

DPPL-xx

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

STEVENT

untuk:

Telkom University

Dipersiapkan oleh:

Athallah Tsany Rakha Dzaky (1301174026)

Muhammad Alfhi Saputra (1301171205)


Kartika Putri Dwi Atmojo (1301170084)

Indri Islamiawati Putri (1301170458)

Program Studi Informatika

Fakultas Informatika

Jl. