodie
DBMS	Database management system Software atau tool yang digunakan untuk menyimpan data(database).
PHP	PHP adalah bahasa pemrograman yang umum dipakai dalam pembuatan dan pengembangan suatu web.

HTML	HTML adalah bahasa markup yang digunakan untuk membuat kerangka website.
Javascript	Javascript adalah bahasa pemrograman yang digunakan dalam pengembangan website agar lebih dinamis dan interaktif. JavaScript dapat meningkatkan fungsionalitas pada halaman web.
CSS	<i>Cascading Style Sheets</i> , yaitu bahasa yang digunakan untuk menentukan tampilan dan format halaman website.
Development Tools	Merupakan alat penunjang selama proses pengembangan sistem aplikasi Ya-Hoodie
Dreamweaver	Adobe Dreamweaver adalah program yang digunakan untuk membuat atau menyunting halaman web
XAMPP	XAMPP adalah software web server lokal untuk membangun website, aplikasi, hingga database secara offline.
phpmyadmin	phpmyadmin adalah aplikasi berbasis web yang digunakan untuk melakukan pengelolaan database MySQL
MySQL	MySQL adalah sebuah <i>database management system</i> (manajemen basis data) menggunakan perintah dasar SQL ( <i>Structured Query Language</i> )

*Tabel 1. Definisi dan Istilah*

#### 1.4 Referensi

Referensi yang digunakan untuk pembuatan website YA-HOODIE adalah:

1. SKPL YA-HOODIE  
Source Link : *SKPL Ya-Hoodie (2020)*
2. YA-HOODIE Online Store

YA-HOODIE Online Store adalah sebuah website yang dapat digunakan oleh customer untuk menemukan design hoodie yang terbaik sesuai keinginan dengan customer.

## 2 Perancangan Global

Deskripsi perancangan global meliputi penjelasan tentang rancangan lingkungan implementasi, deskripsi arsitektural, dan deskripsi komponen yang membangun perangkat lunak.

### 2.1 Rencana Lingkungan Implementasi

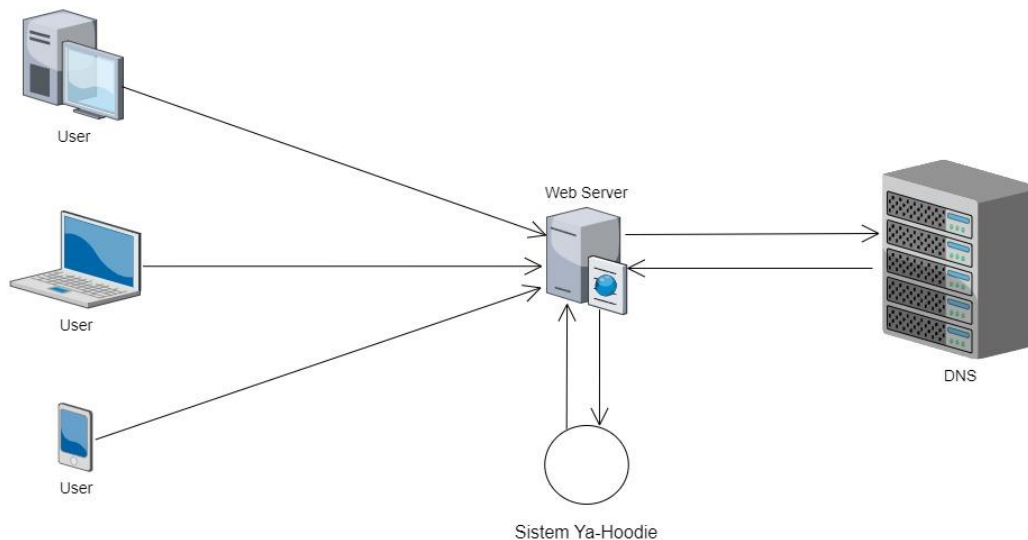
Sistem Ya-Hoodie dikembangkan dan diimplementasikan dalam lingkungan sebagai berikut:

Sistem Ya-Hoodie	Spesifikasi
Sistem Operasi	Windows 8, 10, dan 11
DBMS	MySQL
Development Tools	Dreamweaver, XAMPP, MySQL, phpmyadmin
Bahasa Pemrograman	Javascript, HTML, CSS, PHP

*Tabel 2. Rencana Lingkungan Implementasi*

### 2.2 Deskripsi Arsitektur Perangkat Lunak

Arsitektur yang digunakan dalam pembangunan aplikasi ini adalah arsitektur aplikasi berbasis web, dimana interaksi antara pengguna dengan aplikasi akan berlangsung melalui jaringan internet. Dalam arsitektur ini, semua aktivitas, baik itu interaksi antar pengguna, atau penyimpanan data pengguna atau data lainnya, akan disimpan dalam database melalui jaringan yang digunakan. Pertama user akan masuk ke web server, kemudian web server akan mengarahkan user ke aplikasi setelah melakukan pengecekan data pada database server.



*Gambar 1. Tampilan Arsitektur Perangkat Lunak*

### 3 Perancangan Rinci

#### 3.1 Realisasi Use Case

Berisi TABEL USE CASE sebagai berikut :

No	Nama Use Case	Deskripsi Use Case
#1	Login User	Use case ini dimulai saat user memasukkan username dan password yang terdaftar ke dalam form yang telah disediakan oleh sistem. Selanjutnya sistem akan melakukan validasi terhadap username dan password yang dimasukkan oleh user. Jika validasi berhasil sistem akan menyimpan informasi akun dan pindah ke halaman dashboard Ya-Hoodie user, jika validasi gagal sistem akan menampilkan notifikasi (pop up) bahwa username atau password salah.
#2	Login Admin	Use case ini dimulai saat admin memasukkan username dan password yang terdaftar ke dalam form yang telah disediakan oleh sistem. Selanjutnya sistem akan melakukan validasi terhadap username dan password yang dimasukkan oleh admin. Jika validasi berhasil sistem akan menyimpan informasi akun dan pindah ke halaman dashboard Ya-Hoodie admin, jika validasi gagal sistem akan menampilkan notifikasi (pop up) bahwa username atau password salah.
#3	Login Owner	Use case ini dimulai saat owner memasukkan username dan password yang terdaftar ke dalam form yang telah disediakan oleh sistem. Selanjutnya sistem akan melakukan validasi terhadap username dan password yang dimasukkan oleh owner. Jika validasi berhasil sistem akan menyimpan informasi akun dan pindah ke halaman dashboard Ya-Hoodie owner, jika validasi gagal sistem akan menampilkan notifikasi (pop up) bahwa username atau password salah.
#4	Register	Use case ini dimulai saat user belum memiliki akun untuk login, yang dimana tombol register ini ada berada pada login customer / user . User harus memasukkan full name, username, dan password. Setelah itu sistem akan menyimpan registrasi user untuk login ke dalam sistem dashboard Ya-Hoodie user.
#5	Kelola Hoodie	Use case ini dimulai saat admin sudah dalam keadaan login lalu admin dapat menekan tombol hoodie pada navbar dashboard admin. Setelah itu admin dapat mengelola hoodie yang dimana dapat mengupdate dan delete data hoodie. Dan juga admin dapat meng add data hoodie.
#6	Detail Pembelian	Use case ini dimulai saat admin sudah dalam kondisi login dan menekan tombol sales report pada navbar dashboard admin. Dan admin dapat melihat detail pembelian dari user.

#7	Laporan Penjualan	Use case ini dimulai saat owner sudah dalam kondisi login lalu owner dapat melihat laporan penjualan yang terdapat pada navbar dashboard owner, yang dimana pada halaman laporan penjualan tersebut owner dapat melihat laporan penjualan sesuai dengan tanggal mulai hingga tanggal akhir untuk melihat laporannya.
#8	Order Hoodie	Use case ini dimulai saat user sudah dalam keadaan kondisi login lalu user dapat melakukan pembelian hoodie dengan meng “add to cart”, lalu data pembelian akan disimpan pada keranjang, user dapat menghapus data, melanjutkan pembelian hoodie, dan dapat meng checkout pembelian hoodie.
#9	Review Hoodie	Use case ini dimulai saat user sudah dalam keadaan kondisi login lalu user dapat melakukan review dengan menekan tombol pages, lalu review. lalu sistem akan pindah pada halaman review, yang dimana user dapat menginputkan review atau ulasan dengan memasukkan bintang dengan skala 1-5, username, dan juga komentar. Lalu sistem akan menyimpan review tersebut ke dalam database.
#10	Statistik Review	Use case ini dimulai saat owner sudah dalam keadaan kondisi login. Pada dashboard owner, jika owner ingin melihat statistik ulasan yang telah diberikan pengguna dapat dilakukan dengan menekan tombol “Review Statistic” pada navbar yang terletak di samping, pada laman ini owner dapat melihat semua ulasan yang diberikan pengguna kepada website YA-HOODIE seperti rating, komentar dan rata-rata rating serta username user yang memberikan ulasan.
#11	Delete Customer	Use case ini dimulai saat admin sudah dalam keadaan kondisi login . Pada halaman dashboard admin terdapat tombol customer pada navbar , jika admin ingin menghapus data user bisa dilakukan pada pilihan “Delete” dengan menekan “Customers” pada navbar di sebelah kiri lalu menekan tombol “Delete”. Disini admin bisa menghapus semua data user.
#12	Logout User	Use case ini dimulai saat user sudah dalam keadaan kondisi login dan user menekan tombol keluar yang tersedia pada top bar. Selanjutnya sistem akan melakukan penghapusan data akun yang disimpan saat login.
#13	Logout Admin	Use case ini dimulai saat admin sudah dalam keadaan kondisi login dan user menekan tombol keluar yang tersedia pada top bar. Selanjutnya sistem akan melakukan penghapusan data akun yang disimpan saat login.
#14	Logout Owner	Use case ini dimulai saat owner sudah dalam keadaan kondisi login dan user menekan tombol keluar yang tersedia pada top bar. Selanjutnya sistem akan

		melakukan penghapusan data akun yang disimpan saat login.
--	--	---

*Tabel 3. Deskripsi setiap use case*

### 3.1.1 Use Case #1 <Login User>

#### 3.1.1.1 Use Case Scenario #1 <Login User>

Skenario Use Case #1 :

- i. Pre-Condition  
User belum mengisi username dan password.
- ii. Use Case Description
  - a. Primary Flow

Aktor	Sistem
1. User membuka menu	
	2. Sistem menampilkan halaman Login
3. User dapat mengisi username dan password yang dimiliki	
	4. Sistem akan memvalidasi login dari user
	5. Sistem akan berpindah menampilkan halaman awal dari dashboard utama untuk user

*Tabel 4. Primary Flow*

- b. Alternative Flow
  1. User menekan tombol log in.
  2. User mengetikkan username
  3. User mengetikkan password.
  4. User menekan tombol masuk.
  5. User mendapatkan pesan error karena email atau password salah atau tidak terdaftar
- iii. Post-Condition  
User berhasil masuk dengan identitas yang sesuai.



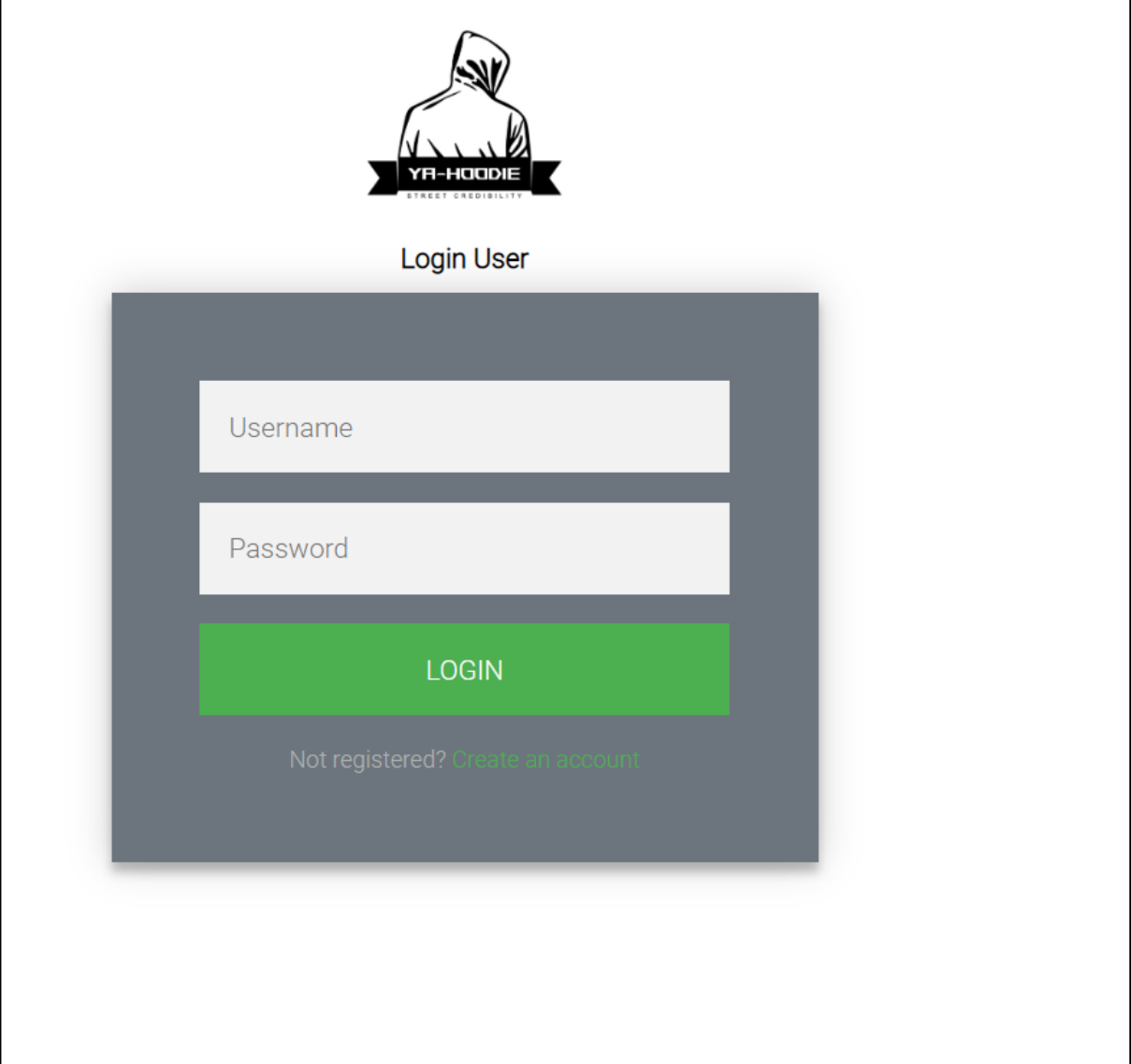
### 3.1.1.2 UI Design dan Deskripsi Objek UI #1 <Login User>

Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case  
Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detailnya.

ID. LAYAR	NAMA LAYAR	DESKRIPSI
	Page Login	Halaman yang menampilkan form untuk melakukan proses login

Tabel 5. UI Design dan Deskripsi Objek UI

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detail  
Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}



Gambar 2. Tampilan login User website YA-HOODIE

*Page LOGIN*

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
<i>TextBox1</i>	<i>Textbox</i>	<i>Username</i>	<i>User memasukkan username</i>
<i>TextBox2</i>	<i>Textbox</i>	<i>Password</i>	<i>User memasukkan password</i>
<i>Button 1</i>	<i>Button</i>	<i>Login</i>	<i>Jika diklik akan mengaktifkan fungsi validasi untuk login</i>

*Tabel 6. Page Login*

**3.1.1.3 Identifikasi Object dan Tipe nya #1 <Login User>**

*Identifikasi object yang terkait dengan use case tersebut.*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.*

*Gunakan tabel di bawah:*

**TABEL OBJECT PERANCANGAN**

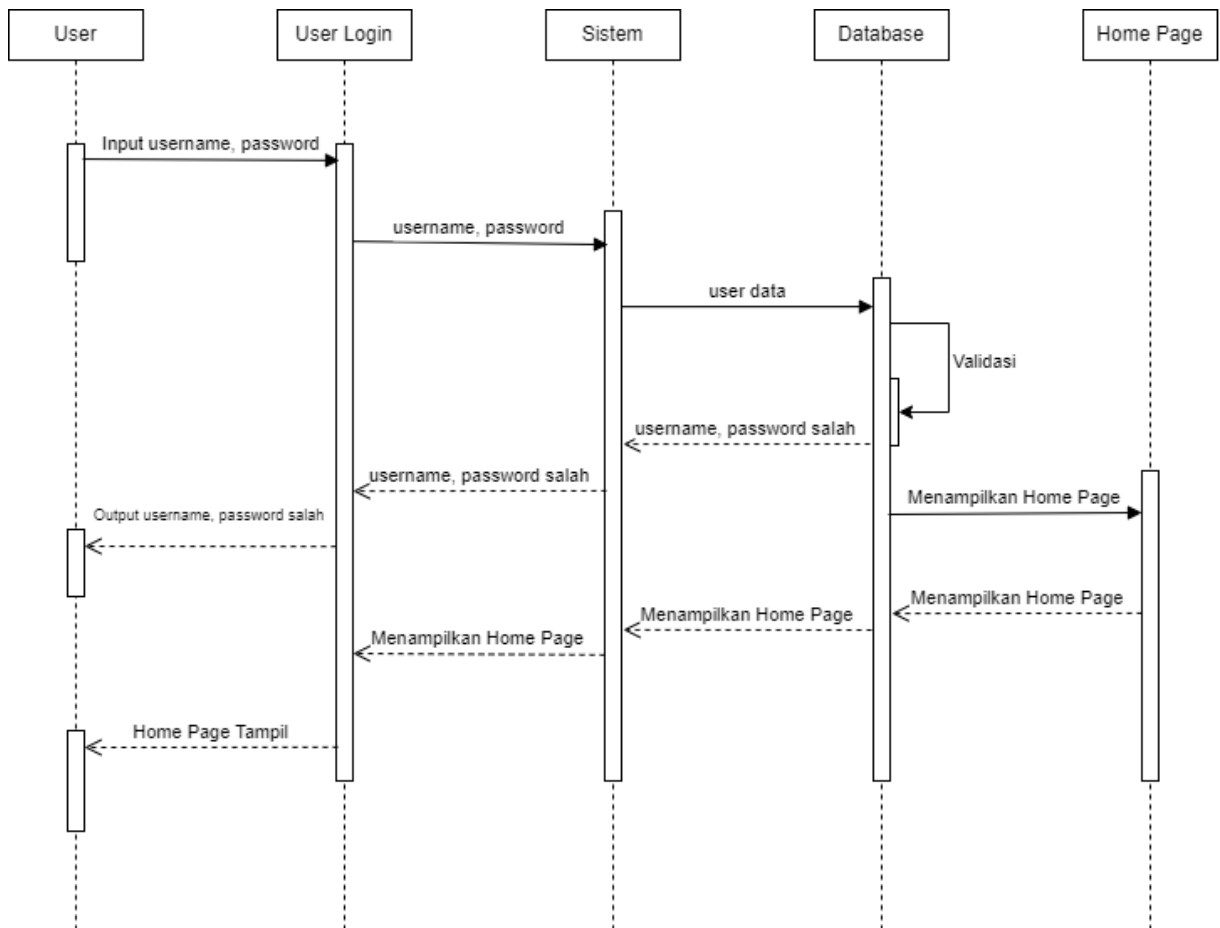
<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
	<i>User login</i>	

*Tabel 7. Object Perancangan*

*\*Tipe kelas diisi dengan Boundary(Interface), Entity(Database), Controller*

### 3.1.1.4 Sequence Diagram #1 <Login User>

Buatlah diagram sequence untuk masing - masing use case sesuai skenario.  
Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.



### 3.1.2 Use Case #2 <Login Admin>

#### 3.1.2.1 Use Case Scenario #2 <Login Admin>

Skenario Use Case #2 :

- i. Pre-Condition  
Admin belum mengisi username dan password.
- ii. Use Case Description
  - a. Primary Flow

Aktor	Sistem
1. Admin membuka menu	
	2. Sistem menampilkan halaman Login
3. Admin dapat mengisi username dan password yang dimiliki	
	4. Sistem akan memvalidasi login dari admin
	5. Sistem akan berpindah menampilkan halaman awal dari dashboard utama untuk admin

Tabel 8. Primary Flow

- b. Alternative Flow
  1. Admin menekan tombol log in.
  2. Admin mengetikkan username
  3. Admin mengetikkan password.
  4. Admin menekan tombol masuk.
  5. Admin mendapatkan pesan error karena email atau password salah atau tidak terdaftar

- iii. Post-Condition  
Admin berhasil masuk dengan identitas yang sesuai.

#### 3.1.2.2 UI Design dan Deskripsi Objek UI #2 <Login Admin>

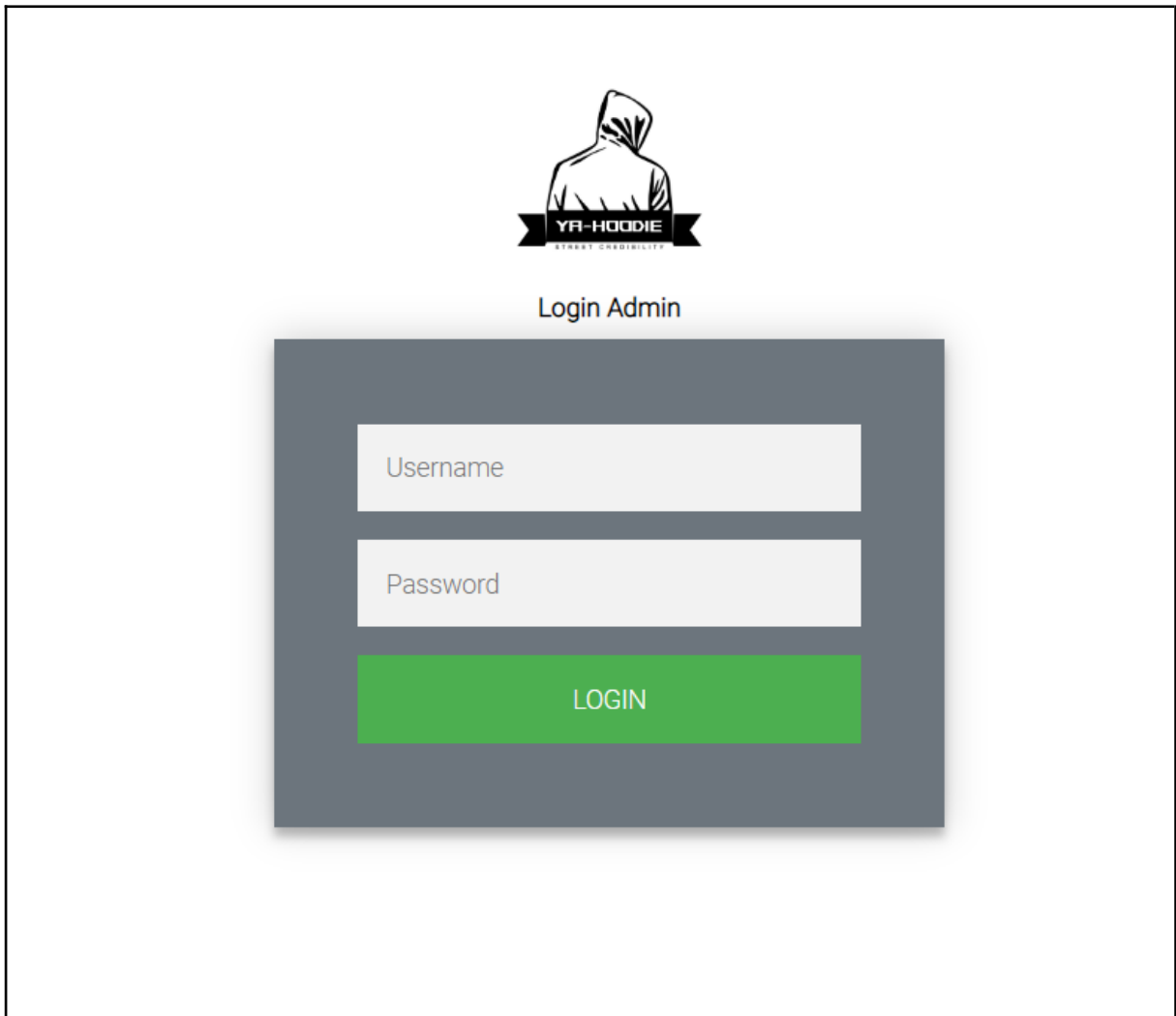
Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case  
Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detailnya.

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
	Page Login	Halaman yang menampilkan form untuk melakukan proses login

Tabel 9. Deskripsi Objek UI

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detail  
Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}



Gambar 3. Tampilan login Admin website YA-HOODIE

#### Page LOGIN

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
<i>TextBox1</i>	<i>Textbox</i>	<i>Username</i>	<i>Admin memasukkan username</i>
<i>TextBox2</i>	<i>Textbox</i>	<i>Password</i>	<i>Admin memasukkan password</i>
<i>Button 1</i>	<i>Button</i>	<i>Login</i>	<i>Jika diklik akan mengaktifkan fungsi validasi untuk login</i>

Tabel 10. Page Login

### 3.1.2.3 Identifikasi Object dan Tipe nya #2 <Login Admin>

Identifikasi object yang terkait dengan use case tersebut.

<b>Prodi S1 Informatika Tel-U</b>	<b>DPPL-XXX</b>	<b>Halaman 21 dari 86</b>
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.  
Gunakan tabel di bawah:

#### TABEL OBJECT PERANCANGAN

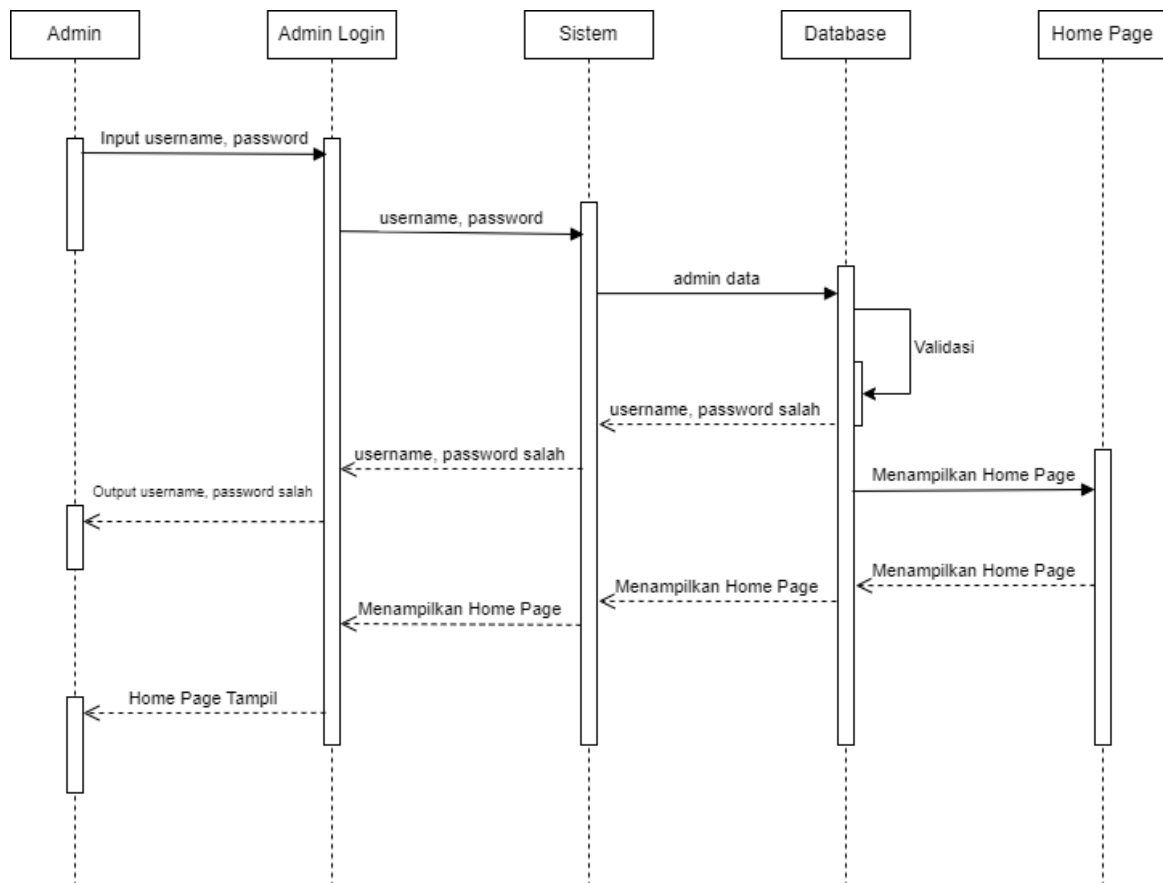
No	Nama Object Baru	Jenis / Tipe Kelas
	<i>Admin Login</i>	

Tabel 11. Object Perancangan

\*Tipe kelas diisi dengan Boundary(Interface), Entity(Database), Controller

#### 3.1.2.4 Sequence Diagram #2 <Login Admin>

Buatlah diagram sequence untuk masing - masing use case sesuai skenario.  
Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.



Gambar 4. Tampilan Sequence Diagram Login Admin

### 3.1.3 Use Case #3 <Login Owner>

#### 3.1.3.1 Use Case Scenario #3 <Login Owner>

Skenario Use Case #3 :

- i. Pre-Condition  
Owner belum mengisi username dan password.
- ii. Use Case Description
  - a. Primary Flow

Aktor	Sistem
1. Owner membuka menu	
	2. Sistem menampilkan halaman Login
3. Owner dapat mengisi username dan password yang dimiliki	
	4. Sistem akan memvalidasi login dari owner
	5. Sistem akan berpindah menampilkan halaman awal dari dashboard utama untuk owner

*Tabel 12. Primary Flow*

- b. Alternative Flow
  - 1. User menekan tombol log in.
  - 2. User mengetikkan username
  - 3. User mengetikkan password.
  - 4. User menekan tombol masuk.
  - 5. User mendapatkan pesan error karena email atau password salah atau tidak terdaftar
- iii. Post-Condition  
User berhasil masuk dengan identitas yang sesuai.

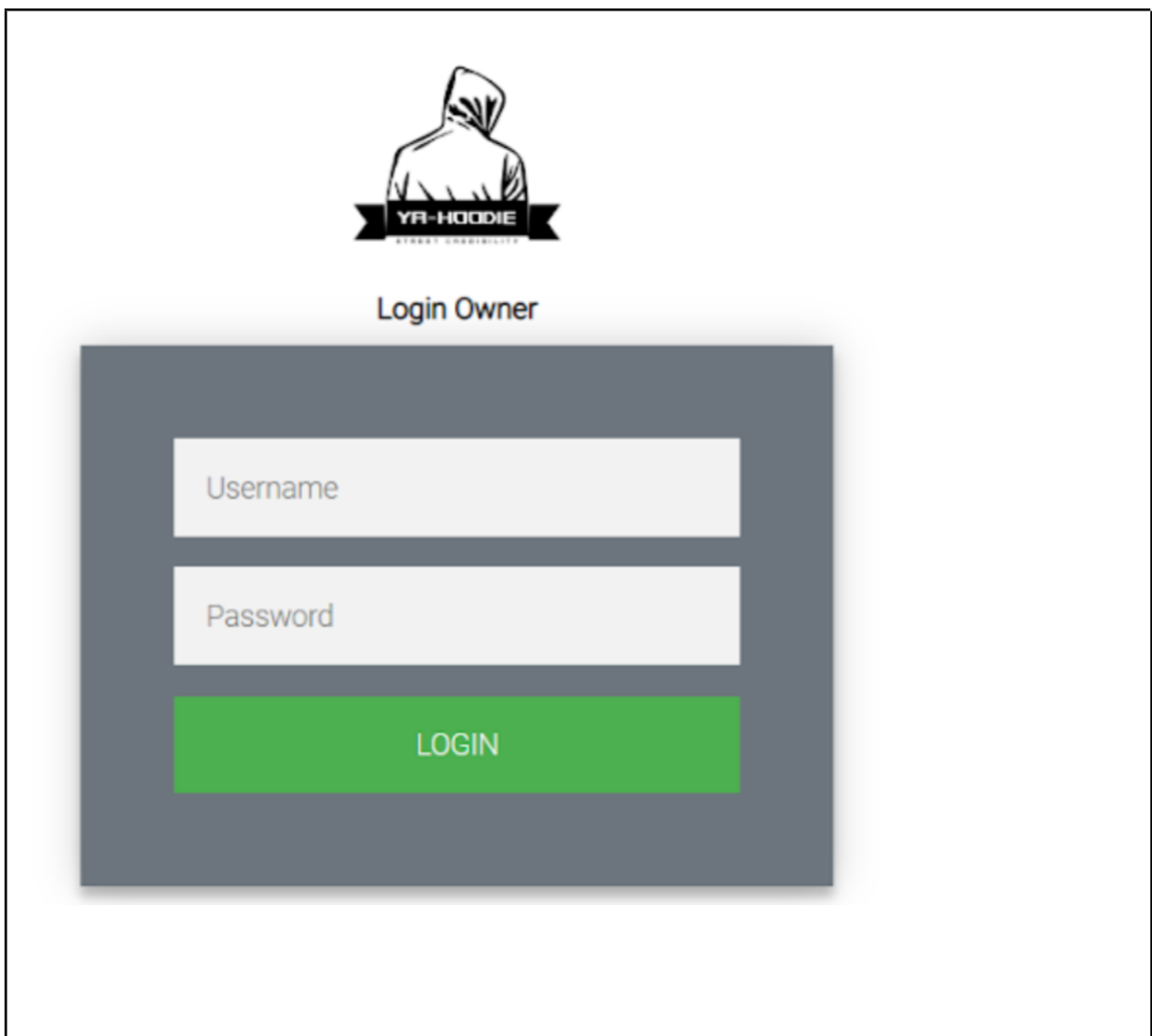
### 3.1.3.2 UI Design dan Deskripsi Objek UI #3 <Login Owner>

Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case  
Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detailnya.

ID. LAYAR	NAMA LAYAR	DESKRIPSI
	Page Login	Halaman yang menampilkan form untuk melakukan proses login

Tabel 13. Deskripsi Objek UI

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detail  
Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}



Gambar 5. Tampilan login Owner website YA-HOODIE



*Page LOGIN*

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
<i>TextBox1</i>	<i>Textbox</i>	<i>Username</i>	<i>Owner memasukkan username</i>
<i>TextBox2</i>	<i>Textbox</i>	<i>Password</i>	<i>Owner memasukkan password</i>
<i>Button 1</i>	<i>Button</i>	<i>Login</i>	<i>Jika diklik akan mengaktifkan fungsi validasi untuk login</i>

*Tabel 14. Page Login*

**3.1.3.3 Identifikasi Object dan Tipe nya #3 <Login Owner>**

*Identifikasi object yang terkait dengan use case tersebut.*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.*

*Gunakan tabel di bawah:*

**TABEL OBJECT PERANCANGAN**

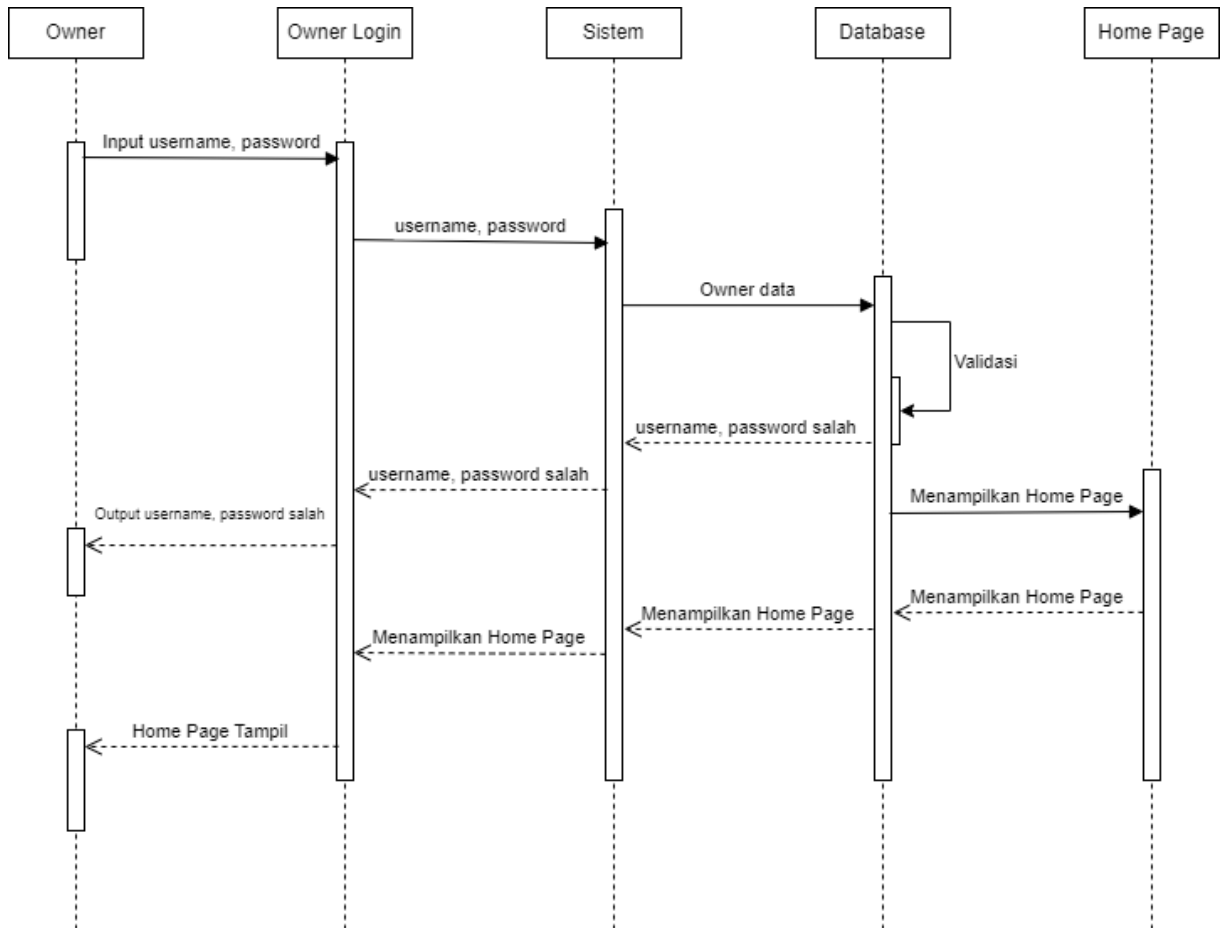
<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
	<i>Owner Login</i>	

*Tabel 15. Identifikasi Object dan Tipe*

*\*Tipe kelas diisi dengan Boundary(Interface), Entity(Database), Controller*

### 3.1.3.4 Sequence Diagram #3 <Login Owner>

Buatlah diagram sequence untuk masing - masing use case sesuai skenario.  
Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.



Gambar 6. Tampilan Sequence Diagram Login Owner

### 3.1.4 Use Case #4 <Register>

*Skenario Use Case #4*

*Primary Flow*

*Alternate Flow*

#### 3.1.4.1 Use Case Scenario #4 <Register>

Skenario Use Case #4 :

- i. Pre-Condition  
User belum terdaftar
- ii. Use Case Description
  - a. Primary Flow

Aktor	Sistem
1. User membuka menu registrasi yang terdapat pada login (jika belum mempunyai akun).	
	2. Sistem menampilkan halaman Registrasi.
3. User dapat mengisi form registrasi dan mengirimkannya pada sistem.	
	4. Sistem akan menyimpan data pembeli.

*Tabel 16. Primary Flow*

b. Alternative Flow

1. User memasukkan full name untuk register
2. User memasukkan username
3. User memasukkan password
4. User tidak berhasil / gagal melakukan register dikarenakan salah dalam penginputan username atau password yang tidak sesuai

iii. Post-Condition

User berhasil melakukan registrasi akun

#### 3.1.4.2 UI Design dan Deskripsi Objek UI #4 <Register>

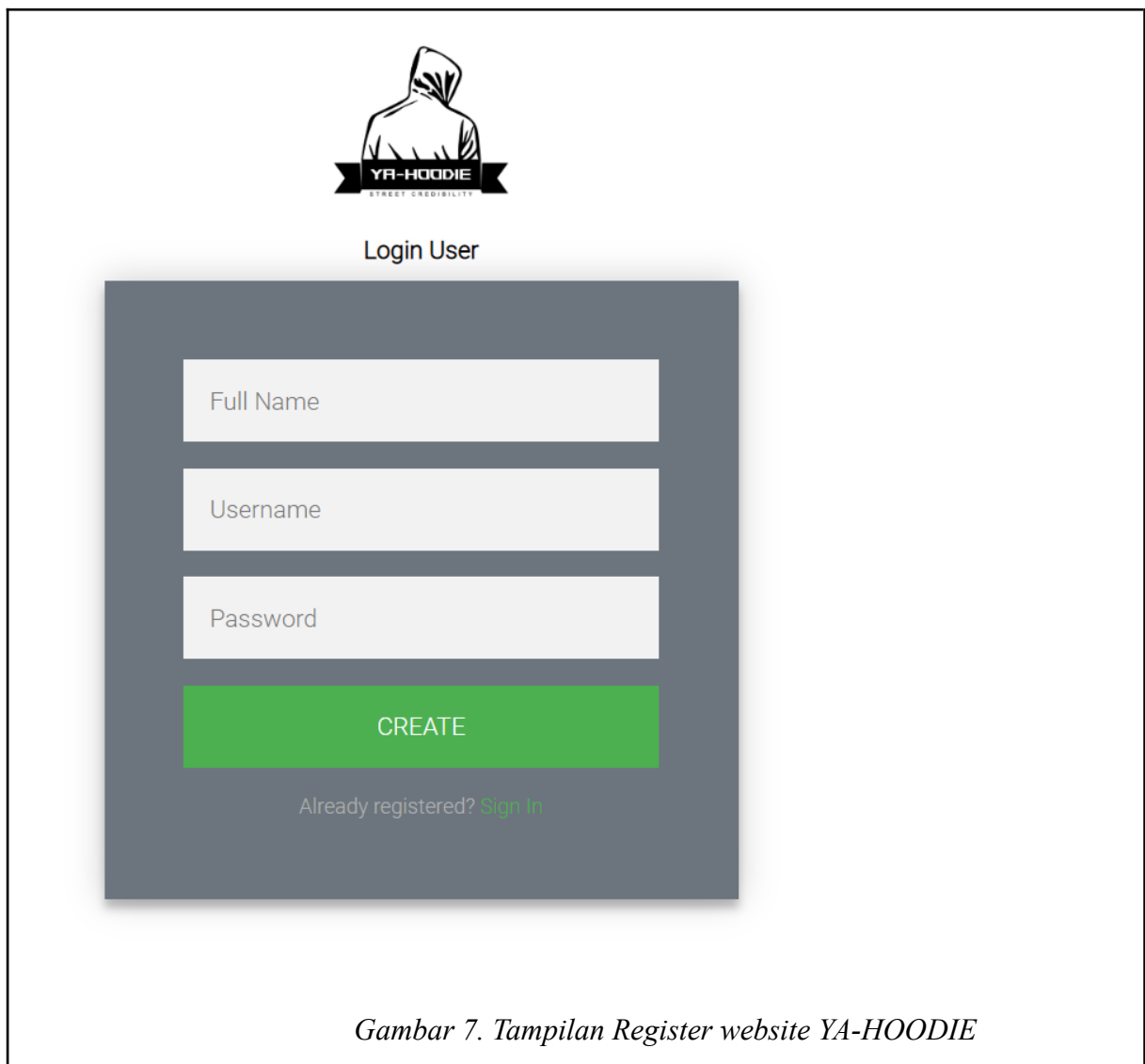
*Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case*

Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detilnya.

ID. LAYAR	NAMA LAYAR	DESKRIPSI
	Page Registrasi	Menampilkan laman form untuk mengisikan registrasi atau pembuatan akun baru dari user.

Tabel 17. Deskripsi Objek UI

UNTUK MASING – MASING ANTARMUKA / PAGE dibuatkan spesifikasi detil Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}



Gambar 7. Tampilan Register website YA-HOODIE

Page REGISTRASI

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
<i>TextBox1</i>	<i>TextBox</i>	<i>Fullname</i>	<i>User memasukkan fullname</i>
<i>TextBox2</i>	<i>Textbox</i>	<i>Username</i>	<i>User memasukkan username</i>
<i>TextBox3</i>	<i>Textbox</i>	<i>Password</i>	<i>User memasukkan password</i>
<i>Button 1</i>	<i>Button</i>	<i>Login</i>	<i>Jika diklik akan mengaktifkan fungsi validasi untuk login</i>

*Tabel 18. Page Registrasi*

### 3.1.4.3 Identifikasi Object Baru & Tipe Kelas #4 <Register>

*Identifikasi object yang terkait dengan use case tersebut.*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.*

*Gunakan tabel di bawah:*

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
	<i>Registration Page</i>	<i>interface</i>
	<i>Database</i>	<i>Entity</i>

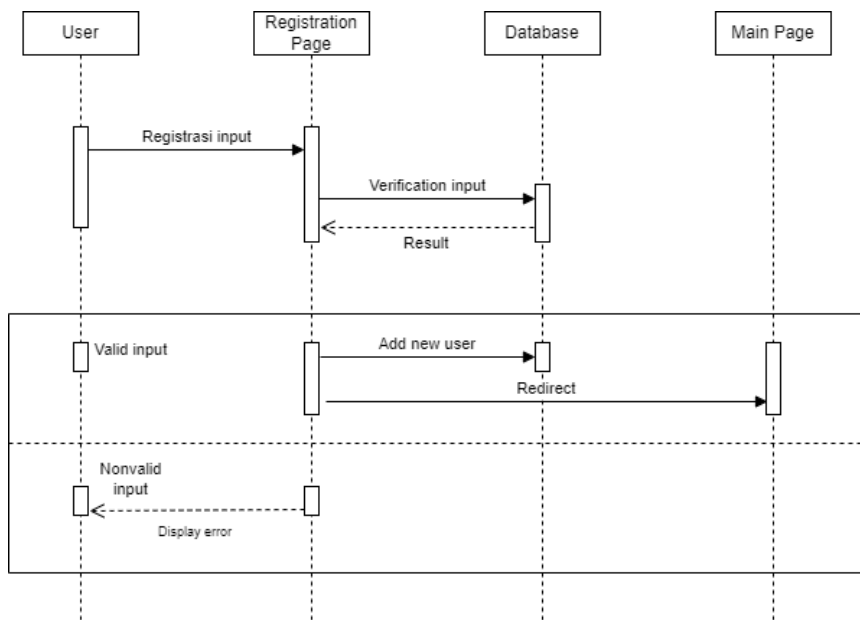
*Tabel 19. Object Perancangan*

*\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller*

### 3.1.4.4 Sequence Diagram #4 <Register>

*Buatlah diagram sequence untuk masing - masing use case sesuai skenario.*

*Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.*



*Gambar 8. Tampilan Sequence Diagram Register*

### 3.1.5 Use Case #5 <Kelola Hoodie>

#### 3.1.5.1 Use Case Scenario #5 <Kelola Hoodie>

Skenario Use Case #5 :

- i. Pre-Condition  
Admin sudah melakukan login
- ii. Use Case Description
  - a. Primary Flow

Aktor	Sistem
1. Admin masuk ke halaman kelola hoodie khusus admin	
	2. Sistem menampilkan halaman Kelola hoodie dan sistem akan memberi pilihan untuk admin
3. Admin dapat mengupdate data hoodie	
	4. Sistem akan mengupdatenya
5. Admin dapat mendelete data hoodie	
	6. Sistem akan mendeletenya
7. Admin dapat meng add data hoodie	
	8. Sistem akan meng add data hoodie sesuai dengan apa yang admin lakukan dan menyimpannya ke dalam database.

*Tabel 20. Use Case Description, Primary Flow*

#### b. Alternative Flow

1. Admin menekan tombol Kelola hoodie, maka
  - Sistem akan memberikan pilihan update stock hoodie
2. b. Admin menekan tombol update stock, maka
  - Sistem akan menampilkan halaman untuk mengupdate jumlah stock hoodie yang dipilih
3. Admin menekan tombol input hoodie, maka
  - Sistem akan menampilkan halaman untuk memasukkan data hoodie baru
4. Admin menekan tombol delete hoodie, maka
  - Sistem akan menampilkan halaman untuk menghapus data hoodie lama

5. Admin menekan tombol home, maka
    - Sistem akan mengembalikan ke halaman utama
  6. Admin menekan tombol akun, maka
    - Sistem akan menampilkan halaman akun admin
- iii. Post-Condition  
Data hoodie telah ter-update

### 3.1.5.2 UI Design dan Deskripsi Objek UI #5 <Kelola Hoodie>

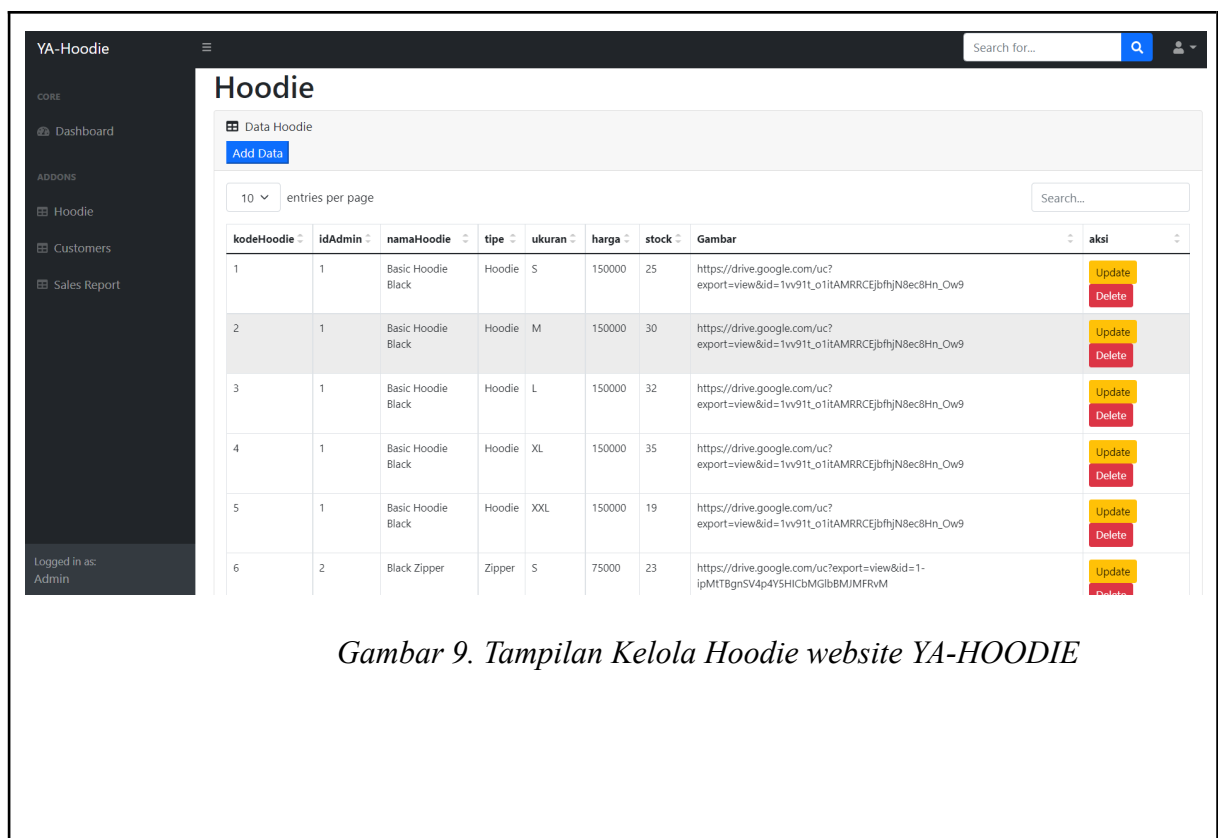
Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case  
Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detailnya.

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
	Page Kelola Hoodie	Menampilkan kelola hoodie pada dashboard admin yang dimana bertujuan untuk mengelola, mengupdate, menambahkan data atau pun menghapus data hoodie.

Tabel 21. UI Design dan Deskripsi Objek UI #5 <Kelola Hoodie>

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detail  
Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}



Gambar 9. Tampilan Kelola Hoodie website YA-HOODIE

[Back to Hoodie](#)

kodeHoodie

idAdmin

namaHoodie

Tipe

Pilih Tipe

Ukuran

Pilih Ukuran

Harga

Stock

Gambar

Add Data

Id_Objek	JENIS	LABEL*	Keterangan**
Button1	Button	Update	Jika diklik, akan mengaktifkan Function update data hoodie
Button 2	Button	Delete	Jika diklik akan mengaktifkan penghapusan data hoodie
Button 3	Button	Add data	Jika diklik akan mengaktifkan penambahan data hoodie , dan mengeluarkan laman add data
TextBox1	TextBox	KodeHoodie	Admin memasukkan kode hoodie
TextBox2	TextBox	idAdmin	Admin memasukkan idAdmin
TextBox3	TextBox	namaHoodie	Admin memasukkan nama hoodie
TextBox4	TextBox	Harga	Admin memasukkan harga
TextBox5	TextBox	Stock	Admin memasukkan stock
TextBox6	TextBox	Gambar	Admin memasukkan gambar
RadioButton1	RadioButton	Tipe	Admin memasukkan tipe hoodie
RadioButton2	RadioButton	Ukuran	Admin memasukkan ukuran hoodie



Tabel 22. Page Kelola Hoodie

### 3.1.5.3 Identifikasi Object dan Tipe nya #5 <Kelola Hoodie>

Identifikasi object yang terkait dengan use case tersebut.

Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.

Gunakan tabel di bawah:

No	Nama Object Baru	Jenis / Tipe Kelas
	TBD	TBD

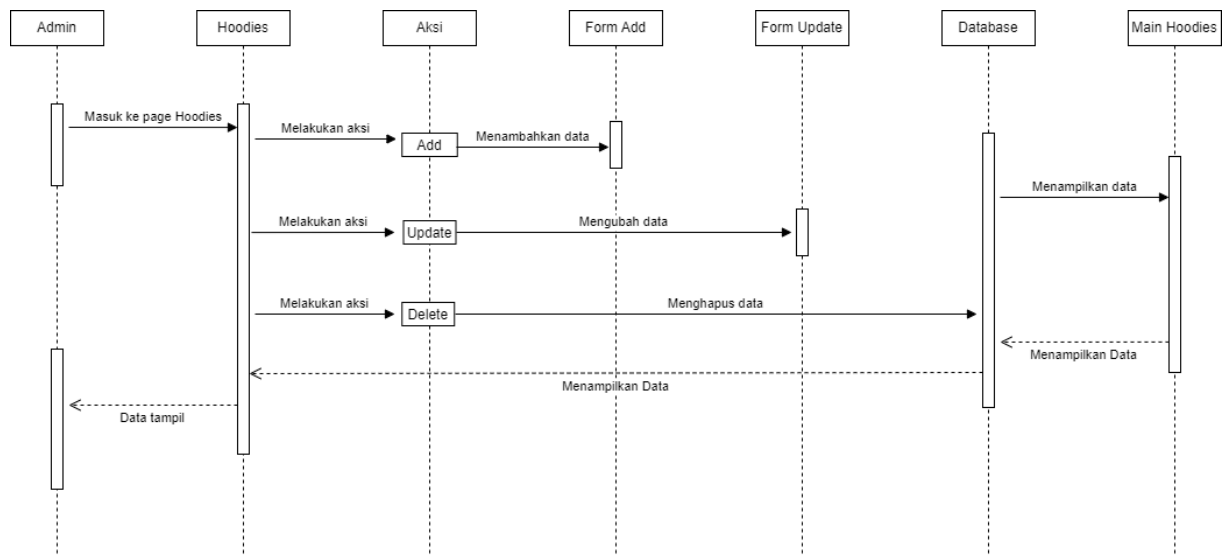
Tabel 23. OBJECT PERANCANGAN

\*Tipe kelas diisi dengan Boundary(Interface), Entity(Database), Controller

### 3.1.5.4 Diagram #5 <Kelola Hoodie>

Buatlah diagram sequence untuk masing - masing use case sesuai skenario.

Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.



Gambar 11. Tampilan Sequence Diagram Kelola Hoodie

### 3.1.6 Use Case #6 <Detail Pembelian >

#### 3.1.6.1 Use Case Scenario #6 <Detail Pembelian >

Skenario Use Case #6 :

i. Pre-Condition

Admin telah melakukan login.

ii. Use Case Description

a. Primary Flow

Aktor	Sistem
1. Admin mengakses laman detail pembelian.	
	2. Sistem menampilkan detail pembelian user

*Tabel 24. primary flow*

b. Alternative Flow

1. Admin dapat mengakses laman detail pembelian , maka
  - sistem akan menampilkan detail pembelian user secara keseluruhan.
2. Admin dapat menekan tombol “detail”
  - Sistem akan menampilkan detail pembelian user dengan salah satu user.

iii. Post-Condition

Admin mengakses laman detail pembelian, dan melihat detail pembelian user.

#### 3.1.6.2 UI Design dan Deskripsi Objek UI #6 <Detail Pembelian >

*Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detailnya*

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
	<i>Page Detail Pembelian</i>	<i>menampilkan sales report atau detail dari pembelian customer yang bisa dipantau melalui admin saja.</i>

*Tabel 25. Deskripsi Objek UI*

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detail  
*Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}*

CODE

Dashboard

ADDONS

Hoodie

Customers

Sales Report

### Tables Sales Reports

10 ▾

entries per page

Search...

NO	Nama	Tanggal	Total	Aksi
1	Yaya Aja	2022-12-31	Rp. 150,000	<a href="#">Detail</a>
2	Vanna Acosta	2022-12-31	Rp. 150,000	<a href="#">Detail</a>
3	Drake Welch	2022-12-31	Rp. 550,000	<a href="#">Detail</a>
4	Yuri Mccar	2023-01-01	Rp. 500,000	<a href="#">Detail</a>
5	Damian Taylor	2023-01-01	Rp. 715,000	<a href="#">Detail</a>
6	Fachrul Febriana	2023-01-01	Rp. 620,000	<a href="#">Detail</a>
7	Jamalia Navarro	2023-01-01	Rp. 300,000	<a href="#">Detail</a>
8	Fachrul Febriana	2023-01-01	Rp. 150,000	<a href="#">Detail</a>
9	Gayuh Cantik	2023-01-02	Rp. 40,000	<a href="#">Detail</a>
10	Gayuh Cantik	2023-01-02	Rp. 75,000	<a href="#">Detail</a>

## Detail Pembelian

### Yaya Aja

Username: yaya

Tanggal Pembelian: 2022-12-31

Total: 150000

No	Nama Hoodie	Ukuran	Harga	Jumlah	Subtotal
1	Basic Hoodie Black S		Rp. 150,000	1	Rp. 150,000
2	Basic Hoodie Black M		Rp. 150,000	1	Rp. 150,000

*Gambar 12. Tampilan Detail Pembelian pada Admin website YA-HOODIE*

Id_Objek	JENIS	LABEL*	Keterangan**
Button1	Button	Detail	Jika diklik, akan mengaktifkan Function detail pembelian pada laman dashboard sales report dan sistem akan berpindah pada detail pembelian dari setiap customer yang dipilih.

Tabel 26. Page Detail Pembelian

### 3.1.6.3 Identifikasi Object dan Tipe nya #6 <Detail Pembelian >

Identifikasi object yang terkait dengan use case tersebut.

Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.

Gunakan tabel di bawah:

**TABEL OBJECT PERANCANGAN**

No	Nama Object Baru	Jenis / Tipe Kelas
	<i>TBD</i>	<i>TBD</i>

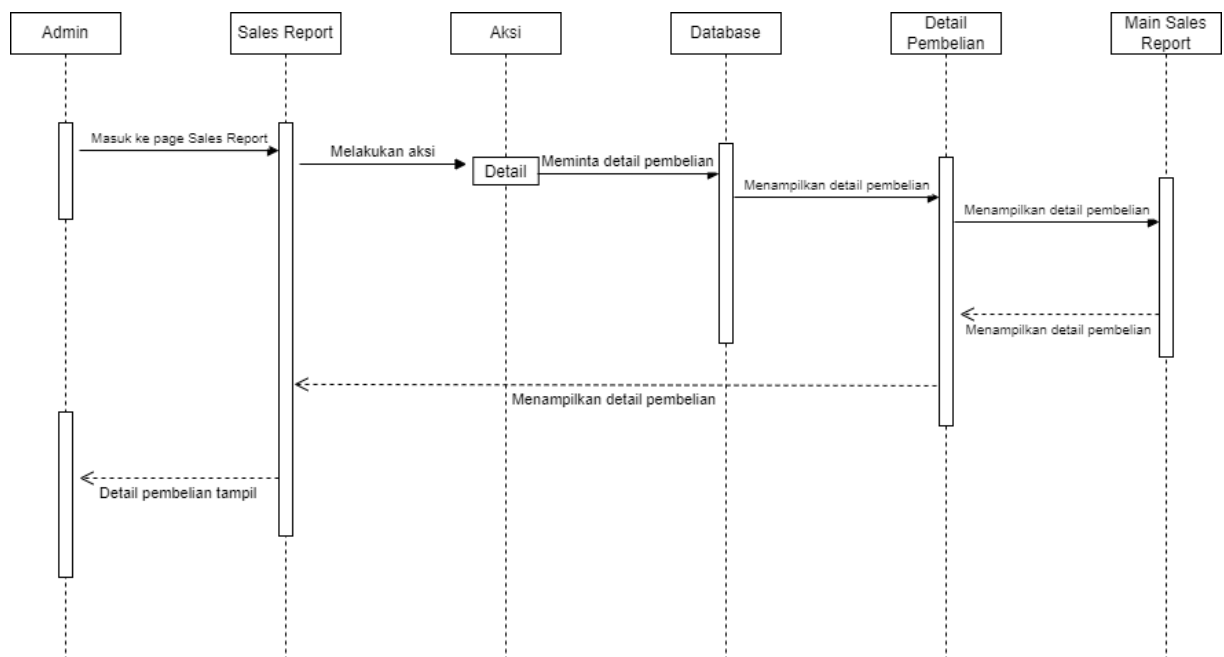
*Tabel 27. OBJECT PERANCANGAN*

*\*Tipe kelas diisi dengan Boundary(Interface), Entity(Database), Controller*

### 3.1.6.4 Sequence Diagram #6 <Detail Pembelian >

Buatlah diagram sequence untuk masing - masing use case sesuai skenario.

Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.



*Gambar 13. Tampilan Sequence Diagram Detail Pembelian*

### 3.1.7 Use Case #7 <Laporan Penjualan >

#### 3.1.7.1 Use Case Scenario #7 <Laporan Penjualan >

Skenario Use Case #7 :

- i. Pre-Condition  
Owner sudah melakukan login
- ii. Use Case Description
  - a. Primary Flow

Aktor	Sistem
1. Owner masuk dalam laman “Sales Report”	
	2. Sistem menampilkan laman laporan penjualan.
3. Owner memasukkan data tanggal mulai dan tanggal selesai laporan yang ingin ditampilkan oleh sistem	
	4. Sistem menampilkan laporan penjualan berdasarkan periode tanggal yang diberikan owner

*Tabel 28. Primary Flow, Use Case Description*

b. Alternative Flow

1. Owner menekan tombol laporan penjualan ingin ditampilkan, maka
    - Sistem akan menampilkan laporan penjualan, lalu owner dapat mendownload laporan penjualan tersebut (Basic flow Langkah1)
  2. Owner menekan tombol home, maka:
    - Sistem akan menampilkan halaman home dari website.
  3. Owner menekan tombol Account, maka:
    - Sistem akan menampilkan halaman akun dari Owner.
- iii. Post-Condition  
Owner telah memilih menu untuk menampilkan laporan penjualan, dan masuk menuju halaman yang dituju

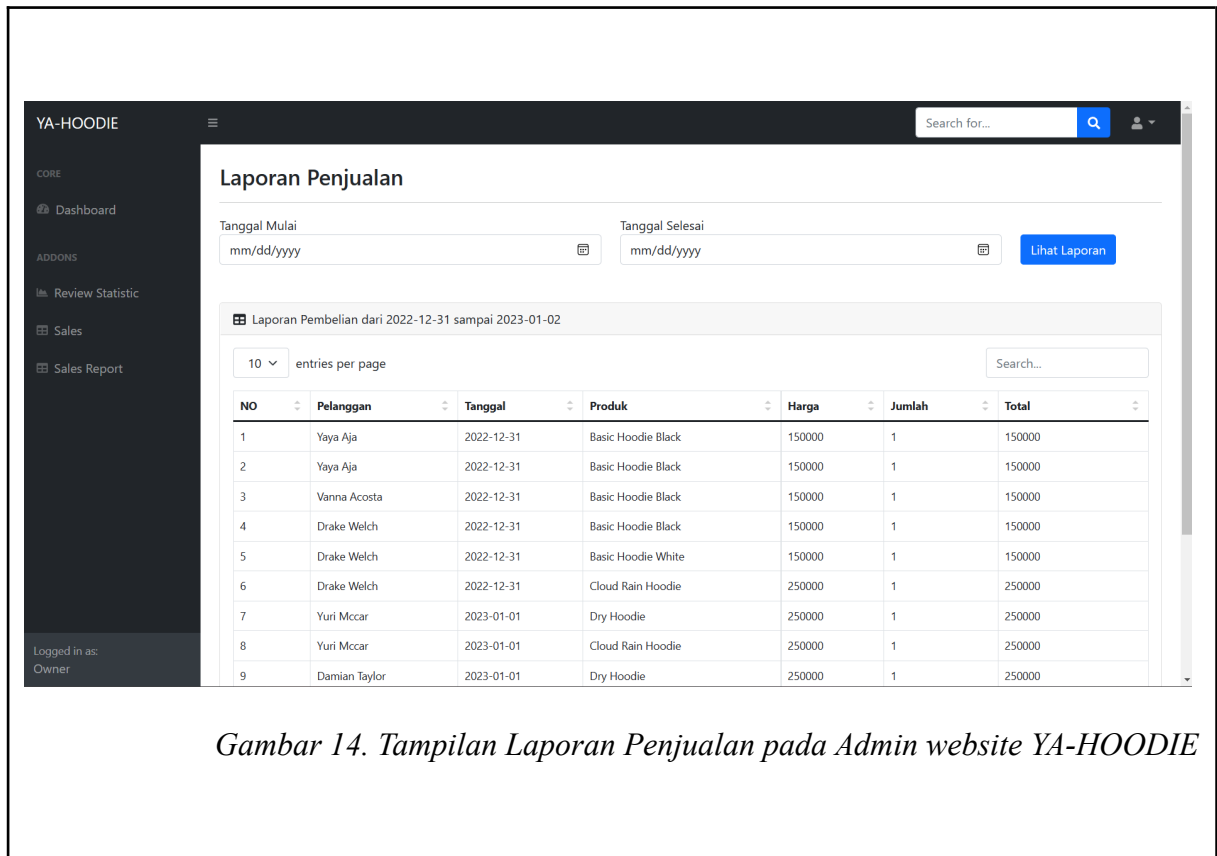
### 3.1.7.2 UI Design dan Deskripsi Objek UI #7 <Laporan Penjualan >

Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detailnya.

ID. LAYAR	NAMA LAYAR	DESKRIPSI
	Page Laporan Penjualan	Menampilkan laporan penjualan pada dashboard owner yang dimana owner dapat melihat laporan tersebut sesuai tanggal mulai dan tanggal selesai yang diinginkan

Tabel 29. Tabel Deskripsi Objek UI

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detail Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}



Gambar 14. Tampilan Laporan Penjualan pada Admin website YA-HOODIE

Id_Objek	JENIS	LABEL*	Keterangan**
Button1	Button	Lihat Laporan	Jika diklik, akan mengaktifkan Function menampilkan laporan sesuai dengan tanggal mulai hingga tanggal selesai
jXDatePicker1	jXDatePicker	tanggal mulai	Owner memilih tanggal mulai

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
<i>jXDatePicker1</i>	<i>jXDatePicker</i>	<i>tanggal selesai</i>	<i>Owner memilih tanggal selesai</i>

*Tabel 30. Page Laporan Penjualan*

### 3.1.7.3 Identifikasi Object dan Tipe nya #7 <Laporan Penjualan >

Identifikasi object yang terkait dengan use case tersebut.

Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.

Gunakan tabel di bawah:

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
	<i>TBD</i>	<i>TBD</i>

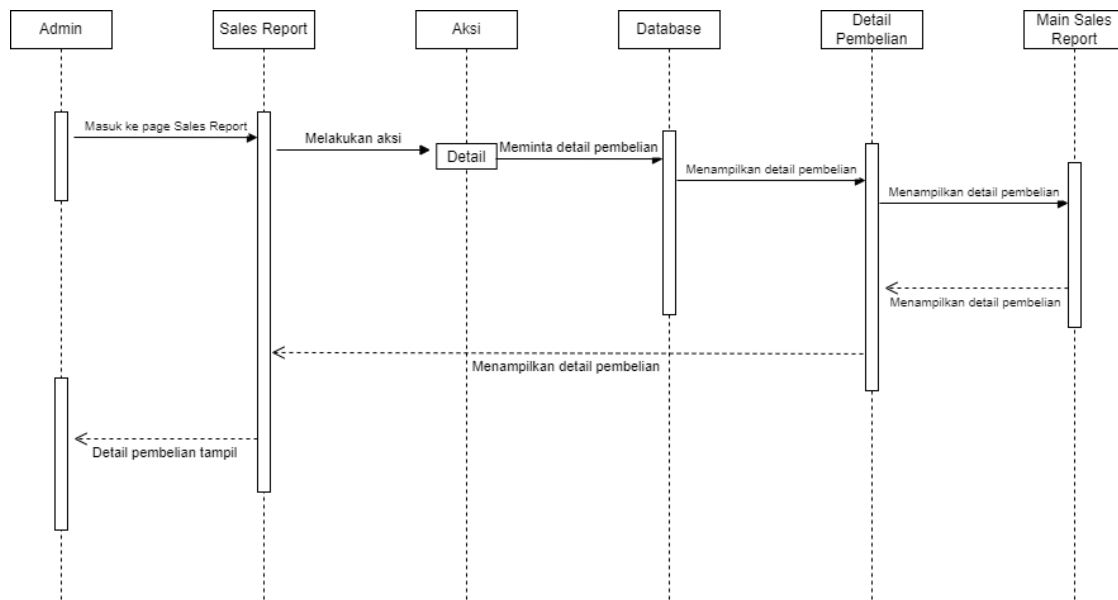
*Tabel 31.OBJECT PERANCANGAN*

*\*Tipe kelas diisi dengan Boundary(Interface), Entity(Database), Controller*

### 3.1.7.4 Sequence Diagram #7 <Laporan Penjualan >

Buatlah diagram sequence untuk masing - masing use case sesuai skenario.

Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.



*Gambar 15. Tampilan Sequence Diagram Laporan Penjualan*

### 3.1.8 Use Case #8 <Order Hoodie >

#### 3.1.8.1 Use Case Scenario #8 <Order Hoodie >

Skenario Use Case #8 :

- i. Pre-Condition  
User telah melakukan login
- ii. Use Case Description
  - a. Primary Flow

Aktor	Sistem
1. User membuka halaman hoodie yang dijual.	
	2. Menampilkan laman hoodie yang dijual.
3. User memilih hoodie yang ingin ditambahkan ke keranjang.	
4. User menekan hoodie yang dipilih.	
	5. Menampilkan laman keranjang.
6. User mengklik tombol checkout.	
	7. Menampilkan laman checkout .
8. User mengisi form alamat dan ongkos kirim lalu menekan tombol checkout	
	9. Data pembelian masuk ke dalam database

*Tabel 32. Use Case Description, Primary Flow*

- b. Alternative Flow
  1. User menekan tombol home, maka:
    - Sistem akan menampilkan halaman utama dari website Ya-Hoodie.
  2. User menekan tombol “Our Hoodie”, maka:
    - Sistem akan menampilkan stok hoodie yang tersedia pada database di halaman product.



3. User menekan tombol buy atau “add to cart” maka:
  - Sistem akan menampilkan orderan hoodie yang akan dibeli.
4. User menekan tombol checkout, maka:
  - Sistem akan menampilkan laman checkout.
5. User mengisi form alamat dan ongkos kirim lalu menekan tombol checkout, maka :
  - Sistem akan menyimpan data pembelian dalam database
6. User menekan tombol unggah bukti pembayaran, maka:
  - Sistem akan menampilkan halaman untuk user mengunggah bukti pembayaran.
7. User menekan tombol batal, maka:
  - Sistem akan membatalkan proses pembelian

iii. Post-Condition

User telah memilih Hoodie yang tersedia dan melakukan unggah bukti pembayaran

### 3.1.8.2 UI Design dan Deskripsi Objek UI #8 <Order Hoodie >


Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case  
Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detailnya.

ID. LAYAR	NAMA LAYAR	DESKRIPSI
	Page Order Hoodie	Setelah user menekan tombol “add to cart” pada our hoodies maka sistem akan menyimpannya di keranjang belanja, yang dimana pada keranjang belanja ini user dapat melakukan checkout, lanjutkan belanja, dan juga mendelete hoodie yang akan dibelanjakan.
	Page Checkout	Setelah user menekan tombol “Checkout “ maka user dapat mengisi data alamat, dan juga pengiriman. Lalu user dapat mengcheckout barang.

Tabel 33.Deskripsi Objek UI

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detail  
Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}

## Keranjang Belanja

NO	Produk	Ukuran	Harga	Jumlah	Subharga	Aksi
1	Street Hoodie	XXL	Rp. 300,000	1	Rp. 300,000	

[Lanjutkan belanja](#)
[Checkout](#)

Reach at..

📍 Location

☎ Call +62 823 2138 4132

✉ customerservice@yahoodie.com

**YA-Hoodie**

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.



Gambar 16. Tampilan Order Hoodie pada website YA-HOODIE

## Checkout

NO	Produk	Ukuran	Harga	Jumlah	Subharga
1	Street Hoodie	XXL	Rp. 300,000	1	Rp. 300,000
<b>Total Belanja</b>					<b>Rp. 300,000</b>






Alamat Pengiriman

jl. Sukabirus no 6 (kost Puri Kharisma)

[Checkout](#)

Gambar 17. Tampilan Order Hoodie (Checkout) pada website YA-HOODIE



[HOME](#)
[PAGES ▾](#)
[PRODUCTS](#)
[RIWAYAT PEMBELIAN](#)
[LOGOUT](#)


**Pembelian**  
**No. Pembelian: 25**  
 Tanggal Pembelian: 2023-01-02  
 Total: 300,000

**Pelanggan**  
**Ujang Sae**  
 ujang

**Pengiriman**  
**Alamat Pelanggan**  
 Jl. Sukabirus no 6 (kost Puri Kharisma)

No	Nama Hoodie	Ukuran	Harga	Jumlah	Subtotal
1	Street Hoodie	XXL	Rp. 300,000	1	Rp. 300,000

Silakan melakukan pembayaran dengan sistem COD sebesar Rp. 300,000

Pembelian anda akan kami proses segera!

Estimasi proses 2-3 hari sampai barang diterima, dan jangan lupa untuk siapkan pembayaran ketika paket hoodie anda datang!

Terima kasih telah memilih YA-Hoodie!

*Gambar 18. Tampilan Bukti Pembelian pada website YA-HOODIE*

Id_Objek	JENIS	LABEL*	Keterangan**
Button 1	Button	Lanjutkan Belanja	User mengklik tombol tersebut, maka user dapat melanjutkan belanja
Button 2	Button	Checkout	User mengklik tombol tersebut, maka user dapat check out barang
Button 3	Button	Aksi, logo X	User mengklik tombol tersebut, maka user dapat menghapus hoodie yang dimasukkan ke keranjang

Tabel 34. Page Order Hoodie

Id_Objek	JENIS	LABEL*	Keterangan**
RadioButton1	RadioButton	Pilih Ongkos Kirim	User dapat memilih ongkos kirim yang digunakan
TextBox1	TextBox	Alamat Pengiriman	User dapat mengisi alamat pengiriman
Button 1	Button	Checkout	User dapat menekan tombol Checkout , dan sistem akan melanjutkan pada laman bukti pembelian, dan pembelian pun selesai.

Tabel 35. Page Checkout

### 3.1.8.3 Identifikasi Object dan Tipe nya #8 <Order Hoodie >

Identifikasi object yang terkait dengan use case tersebut.

Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.

Gunakan tabel di bawah:

**TABEL OBJECT PERANCANGAN**

No	Nama Object Baru	Jenis / Tipe Kelas
	<i>TBD</i>	<i>TBD</i>

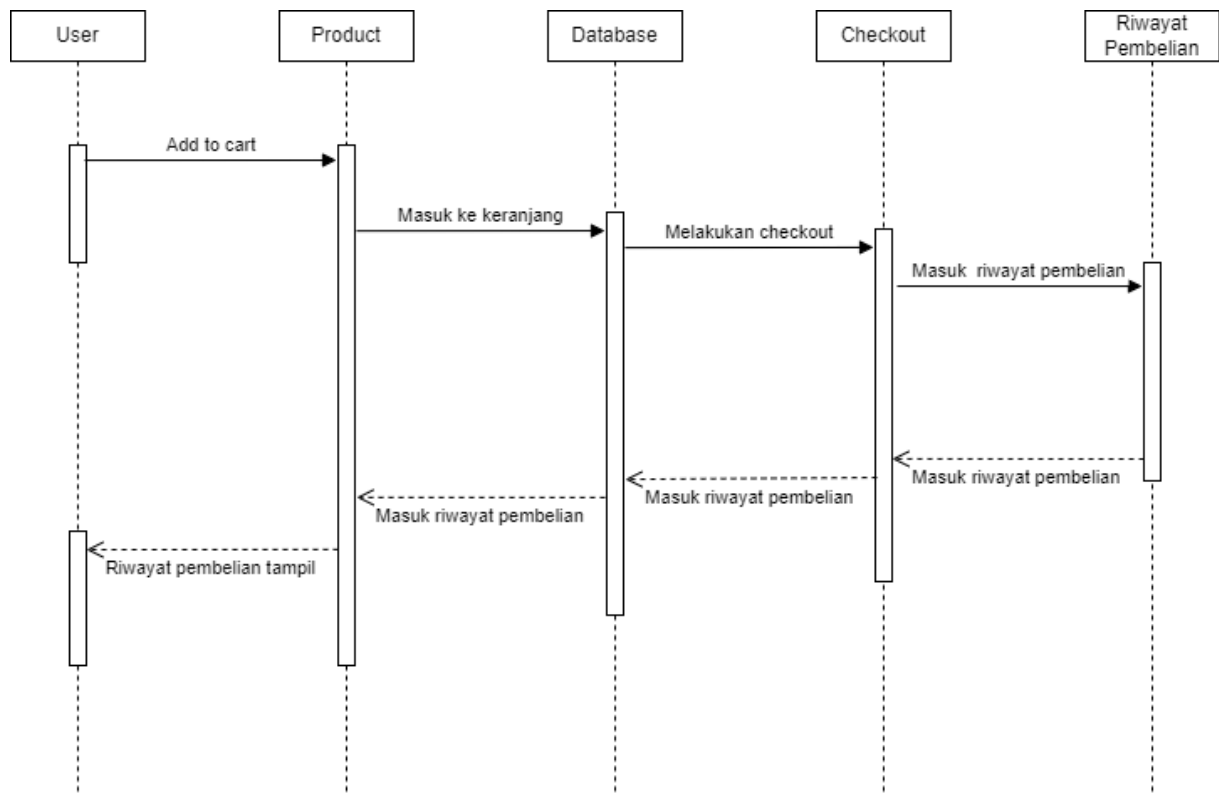
*Tabel 35. Object Perancangan*

*\*Tipe kelas diisi dengan Boundary(Interface), Entity(Database), Controller*

### 3.1.8.4 Sequence Diagram #8 <Order Hoodie >

Buatlah diagram sequence untuk masing - masing use case sesuai skenario.

Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.



*Gambar 19. Tampilan Sequence Diagram Order Hoodie*

### 3.1.9 Use Case #9 <Review Hoodie>

#### 3.1.9.1 Use Case Scenario #9 <Review Hoodie>

Skenario Use Case #9 :

- i. Pre-Condition  
User sudah melakukan login.
- ii. Use Case Description
  - a. Primary Flow

Aktor	Sistem
1. User membuka halaman review pada navbar pages.	
	2. Sistem akan menampilkan laman review
3. User mengklik tombol “Review”.	
	4. Sistem menampilkan form review
5. User mengisi rating, kolom username, dan komentar, lalu menekan tombol “submit”	
	6. Sistem menyimpan data review yang diberikan user ke dalam database

*Tabel 36. Use Case Description, Primary Flow*

#### b. Alternative Flow

1. User menekan tombol Review setelah pembelian hoodie selesai pada dashboard pages, maka
  - Sistem akan menampilkan tampilan Review & Rating of Ya-Hoodie
2. . Jika user menekan tombol Review, maka:
  - Sistem akan menampilkan sebuah pilihan kepada user yaitu berupa bintang dari skala 1-5 lalu username dan komentar yang akan diberikan oleh user.
3. Jika user sudah menentukan review, maka:
  - Sistem akan menyimpan ulasan itu di database.

4. Jika user menekan tombol home, maka:
  - Sistem akan kembali ke halaman utama.

iii. Post-Condition

User telah memilih halaman review hoodie dan akan memberikan ulasan untuk website dari toko YA-HOODIE ini.

### 3.1.9.2 UI Design dan Deskripsi Objek UI #9 <Review Hoodie>

Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detailnya.

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
	<i>Page Review</i>	<i>Menampilkan review user terhadap website</i>
	<i>Page AddReview</i>	<i>berfungsi untuk memberi ulasan dari user terhadap Website YA-HOODIE. Review ini berupa bintang dari skala 5, inputan username, dan juga komentar. Yang nantinya akan disimpan pada database.</i>

Tabel 37. Deskripsi Objek UI

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detail  
*Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}*

## Review & Rating of Ya-Hoodie

All Product

4.1 / 5

★★★★★

8 Review

5 ★ (5)

4 ★ (1)

3 ★ (1)

2 ★ (0)

1 ★ (1)

Write Review Here

Review

h

hamba Allah

★★★★★

masyaallah tabarakallah

F

Fachrul

★★★★★

Mantap banget guys kalian harus beli hoodie disini

Gambar 20. Tampilan Review (testimonial) pada website YA-HOODIE

Review & Rating of Ya-Hoodie

All Product

4.1 / 5

★★★★★

8 Review

★★★★★

Username

Type Review Here

Submit

Write Review Here

Review

h

hamba Allah

★★★★★

Gambar 21. Tampilan addReview pada website YA-HOODIE

*Page Review*

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
<i>Button1</i>	<i>Button</i>	<i>Review</i>	<i>Jika diklik, akan mengaktifkan Function memulai review yang dilakukan oleh user.</i>

*Tabel 38. Page Review*

*Page addReview*

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
<i>JLabelStarRating1</i>	<i>JLabelStarRating</i>	<i>Bintang</i>	<i>User dapat memilih bintang dari rate 1-5</i>
<i>TextBox1</i>	<i>TextBox</i>	<i>Username</i>	<i>User dapat mengisi username untuk review</i>
<i>TextBox2</i>	<i>TextBox</i>	<i>Type Review Here</i>	<i>User dapat mengisi komentar untuk review</i>
<i>Button 1</i>	<i>Button</i>	<i>Submit</i>	<i>Jika User Mengklik tombol tersebut, maka sistem akan menyimpan review dari user ke dalam database.</i>

*Tabel 39. Page addReview*

**3.1.9.3 Identifikasi Object dan Tipe nya #9 <Review Hoodie>**

*Identifikasi object yang terkait dengan use case tersebut.*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.*

*Gunakan tabel di bawah:*

**TABEL OBJECT PERANCANGAN**

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
	<i>TBD</i>	<i>TBD</i>

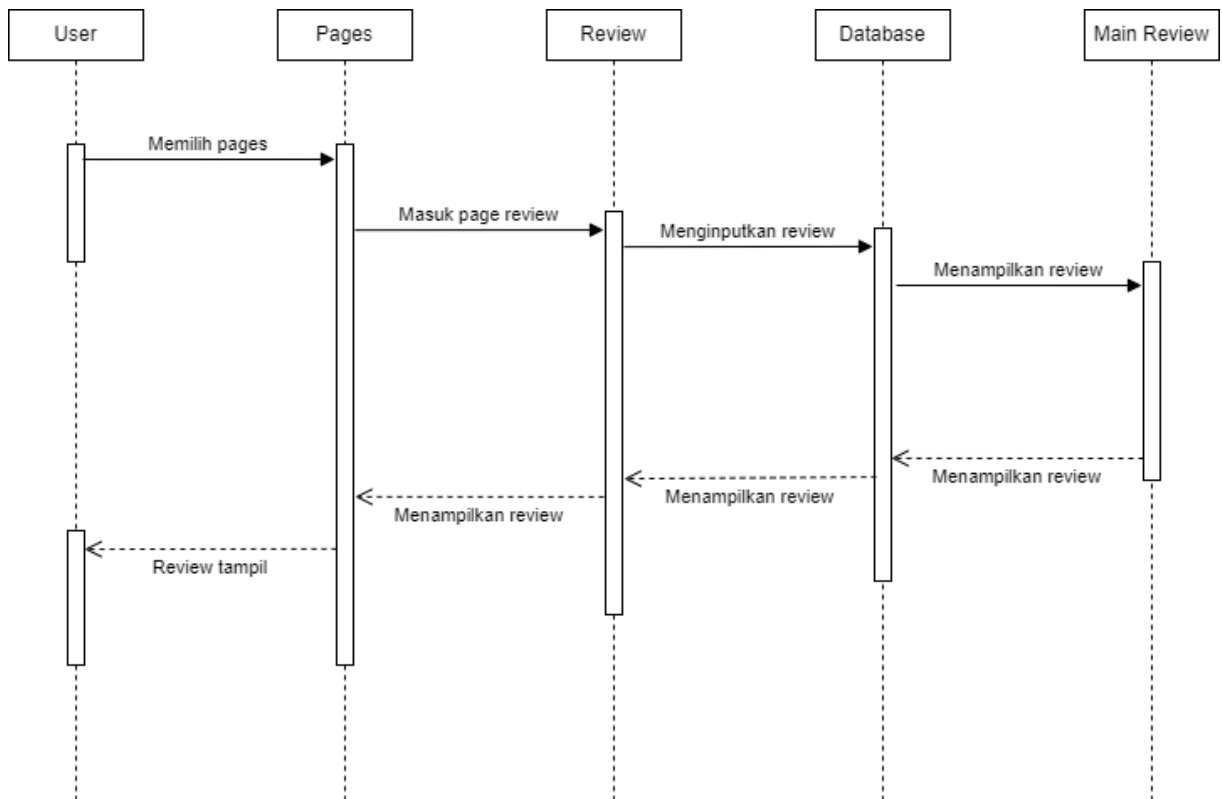
*Tabel 40. Object Perancangan*

*\*Tipe kelas diisi dengan Boundary(Interface), Entity(Database), Controller*



### 3.1.9.4 Sequence Diagram #9 <Review Hoodie>

Buatlah diagram sequence untuk masing - masing use case sesuai skenario.  
Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.



Gambar 22. Tampilan Sequence Diagram Review Hoodie

### 3.1.10 Use Case #10 <Statistik Review>

#### 3.1.10.1 Use Case Scenario #10 <Statistik Review>

Skenario Use Case #10 :

- i. Pre-Condition  
Owner sudah melakukan login.
- ii. Use Case Description
  - a. Primary Flow

Aktor	Sistem
1. Owner masuk ke laman "Review Statistic"	
	2. Sistem menampilkan laman yang berisi statistik ulasan user.

*Tabel 41. Use Case Description, Primary Flow*

b. Alternative Flow

1. Owner menekan tombol “Review Statistic” pada laman dashboard owner.
  - Sistem akan menampilkan tampilan statistik review.

iii. Post-Condition

Owner dapat melihat statistik ulasan.

**3.1.10.2 UI Design dan Deskripsi Objek UI #10 <Statistik Review>**

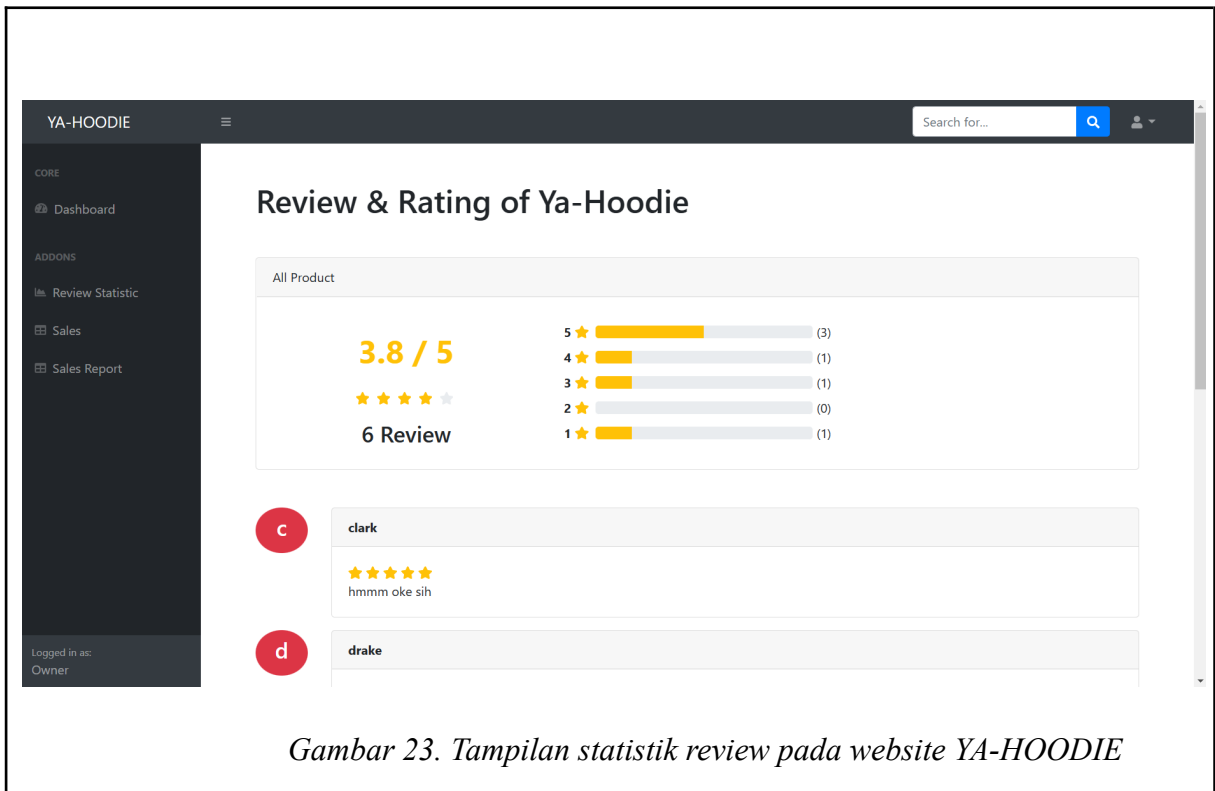
*Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detailnya.*

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
	<i>Page Statistik review</i>	<i>Page ini menampilkan statistik review yang dapat dilihat oleh owner saja.</i>

*Tabel 42. Deskripsi Objek UI*

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detail  
Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}



Gambar 23. Tampilan statistik review pada website YA-HOODIE

#### Page Statistik Review

Id_Objek	JENIS	LABEL*	Keterangan**
-	-	-	Tidak ada objek dari page Statistik Review dikarenakan owner hanya bisa melihat hasil review dari user saja.

Tabel 43. Page Statistik Review

#### 3.1.10.3 Identifikasi Object dan Tipe nya #10 <Statistik Review>

Identifikasi object yang terkait dengan use case tersebut.

Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.

Gunakan tabel di bawah:

#### TABEL OBJECT PERANCANGAN

No	Nama Object Baru	Jenis / Tipe Kelas
	TBD	TBD

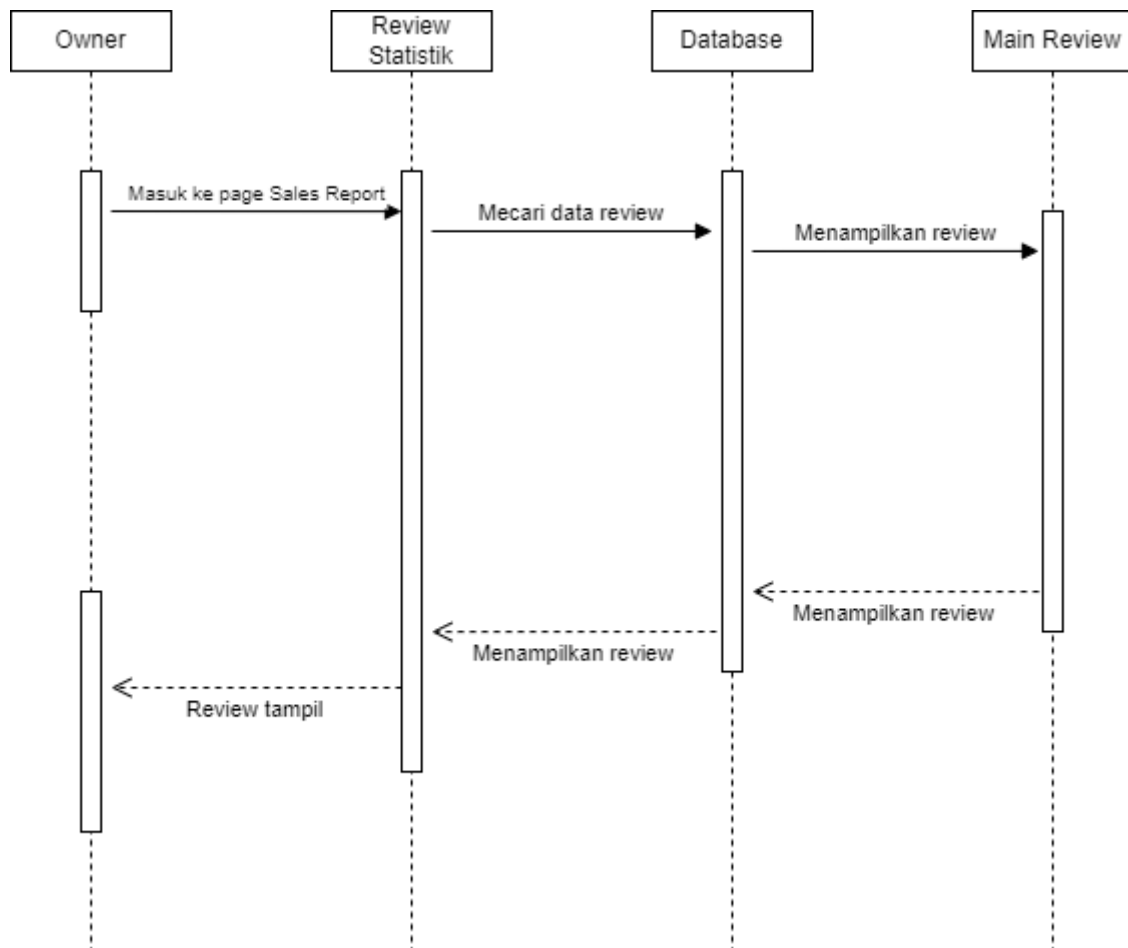
Tabel 44. Object Perancangan

\*Tipe kelas diisi dengan Boundary(Interface), Entity(Database), Controller

#### 3.1.10.4 Sequence Diagram #10 <Statistik Review>

##### Sequence Diagram #10 <Statistik Review>

Buatlah diagram sequence untuk masing - masing use case sesuai skenario.  
Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.



Gambar 24. Tampilan Sequence Diagram Statistik Review

#### 3.1.11 Use Case #11 <Delete Customer>

##### 3.1.11.1 Use Case Scenario #11 <Delete Customer>

Skenario Use Case #11 :

- i. Pre-Condition  
Admin sudah melakukan login.
- ii. Use Case Description
  - a. Primary Flow

Aktor	Sistem
1. Admin mengakses laman “Customers”	
	2. Sistem menampilkan laman “Customers” dan menampilkan data user yang terdapat di database.
3. Admin menekan tombol “Delete” pada data customer yang dipilih.	
	4. Sistem menghapus data customer dari database.

*Tabel 45. Primary Flow*

b. Alternative Flow

- Owner menekan tombol “Delete” pada laman customer dari dashboard admin.  
- Sistem akan menghapus data customer dari database..

iii. Post-Condition

Data customer terhapus dari database.

**3.1.11.2 UI Design dan Deskripsi Objek UI #11 <Delete Customer>**

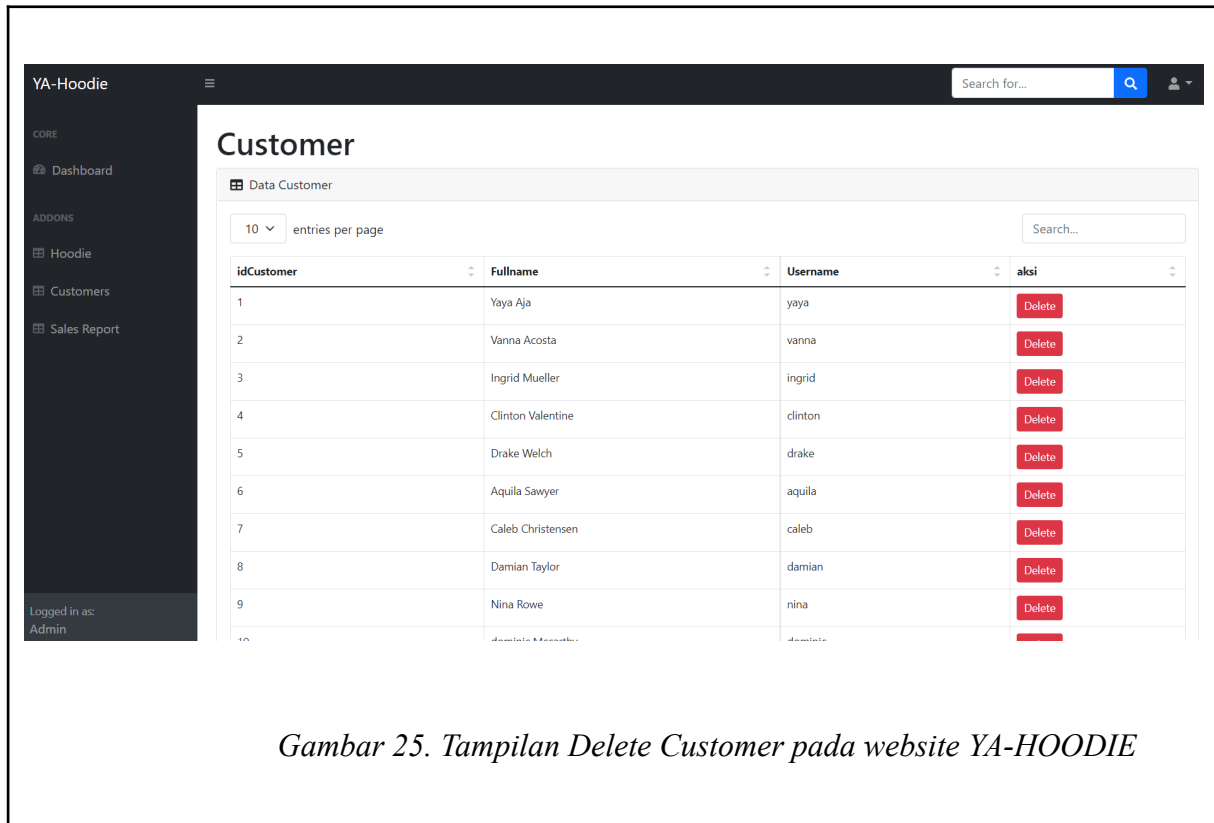
*Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detailnya.*

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
	<i>Page Delete Customer</i>	<i>Pada page ini menampilkan data customer yang tertampil pada dashboard customer pada admin. Admin dapat menghapus data dari customer.</i>

Tabel 46. Deskripsi Objek UI

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detail  
Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}



Gambar 25. Tampilan Delete Customer pada website YA-HOODIE

#### Page Delete Customer

Id_Objek	JENIS	LABEL*	Keterangan**
Button1	Button	Delete	Jika diklik, akan mengaktifkan Function delete pada data customer yang diinginkan oleh admin.

Tabel 47. Page Delete Customer

#### 3.1.11.3 Identifikasi Object dan Tipe nya #11 <Delete Customer>

Identifikasi object yang terkait dengan use case tersebut.

Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.

Gunakan tabel di bawah:

#### TABEL OBJECT PERANCANGAN

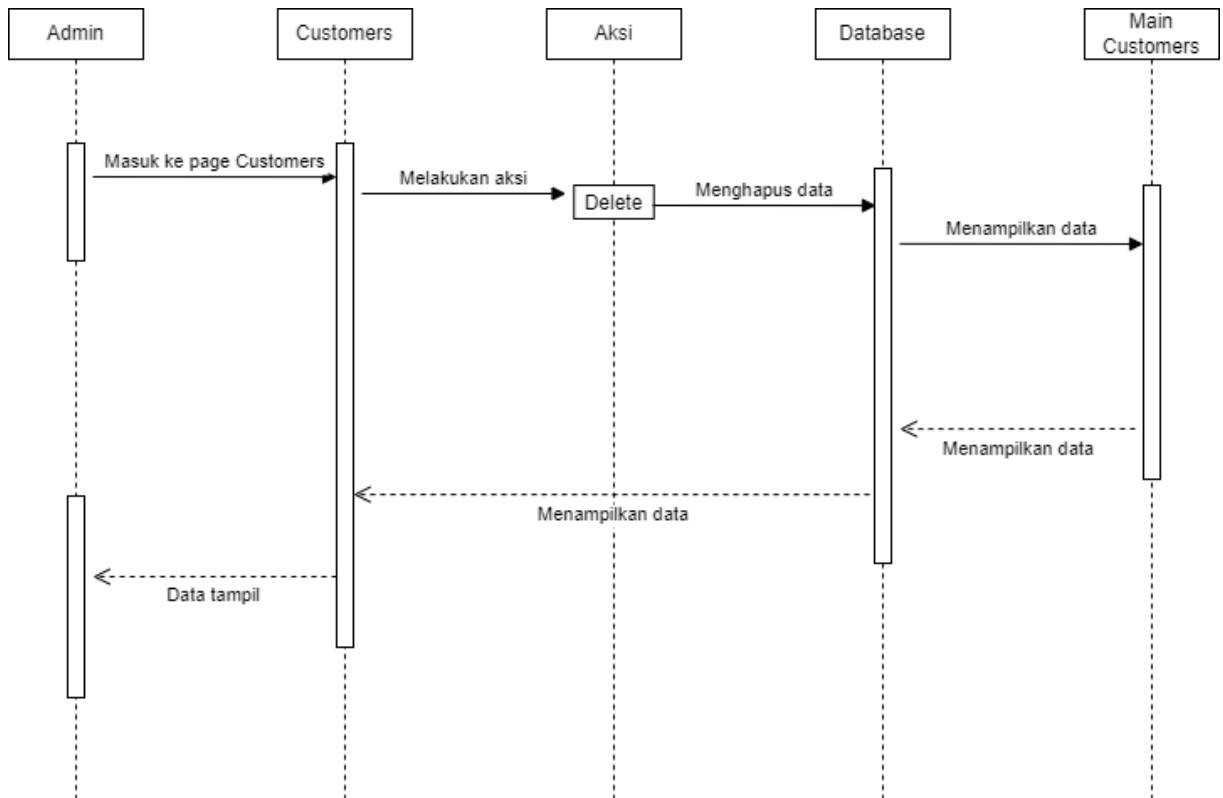
No	Nama Object Baru	Jenis / Tipe Kelas
	TBD	TBD

Tabel 48. Object Perancangan

\*Tipe kelas diisi dengan Boundary(Interface), Entity(Database), Controller

#### 3.1.11.4 Sequence Diagram #11 <Delete Customer>

Buatlah diagram sequence untuk masing - masing use case sesuai skenario.  
Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.



Gambar 26. Tampilan Sequence Diagram Delete Customer

#### 3.1.12 Use Case #12 <Logout User>

##### 3.1.12.1 Use Case Scenario #12 <Logout User>

Skenario Use Case #12 :

- i. Pre-Condition  
User sudah melakukan login.

- ii. Use Case Description  
 a. Primary Flow

Aktor	Sistem
1. User menekan tombol logout	
	2. Sistem menghapus sesi User

*Tabel 49. Primary Flow*

- b. Alternative Flow

-

- iii. Post-Condition

User keluar dari sistem.

### **3.1.12.2 UI Design dan Deskripsi Objek UI #12 <Logout User>**

*Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detailnya.*

*Tabel Deskripsi Objek UI*

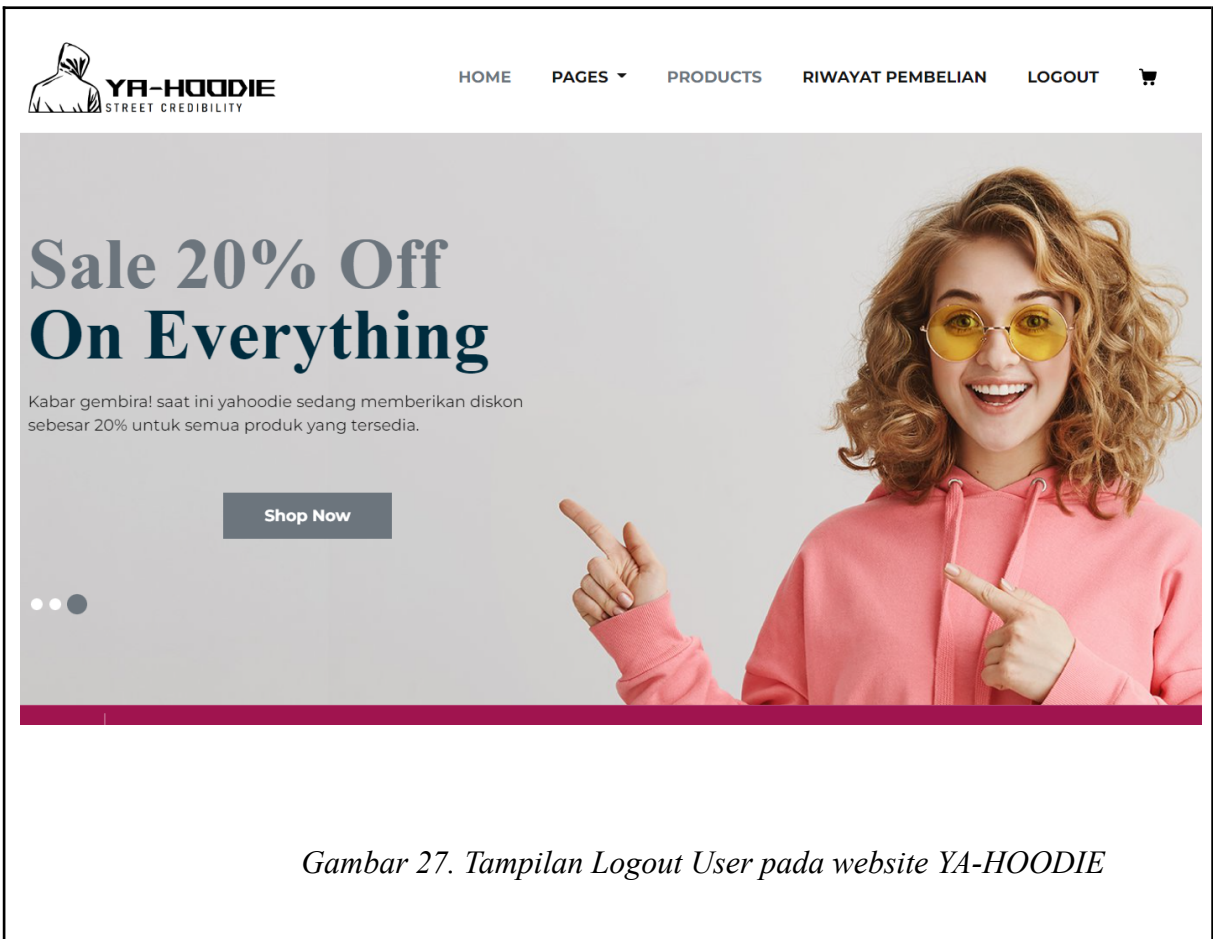
ID. LAYAR	NAMA LAYAR	DESKRIPSI
	<i>Page Logout User</i>	<i>User dapat logout pada halaman utama atau dashboard pada navbar di atas . Lalu sistem akan meng logout otomatis dari laman Ya-Hoodie.</i>

*Tabel 50. Deskripsi Objek UI*

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detail  
*Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}*

--





Gambar 27. Tampilan Logout User pada website YA-HOODIE

#### Page Logout User

Id_Objek	JENIS	LABEL*	Keterangan**
Button1	Button	Logout	Jika diklik, maka sistem akan keluar dari laman sistem dan user harus login ulang jika ingin memasuki laman Ya-Hoodie User.

Tabel 51. Page Logout User

#### 3.1.12.3 Identifikasi Object dan Tipe nya #12 <Logout User>

Identifikasi object yang terkait dengan use case tersebut.

Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.

Gunakan tabel di bawah:

#### TABEL OBJECT PERANCANGAN

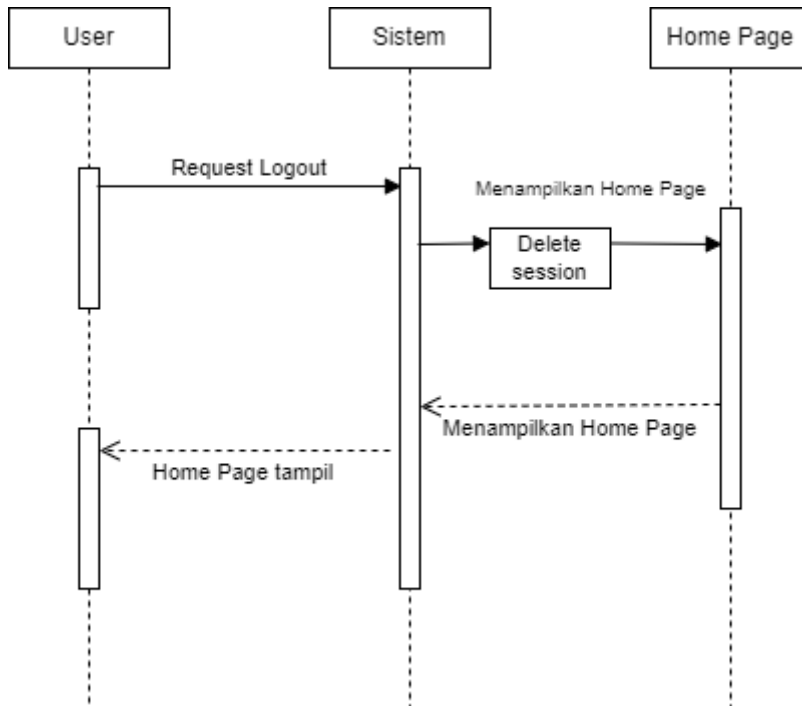
No	Nama Object Baru	Jenis / Tipe Kelas
	TBD	TBD


Tabel 52. OBJECT PERANCANGAN

\*Tipe kelas diisi dengan Boundary(Interface), Entity(Database), Controller

#### 3.1.12.4 Sequence Diagram #12 <Logout User>

Buatlah diagram sequence untuk masing - masing use case sesuai skenario.  
Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.



Gambar 28. Tampilan Sequence Diagram Logout User

#### 3.1.13 Use Case #13 <Logout Admin>

##### 3.1.13.1 Use Case Scenario#13 <Logout Admin>

Skenario Use Case #13 :

- i. Pre-Condition  
Admin sudah melakukan login.
- ii. Use Case Description
  - a. Primary Flow

Aktor	Sistem
-------	--------

1. Admin menekan tombol logout	
	2. Sistem menghapus sesi Admin

*Tabel 53. Primary Flow*

b. Alternative Flow

-

iii. Post-Condition

Admin keluar dari sistem.

**3.1.13.2 UI Design dan Deskripsi Objek UI #13 <Logout Admin>**

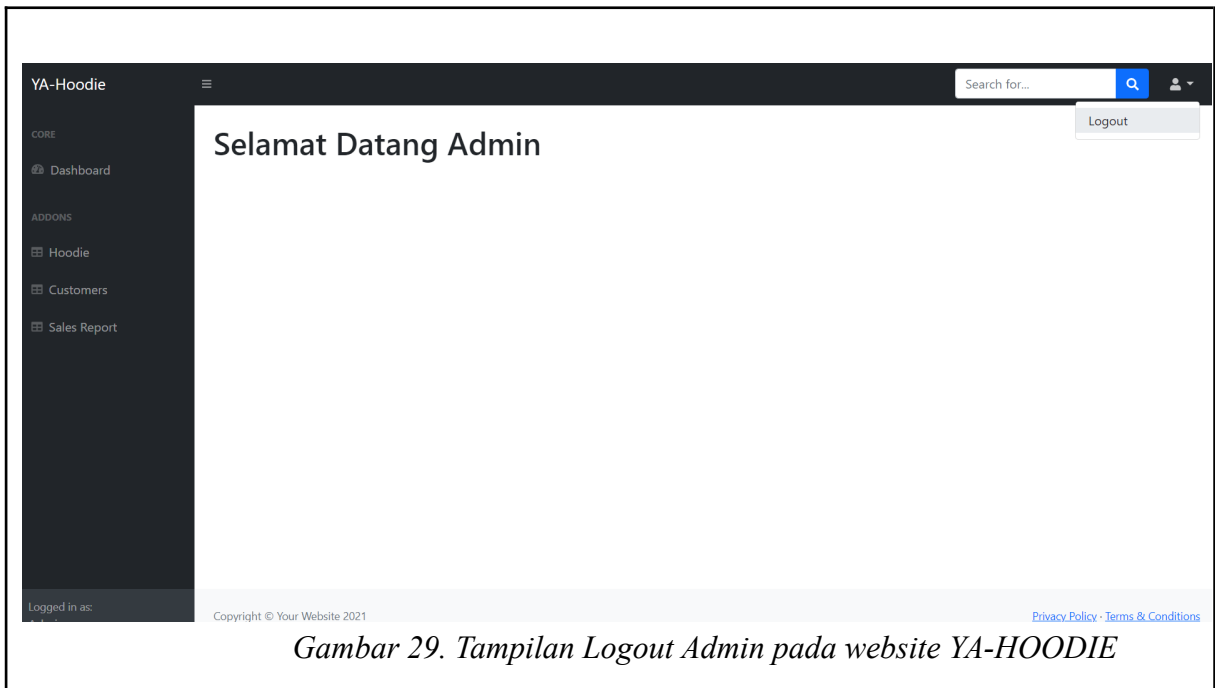
*Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detailnya.*

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
	<i>Page Logout Admin</i>	<i>Admin dapat logout pada halaman utama atau dashboard pada navbar di atas kiri . Lalu sistem akan meng logout otomatis dari laman Ya-Hoodie.</i>

*Tabel 54. Deskripsi Objek UI*

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detail  
Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}



Gambar 29. Tampilan Logout Admin pada website YA-HOODIE

#### Page Logout Admin

Id_Objek	JENIS	LABEL*	Keterangan**
Button1	Button	Logout	Jika diklik, maka sistem akan keluar dari laman sistem dan admin harus login ulang jika ingin memasuki laman Ya-Hoodie Admin.

Tabel 55. Page Logout Admin

#### 3.1.13.3 Identifikasi Object dan Tipe nya #13 <Logout Admin>

Identifikasi object yang terkait dengan use case tersebut.

Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.

Gunakan tabel di bawah:

#### TABEL OBJECT PERANCANGAN

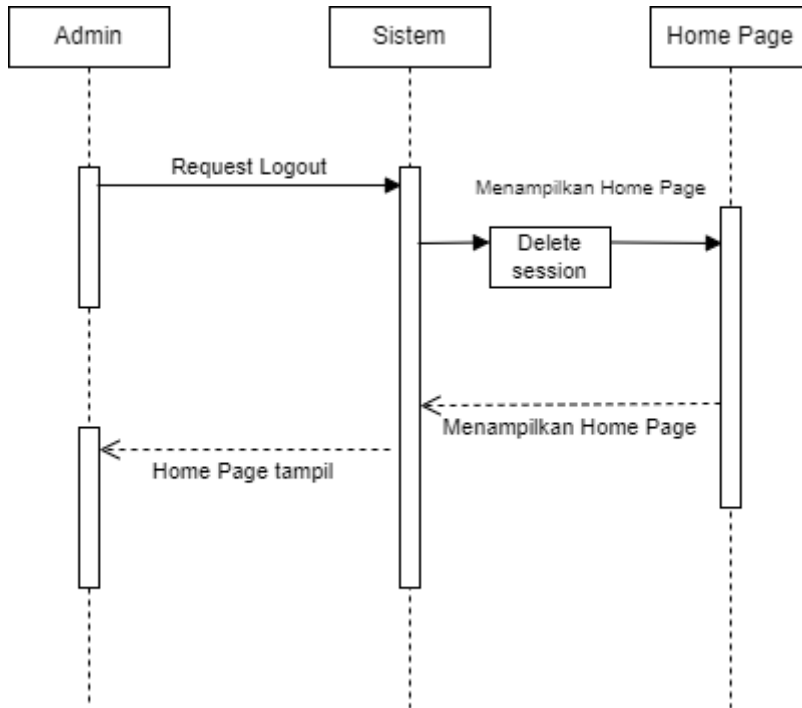
No	Nama Object Baru	Jenis / Tipe Kelas
	TBD	TBD

Tabel 56. OBJECT PERANCANGAN

\*Tipe kelas diisi dengan Boundary(Interface), Entity(Database), Controller

#### 3.1.13.4 Sequence Diagram #13 <Logout Admin>

Buatlah diagram sequence untuk masing - masing use case sesuai skenario.  
Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.



Gambar 30. Tampilan Sequence Diagram Logout Admin

#### 3.1.14 Use Case #14 <Logout Owner>

##### 3.1.14.1 Use Case Scenario #14 <Logout Owner>

Skenario Use Case #14 :

- i. Pre-Condition  
Owner sudah melakukan login.
- ii. Use Case Description
  - a. Primary Flow

Aktor	Sistem
1. Owner menekan tombol logout	
	2. Sistem menghapus sesi Owner

Tabel 57. Primary Flow

b. Alternative Flow

-

iii. Post-Condition

Owner keluar dari sistem.

### 3.1.14.2 UI Design dan Deskripsi Objek UI #14 <Logout Owner>

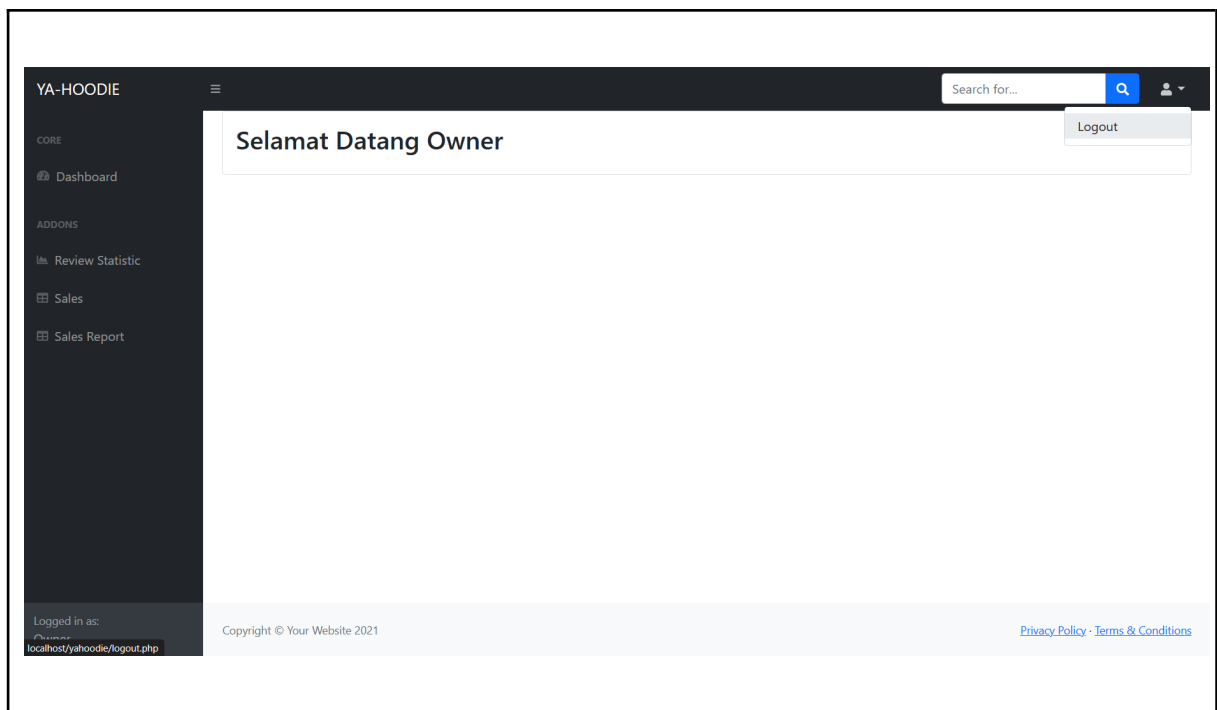
Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detailnya.

Tabel Deskripsi Objek UI

ID. LAYAR	NAMA LAYAR	DESKRIPSI
	Page Logout Admin	Owner dapat logout pada halaman utama atau dashboard pada navbar di atas kiri . Lalu sistem akan meng logout otomatis dari laman Ya-Hoodie.

Tabel 58. Deskripsi Objek UI

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detail Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}



*Gambar 31. Tampilan Logout Owner pada website YA-HOODIE*

*Page Logout Owner*

<b>Id_Objek</b>	<b>JENIS</b>	<b>LABEL*</b>	<b>Keterangan**</b>
<i>Button1</i>	<i>Button</i>	<i>Logout</i>	<i>Jika diklik, maka sistem akan keluar dari laman sistem dan owner harus login ulang jika ingin memasuki laman Ya-Hoodie Owner</i>

#### **3.1.14.3 Identifikasi Object dan Tipe nya#14 <Logout Owner>**

*Identifikasi object yang terkait dengan use case tersebut.*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.*

*Gunakan tabel di bawah:*

**TABEL OBJECT PERANCANGAN**

<b>No</b>	<b>Nama Object Baru</b>	<b>Jenis / Tipe Kelas</b>
	<i>TBD</i>	<i>TBD</i>

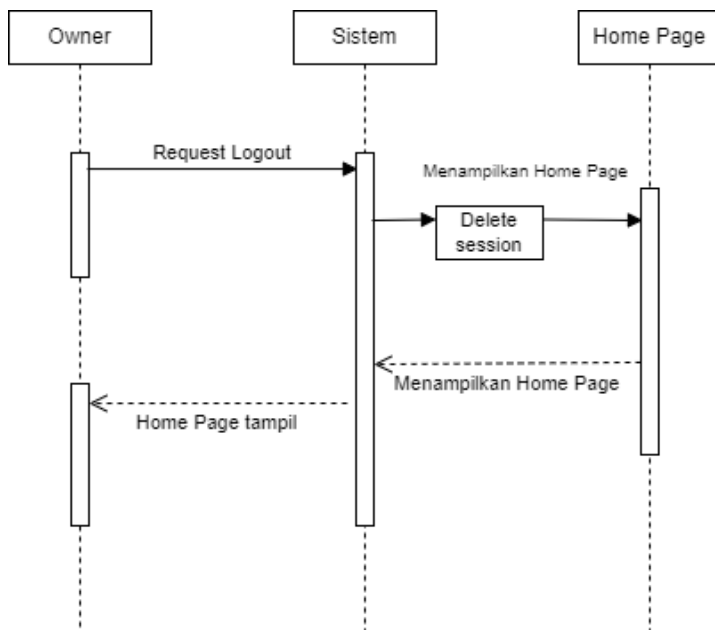
*Tabel 59. OBJECT PERANCANGAN*

*\*Tipe kelas diisi dengan Boundary(Interface), Entity(Database), Controller*

#### **3.1.14.4 Sequence Diagram #14 <Logout Owner>**

*Buatlah diagram sequence untuk masing - masing use case sesuai skenario.*

*Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.*

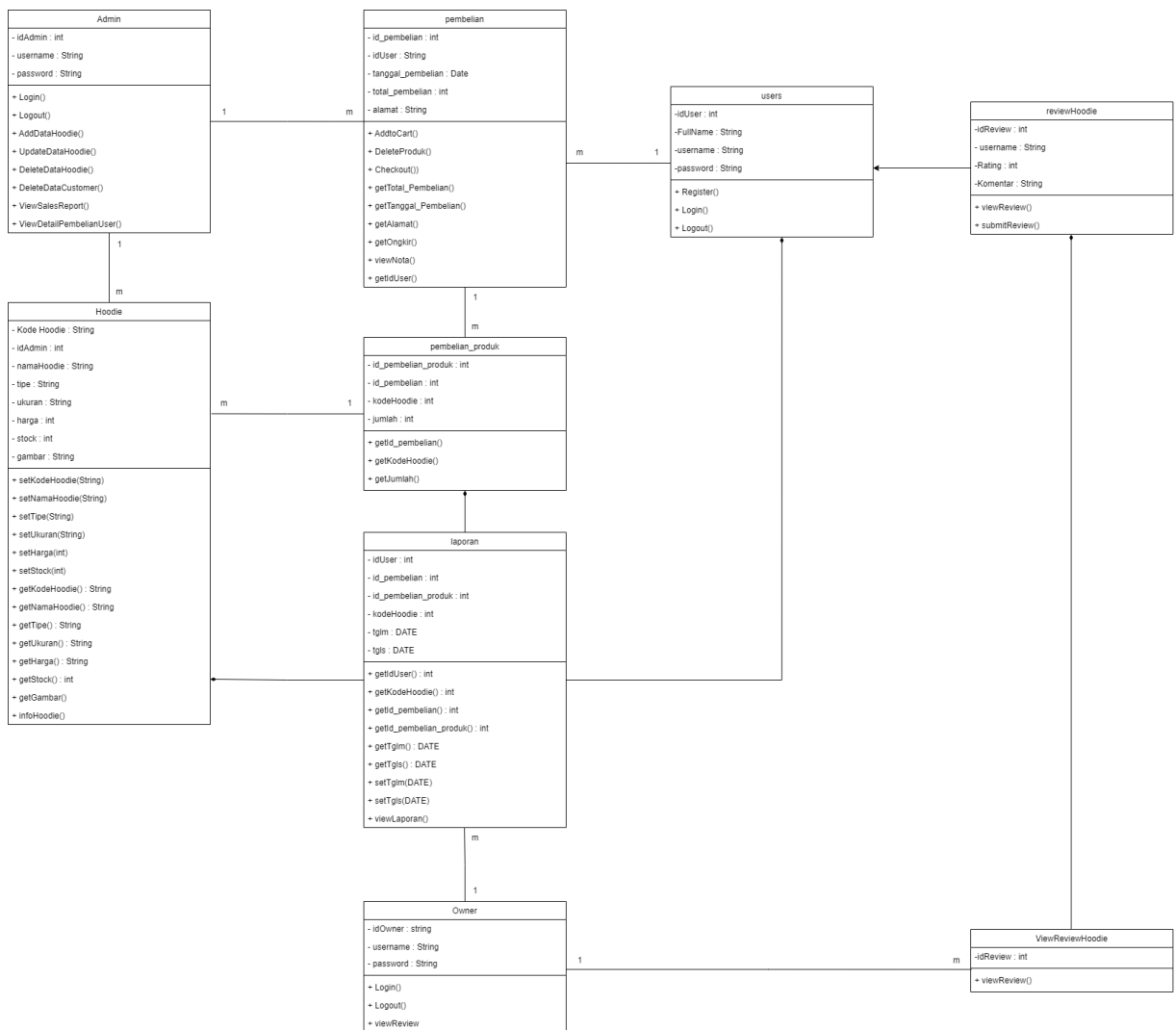


Gambar 32. Tampilan Sequence Diagram Logout Owner

### 3.2 Diagram Kelas Keseluruhan

Bagian ini diisi dengan diagram kelas keseluruhan yang akan digunakan dalam PL menggunakan model MVC





Gambar 33. Tampilan Diagram Class Keseluruhan

#### TABEL KELAS : User

ID Kelas	Nama Kelas Perancangan	Attribute (visibility)	Method / Operation
1.	User	idUser : int	Register()
		username : string	Login()
		password : string	Logout()
		fullname : string	

Tabel 60. TABEL KELAS : User

#### TABEL KELAS : Admin

ID Kelas	Nama Kelas Perancangan	Attribute (visibility)	Method / Operation
1.	<i>Admin</i>	<i>idAdmin : int</i>	<i>Login()</i>
		<i>username : string</i>	<i>Logout()</i>
		<i>password : string</i>	<i>AddDataHoodie()</i>
			<i>UpdateDataHoodie()</i>
			<i>DeleteDataHoodie()</i>
			<i>DeleteDataCustomer()</i>
			<i>ViewSalesReport()</i>
			<i>ViewDetailPembelianUser()</i>

*Tabel 61. TABEL KELAS : Admin*

**TABEL KELAS : Owner**

ID Kelas	Nama Kelas Perancangan	Attribute (visibility)	Method / Operation
1.	<i>Owner</i>	<i>idOwner : string</i>	<i>Login()</i>
		<i>username : string</i>	<i>Logout()</i>
		<i>password : string</i>	<i>viewReview()</i>

*Tabel 62. TABEL KELAS : Owner*

**TABEL KELAS : Pembelian**

ID Kelas	Nama Kelas Perancangan	Attribute (visibility)	Method / Operation
1.	<i>Pembelian</i>	<i>id pembelian: inr</i>	<i>AddtoCart()</i>
		<i>idUser : string</i>	<i>DeleteProduk()</i>
		<i>tanggal pembelian : Date</i>	<i>Checkout()</i>
		<i>total pembelian : int</i>	<i>getTotal Pembelian()</i>
		<i>alamat : string</i>	<i>getTanggal Pembelian()</i>
			<i>getAlamat()</i>
			<i>getOngkir()</i>
			<i>viewNota()</i>
			<i>getIdUser()</i>

*Tabel 63. TABEL KELAS : Pembelian*

**TABEL KELAS : Hoodie**

<b>ID Kelas</b>	<b>Nama Kelas Perancangan</b>	<b>Attribute (visibility)</b>	<b>Method / Operation</b>
1.	<i>Hoodie</i>	<i>KodeHoodie : string</i>	<i>setKodeHoodie(String)</i>
		<i>idAdmin : int</i>	<i>setNamaHoodie(String)</i>
		<i>namaHoodie : string</i>	<i>setTipe(String)</i>
		<i>tipe : string</i>	<i>setUkuran(String)</i>
		<i>ukuran : string</i>	<i>setHarga(int)</i>
		<i>harga : int</i>	<i>setStock(int)</i>
		<i>stock : int</i>	<i>getKodeHoodie(): string</i>
		<i>gambar : string</i>	<i>getNamaHoodie(): string</i>
			<i>getTipe() : string</i>
			<i>getUkuran() : string</i>
			<i>getHarga() : string</i>
			<i>getStock() : int</i>
			<i>getGambar()</i>
			<i>infoHoodie()</i>

*Tabel 64. TABEL KELAS : Hoodie***TABEL KELAS : Pembelian\_produk**

<b>ID Kelas</b>	<b>Nama Kelas Perancangan</b>	<b>Attribute (visibility)</b>	<b>Method / Operation</b>
1.	<i>pembelian produk</i>	<i>id pembelian produk : int</i>	<i>getId pembelian()</i>
		<i>id pembelian : int</i>	<i>getKodeHoodie()</i>
		<i>kodeHoodie : int</i>	<i>getJumlah()</i>
		<i>jumlah : int</i>	

*Tabel 65. TABEL KELAS : Pembelian\_produk*

**TABEL KELAS : laporan**

ID Kelas	Nama Kelas Perancangan	Attribute (visibility)	Method / Operation
1.	<i>laporan</i>	<i>idUser: int</i>	<i>getIdUser() : int</i>
		<i>id pembelian : int</i>	<i>getKodeHoodie() : int</i>
		<i>id pembelian produk : int</i>	<i>getId pembelian() : int</i>
		<i>kodeHoodie : int</i>	<i>getId_pembelian_produk() : int</i>
		<i>tglm : Date</i>	<i>getTglm() : Date</i>
		<i>tgls : Date</i>	<i>getTgls() : Date</i>
			<i>setTglm(Date)</i>
			<i>setTgls(Date)</i>
			<i>viewLaporan()</i>

*Tabel 66. TABEL KELAS : laporan***TABEL KELAS : reviewHoodie**

ID Kelas	Nama Kelas Perancangan	Attribute (visibility)	Method / Operation
1.	<i>Review Hoodie</i>	<i>idReview : int</i>	<i>viewReview()</i>
		<i>username : string</i>	<i>sumbitReview()</i>
		<i>rating : int</i>	
		<i>komentar : string</i>	

*Tabel 67. TABEL KELAS : reviewHoodie***TABEL KELAS : ViewReviewHoodie**

ID Kelas	Nama Kelas Perancangan	Attribute (visibility)	Method / Operation
1.	View Review Hoodie	<i>idReview : int</i>	<i>viewReview()</i>

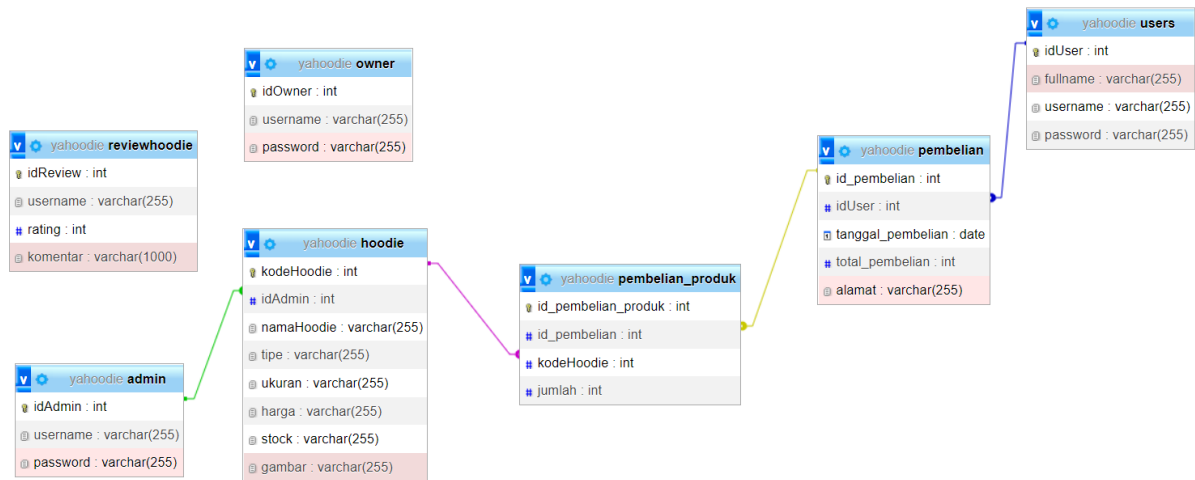
*Tabel 68. TABEL KELAS : ViewReviewHoodie*

Untuk setiap kelas:

- *identifikasi operasi (mengacu pada tanggung-jawab kelas), identifikasi atribut, termasuk visibility-nya*

### 3.3 Perancangan Data / Basis Data

Bagian ini diisi dengan penjelasan rancangan dan rencana implementasi semua Basis Data yang muncul di Diagram Kelas



Gambar 34. Tampilan Rancangan

#### 1. Tabel Admin

Nama file : *admin.sql* (File MySQL)

Tempat Penyimpanan : Harddisk

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	idAdmin	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
2	username	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
3	password	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More

Gambar 35. Tampilan Tabel Admin

## 2. Tabel Hoodie

Nama file : hoodie.sql (File MySQL)

Tempat Penyimpanan : Harddisk

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	kodeHoodie	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
2	idAdmin	int(11)			No	None			Change  Drop  More
3	namaHoodie	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
4	tipe	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
5	ukuran	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
6	harga	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
7	stock	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
8	gambar	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change  Drop  More

Gambar 36. Tampilan Tabel Hoodie

## 3. Tabel Owner

Nama file : owner.sql (File MySQL)

Tempat Penyimpanan : Harddisk

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	idOwner	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
2	username	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
3	password	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More

Gambar 37. Tampilan Tabel Owner

## 4. Tabel Pembelian

Nama file : pembelian.sql (File MySQL)

Tempat Penyimpanan : Harddisk














#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_pembelian	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
2	idUser	int(11)			No	None			Change  Drop  More
3	tanggal_pembelian	date			No	None			Change  Drop  More
4	total_pembelian	int(11)			No	None			Change  Drop  More
5	alamat	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More

Gambar 38. Tampilan Tabel Pembelian

### 5. Tabel Pembelian\_produk

Nama file : pembelian\_produk.sql (File MySQL)

Tempat Penyimpanan : Harddisk














#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_pembelian_produk 	int(11)			No	None		AUTO_INCREMENT	 Change  Drop  More
2	id_pembelian	int(11)			No	None			 Change  Drop  More
3	kodeHoodie	int(11)			No	None			 Change  Drop  More
4	jumlah	int(11)			No	None			 Change  Drop  More

Gambar 39. Tampilan Tabel Pembelian\_produk

### 6. Tabel Review Hoodie

Nama file : reviewhoodie.sql (File MySQL)

Tempat Penyimpanan : Harddisk














#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	idReview 	int(11)			No	None		AUTO_INCREMENT	 Change  Drop  More
2	username	varchar(255)	utf8mb4_general_ci		Yes	NULL			 Change  Drop  More
3	rating	int(11)			Yes	NULL			 Change  Drop  More
4	komentar	varchar(1000)	utf8mb4_general_ci		Yes	NULL			 Change  Drop  More

Gambar 40. Tampilan Tabel Review Hoodie

### 7. Tabel User

Nama file : user.sql (File MySQL)

Tempat Penyimpanan : Harddisk

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	idUser 	int(11)			No	None		AUTO_INCREMENT	 Change  Drop  More
2	fullname	varchar(255)	utf8mb4_general_ci		No	None			 Change  Drop  More
3	username	varchar(255)	utf8mb4_general_ci		No	None			 Change  Drop  More
4	password	varchar(255)	utf8mb4_general_ci		No	None			 Change  Drop  More

Gambar 41. Tampilan Tabel User

### 3.4 Perancangan Algoritma dan/atau Query

Bagian ini hanya diisi untuk kerangka algoritma untuk proses-proses yang dianggap cukup penting. Implementasi skeleton code juga sudah dapat dilakukan untuk kelas-kelas yang terdefinisi pada bahasa pemrograman tertentu. Boleh dibuat sub bab per kelas.

#### Algoritma #1

Nama Kelas : Admin

Nama Operasi : login()

Algoritma : cekloginAdmin

```
<?php
    session_start();
    if(isset($_POST['submit'])){
        include 'db.php';

        $admin = $_POST['admin'];
        $pass = $_POST['pass'];

        $cek = mysqli_query($mysqli, "SELECT * FROM admin WHERE username =
        '" . $admin . "' AND password = '" . $pass . "'");
        if(mysqli_num_rows($cek) > 0) {
            $_SESSION['username'] = $admin;
            $_SESSION['level'] = "admin";
            echo '<script>window.location="dashboardAd.php"</script>';
            return true;
        } else {
            echo '<script>alert("Username atau password anda salah!")</script>';
            return false;
        }
    }
?>
```

#### Algoritma #2

Nama Kelas : Admin

Nama Operasi : AddDataHoodie()

Algoritma : addData

```
<?php
    if(isset($_POST['tambah'])){
        $kodeHoodie = $_POST['kodeHoodie'];
        $idAdmin = $_POST['idAdmin'];
        $namaHoodie = $_POST['namaHoodie'];
        $tipe = $_POST['tipe'];
        $ukuran = $_POST['ukuran'];
        $harga = $_POST['harga'];
        $stock = $_POST['stock'];
        $gambar = $_POST['gambar'];

        $sqlGet = "SELECT * FROM hoodie WHERE kodeHoodie='$kodeHoodie';
```



```

$queryGet = mysqli_query($mysqli, $sqlGet);
$cek = mysqli_num_rows($queryGet);

if ($cek <= 0){
    $sqlInsert = "INSERT INTO
hoodie(kodeHoodie,idAdmin,namaHoodie,tipe,ukuran,harga,stock,gambar)
VALUES
('$kodeHoodie','$idAdmin','$namaHoodie','$tipe','$ukuran','$harga','$stock','$gambar')";
    $queryInsert=mysqli_query($mysqli, $sqlInsert);
    echo "
        <div class='alert alert-success'>Data berhasil ditambahkan<a href='dataHoodie.php'>
Lihat Data </a></div>
        ";
    }else {
        echo "
        <div class='alert alert-danger'>Data gagal ditambahkan<a href='dataHoodie.php'> Lihat
Data </a></div>
        ";
    }
}
?>

```

### Algoritma #3

*Nama Kelas : Admin*

*Nama Operasi : UpdateDataHoodie()*

*Algoritma : updateData*

```

<?php
function UpdateData() {
    include "db.php";
    if(isset($_POST['tambah'])){
        $kodeHoodie = $_POST['kodeHoodie'];
        $idAdmin = $_POST['idAdmin'];
        $namaHoodie = $_POST['namaHoodie'];
        $tipe = $_POST['tipe'];
        $ukuran = $_POST['ukuran'];
        $harga = $_POST['harga'];
        $stock = $_POST['stock'];
        $gambar = $_POST['gambar'];

        $sqlUpdate = "UPDATE hoodie
            SET idAdmin='$idAdmin', namaHoodie='$namaHoodie',
tipe='$tipe', ukuran='$ukuran', harga='$harga', stock='$stock', gambar='$gambar'
            WHERE kodeHoodie='$kodeHoodie'";
        $queryUpdate = mysqli_query($mysqli, $sqlUpdate);

        header("location: dataHoodie.php");
    }
}

```

```
}  
?>
```

#### Algoritma #4

*Nama Kelas : Admin*

*Nama Operasi : DeleteDataHoodie()*

*Algoritma : deleteData*

```
<?php  
    include 'db.php';  
    $kodeHoodie = $_GET['kodeHoodie'];  
    $sqlDelete = "DELETE FROM hoodie WHERE kodeHoodie='$kodeHoodie'";  
    mysqli_query($mysqli, $sqlDelete);  
  
    header("location: dataHoodie.php");  
?>
```

#### Algoritma #5

*Nama Kelas : Admin*

*Nama Operasi : DeleteDataCustomer()*

*Algoritma : deleteCustomer*

```
<?php  
    include 'db.php';  
    $idUser = $_GET['idUser'];  
    $sqlDelete = "DELETE FROM users WHERE idUser='$idUser'";  
    mysqli_query($mysqli, $sqlDelete);  
  
    header("location: dataPelanggan.php");  
?>
```

#### Algoritma #6

*Nama Kelas : Admin*

*Nama Operasi : viewSalesReport()*

*Algoritma : dataPembelian*

```
<?php  
    $nomor = 1;  
    include 'db.php';  
    $hasil = mysqli_query($mysqli, "SELECT * FROM pembelian JOIN users ON  
pembelian.idUser=users.idUser");  
    while ($pecah = $hasil->fetch_assoc()) {  
?>  
        <tr>  
            <td><?php echo $nomor; ?></td>  
            <td><?php echo $pecah['fullname']; ?></td>
```

```

<td><?php echo $specah['tanggal_pembelian']; ?></td>
<td>Rp. <?php echo number_format($specah['total_pembelian']);?></td>
<td>
    <a class="btn btn-info" href="detail.php?halaman=detail&id=<?php echo
$specah['id_pembelian'];?>">Detail</a>
</td>
</tr>
<?php $nomor++;?>
<?php } ?>

```

#### Algoritma #7

Nama Kelas : User

Nama Operasi : registrasi() dan login()

Algoritma : cekloginUser

```

<?php
session_start();
include 'db.php';
if(isset($_POST['submit']))
{
    $user = $_POST['user'];
    $pass = $_POST['pass'];
    $sql = "SELECT * FROM users WHERE username = '". $user.'" AND password
= '". $pass."'";
    $cek = mysqli_query($mysqli, $sql);
    if(mysqli_num_rows($cek) > 0)
    {
        $akun = $cek->fetch_assoc();
        $_SESSION['pelanggan'] = $akun;
        $_SESSION['username'] = $user;
        $_SESSION['level'] = "user";
        $_SESSION['cart'];
        echo '<script>>window.location="dashboard.php"</script>';
    }
    else
    {
        echo '<script>alert("Username atau password anda salah!")</script>';
    }
}
else if(isset($_POST['regist']))
{
    $fullname = $_POST['fullname'];
    $username = $_POST['username'];
    $password = $_POST['password'];

    $sql = "SELECT * FROM users WHERE username='$username'";
    $result = mysqli_query($mysqli, $sql);

```

```

        if (!$result->num_rows > 0)
        {
            $sql = "INSERT INTO users (idUser, fullname, username, password)
VALUES (NULL, ".$fullname.", ".$username.", ".$password.)";
            $result = mysqli_query($mysqli, $sql);

            if($result)
            {
                echo "<script>alert('Selamat, registrasi berhasil!')</script>";
                echo '<script>>window.location="loginUser.html"</script>';
            }
            else
            {
                echo "<script>alert('Woops! Terjadi kesalahan.')</script>";
            }
        }
        else
        {
            echo "<script>alert('Woops! Username Sudah Terdaftar.')</script>";
        }
    }
?>

```

#### Algoritma #8

*Nama Kelas : Pembelian*

*Nama Operasi : AddtoCart()*

*Algoritma : product*

```

<?php
    include 'myfunctions.php';
    $hoodie = getAll('hoodie');

    if (mysqli_num_rows($hoodie) > 0)
    {
        foreach($hoodie as $item)
        {
            ?>
            <div class="col-sm-6 col-md-4 col-lg-4">
                <div class="box">
                    <div class="option_container">
                        <div class="options">
                            <a href="beli.php?id=<?php echo $item['kodeHoodie'];
?>" class="option2">
                                Add to Cart
                            </a>
                        </div>
                    </div>
                </div>
            </div>
        }
    }

```

```

        <div class="img-box">
            
        </div>

        <div class="detail-box">
            <h5><?= $item['namaHoodie']; ?> <p style="text-align:left;
margin-top:20px;"> <?= $item['ukuran']; ?> </p></h5>
            <h6><?php echo number_format($item['harga']);?></h6>
        </div>
    </div>
</div>

<?php
    }
}
else
{
    echo "No data available";
}
?>

```

#### Algoritma #9

Nama Kelas : Pembelian

Nama Operasi : Checkout()

Algoritma : checkout

```

<?php
    $nomor = 1;
    $totalbelanja = 0;
?>
<?php foreach($_SESSION['keranjang'] as $id_produk => $jumlah):?>
<?php
    $hasil = mysqli_query($mysqli, "SELECT * FROM hoodie WHERE
kodeHoodie = '". $id_produk ."'");
    $pecah = $hasil->fetch_assoc();
    $subharga = $pecah['harga']*$jumlah;
?>
<tr>
    <td><?php echo $nomor?></td>
    <td><?php echo $pecah['namaHoodie'];?></td>
    <td><?php echo $pecah['ukuran'];?></td>
    <td>Rp. <?php echo number_format($pecah['harga']);?></td>
    <td><?php echo $jumlah?></td>
    <td>Rp. <?php echo number_format($subharga);?></td>
</tr>
<?php $nomor++;?>
<?php $totalbelanja+=$subharga;?>

```

```
<?php endforeach?>

?>
```

#### Algoritma #10

*Nama Kelas : Pembelian*

*Nama Operasi : viewNota()*

*Algoritma : nota*

```
<?php
    include 'db.php';
    $hasil = mysqli_query($mysqli, "SELECT * FROM pembelian JOIN
users ON pembelian.idUser=users.idUser WHERE
pembelian.id_pembelian='$_GET[id]'");
    $detail = $hasil->fetch_assoc();
    $sidpelangganyangbeli = $detail['idUser'];

    $sidpelangganyanglogin = $_SESSION['pelanggan']['idUser'];
    if ($sidpelangganyangbeli !== $sidpelangganyanglogin)
    {
        echo '<script>alert("Anda tidak memiliki akses ke halaman
ini")</script>';
        echo "<script>location='riwayat.php';</script>";
        exit();
    }
?>
```

#### Algoritma #11

*Nama Kelas : reviewHoodie*

*Nama Operasi : submitReview()*

*Algoritma : addReview*

```
<?php

//submit_rating.php

$connect = new PDO("mysql:host=localhost:3307;dbname=yahoodie", "root",
"Arya312610@");

if(isset($_POST["rating_data"]))
{

    $data = array(
        ':idReview' => NULL,
```

```

        ':user_name'    => $_POST["user_name"],
        ':user_rating'  => $_POST["rating_data"],
        ':user_review'  => $_POST["user_review"],
    );

    $query = "
    INSERT INTO reviewhoodie
    (idReview, username, rating, komentar)
    VALUES (:idReview, :user_name, :user_rating, :user_review)
    ";

    $statement = $connect->prepare($query);

    $statement->execute($data);

    echo "Your Review & Rating Successfully Submitted";

}

if(isset($_POST["action"]))
{
    $average_rating = 0;
    $total_review = 0;
    $five_star_review = 0;
    $four_star_review = 0;
    $three_star_review = 0;
    $two_star_review = 0;
    $one_star_review = 0;
    $total_user_rating = 0;
    $review_content = array();

    $query = "
    SELECT * FROM reviewhoodie
    ORDER BY idReview DESC
    ";

    $result = $connect->query($query, PDO::FETCH_ASSOC);

    foreach($result as $row)
    {
        $review_content[] = array(
            'user_name'    => $row["username"],
            'user_review'  => $row["komentar"],
            'rating'       => $row["rating"],
        );

        if($row["rating"] == '5')
        {
            $five_star_review++;
        }
    }
}

```

```

    }

    if($row["rating"] == '4')
    {
        $four_star_review++;
    }

    if($row["rating"] == '3')
    {
        $three_star_review++;
    }

    if($row["rating"] == '2')
    {
        $two_star_review++;
    }

    if($row["rating"] == '1')
    {
        $one_star_review++;
    }

    $total_review++;

    $total_user_rating = $total_user_rating + $row["rating"];
}

$average_rating = $total_user_rating / $total_review;

$output = array(
    'average_rating' => number_format($average_rating, 1),
    'total_review' => $total_review,
    'five_star_review' => $five_star_review,
    'four_star_review' => $four_star_review,
    'three_star_review' => $three_star_review,
    'two_star_review' => $two_star_review,
    'one_star_review' => $one_star_review,
    'review_data' => $review_content
);

echo json_encode($output);
}

?>

```



## Algoritma #12

Nama Kelas : Owner

Nama Operasi : viewLaporan()

Algoritma : laporan

```
<head>
  <meta charset="utf-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1,
shrink-to-fit=no" />
  <meta name="description" content="" />
  <meta name="author" content="" />
  <title>Laporan - YA-HOODIE Owner</title>
  <link href="https://cdn.jsdelivr.net/npm/simple-datatables@latest/dist/style.css"
rel="stylesheet" />
  <link href="css/styles-sb-admin.css" rel="stylesheet" />
  <script
src="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/js/all.min.js"
crossorigin="anonymous"></script>
</head>
<body>
  <?php
    include 'db.php';
    $semuadata=array();
    $tgl_mulai='-';
    $tgl_selesai='-';
    if(isset($_POST[" kirim"]))
    {
      $tgl_mulai = $_POST["tglm"];
      $tgl_selesai = $_POST["tgls"];
      $ambil = $mysqli->query("SELECT * FROM users us INNER
JOIN pembelian pm ON us.idUser=pm.idUser INNER JOIN pembelian_produk pp
ON pm.id_pembelian=pp.id_pembelian INNER JOIN hoodie hd ON
pp.kodeHoodie=hd.kodeHoodie WHERE tanggal_pembelian BETWEEN
'$tgl_mulai' AND '$tgl_selesai'
");
      while ($specah = $ambil->fetch_assoc())
      {
        $semuadata[] = $specah;
      }
    }
  ?>
  <main>
    <div class="container-fluid px-4">
      <h3 class="mt-4">Laporan Penjualan</h3>
      <hr>
      <div>
        <form method="POST">
          <div class="row">
```

```

<div class="col-md-5">
  <div class="form-group">
    <label>Tanggal Mulai</label>
    <input type="date" class="form-control" name="tglm">
  </div>
</div>
<div class="col-md-5">
  <div class="form-group">
    <label>Tanggal Selesai</label>
    <input type="date" class="form-control" name="tgls">
  </div>
</div>
<div class="col-md-2">
  <label>&nbsp;</label><br>
  <button class="btn btn-primary" name=" kirim">Lihat
Laporan</button>
</div>
</div>
</form>
<br>
<br>
<div class="card mb-4">
  <div class="card-header">
    <i class="fas fa-table me-1"></i>
    Laporan Pembelian dari <?php echo $tgl_mulai?> sampai <?php
echo $tgl_selesai?>
  </div>
  <div class="card-body">
    <table id="datatablesSimple">
      <thead>
        <tr>
          <th>NO</th>
          <th>Pelanggan</th>
          <th>Tanggal</th>
          <th>Produk</th>
          <th>Harga</th>
          <th>Jumlah</th>
          <th>Total</th>
        </tr>
      </thead>
      <tfoot>
        <tr>
          <th>NO</th>
          <th>Pelanggan</th>
          <th>Tanggal</th>
          <th>Produk</th>
          <th>Harga</th>
          <th>Jumlah</th>
          <th>Total</th>
        </tr>
      </tfoot>
    </table>
  </div>
</div>

```

```

        </tr>
    </tfoot>
    <tbody>
        <?php
            $total=0;
            foreach($semuadata as $key => $value):
                $subharga = $value['harga']*$value['jumlah'];
            ?>
        <tr>
            <td><?php echo $key+1; ?></td>
            <td><?php echo $value['fullname']; ?></td>
            <td><?php echo $value['tanggal_pembelian']; ?></td>
            <td><?php echo $value['namaHoodie']; ?></td>
            <td><?php echo $value['harga']; ?></td>
            <td><?php echo $value['jumlah']; ?></td>
            <td><?php echo $subharga; ?></td>
        </tr>
        <?php $total+=$subharga;?>
        <?php endforeach?>
    </tbody>
    <tfoot>
        <tr>
            <th colspan="6">Total Penjualan</th>
            <th>Rp. <?php echo number_format($total); ?></th>
        </tr>
    </tfoot>
</table>
</div>
</div>
</div>
</div>
</main>

```

### Query #1

Nama Kelas : Admin

Nama Operasi : login()

Query :

"SELECT \* FROM admin WHERE username = ".\$admin." AND password = ".\$pass."

### Query #2

Nama Kelas : Admin

Nama Operasi : AddDataHoodie()

Query :

"SELECT \* FROM hoodie WHERE kodeHoodie='".\$kodeHoodie'"

"INSERT INTO hoodie(kodeHoodie,idAdmin,namaHoodie,tipe,ukuran,harga,stock,gambar)

VALUES ('\$kodeHoodie','\$idAdmin','\$namaHoodie','\$tipe','\$ukuran','\$harga','\$stock','\$gambar')

### Query #3

*Nama Kelas : Admin*

*Nama Operasi : UpdateDataHoodie()*

*Query :*

"UPDATE hoodie SET idAdmin='\$idAdmin', namaHoodie='\$namaHoodie', tipe='\$tipe', ukuran='\$ukuran', harga='\$harga', stock='\$stock', gambar='\$gambar' WHERE kodeHoodie='\$kodeHoodie'"

### Query #4

*Nama Kelas : Admin*

*Nama Operasi : DeleteDataHoodie()*

*Query :*

"DELETE FROM hoodie WHERE kodeHoodie='\$kodeHoodie'"

### Query #5

*Nama Kelas : Admin*

*Nama Operasi : DeleteDataCustomer()*

*Query :*

"DELETE FROM users WHERE idUser='\$idUser'"

### Query #6

*Nama Kelas : Admin*

*Nama Operasi : viewSalesReport()*

*Query :*

"SELECT \* FROM pembelian JOIN users ON pembelian.idUser=users.idUser"

### Query #7

*Nama Kelas : User*

*Nama Operasi : registrasi() dan login()*

*Query :*

"SELECT \* FROM users WHERE username = '\$user' AND password = '\$pass.'"

"SELECT \* FROM users WHERE username='\$username';

"INSERT INTO users (idUser, fullname, username, password) VALUES (NULL, '\$fullname', '\$username', '\$password.)"

### Query #8

*Nama Kelas : Pembelian*  
*Nama Operasi : AddtoCart()*  
*Query : -*

#### **Query #9**

*Nama Kelas : Pembelian*  
*Nama Operasi : Checkout()*  
*Query :*

"SELECT \* FROM hoodie WHERE kodeHoodie = ".\$id\_produk."

#### **Query #10**

*Nama Kelas : Pembelian*  
*Nama Operasi : viewNota()*  
*Query :*

"SELECT \* FROM pembelian JOIN users ON pembelian.idUser=users.idUser WHERE pembelian.id\_pembelian='\$\_GET[id]'"

#### **Query #11**

*Nama Kelas : reviewHoodie*  
*Nama Operasi : submitReview()*  
*Query :*

" INSERT INTO reviewhoodie (idReview, username, rating, komentar) VALUES (:idReview, :user\_name, :user\_rating, :user\_review)"

"SELECT \* FROM reviewhoodie ORDER BY idReview DESC"

#### **Query #12**

*Nama Kelas : Owner*  
*Nama Operasi : viewLaporan()*  
*Query :*

"SELECT \* FROM users us INNER JOIN pembelian pm ON us.idUser=pm.idUser INNER JOIN pembelian\_produk pp ON pm.id\_pembelian=pp.id\_pembelian INNER JOIN hoodie hd ON pp.kodeHoodie=hd.kodeHoodie WHERE tanggal\_pembelian BETWEEN '\$tgl\_mulai' AND '\$tgl\_selesai'"

## **4. Matriks Keruntutan (Requirement Traceability Matrix)**

<b>Prodi S1 Informatika Tel-U</b>	<b>DPPL-XXX</b>	<b>Halaman 85 dari 86</b>
Template dokumen ini dan informasi yang dimilikinya adalah milik Prodi S1 Informatika Tel-U dan bersifat rahasia. Dilarang me-reproduksi dokumen ini tanpa diketahui oleh Prodi S1 Informatika Tel-U.		

### Mapping Requirement dengan Use Case yang direalisasikan

Kode FR	Nama Functional Requirement	Nama Use Case
FR-01	Login User	Login User
FR-02	Login Admin	Login Admin
FR-03	Login Owner	Login Owner
FR-04	Register	Register
FR-05	Kelola Hoodie	Kelola Hoodie
FR-06	Detail Pembelian	Detail Pembelian
FR-07	Laporan Penjualan	Laporan Penjualan
FR-08	Order Hoodie	Order Hoodie
FR-09	Review	Review
FR-10	Statistik Review	Statistik Review
FR-11	Delete Customer	Delete Customer
FR-12	Logout User	Logout User
FR-13	Logout Admin	Logout Admin
FR-14	Logout Owner	Logout Owner

*Tabel 69. Matriks Kerunutan (Requirement Traceability Matrix)*

### Lampiran

#### Progress

<https://docs.google.com/document/d/1UM8gn8vf-7qMWP8pHvO2B0WOZ8gjLYMN/edit?usp=sharing&ouid=105015904708726665666&rtpof=true&sd=true>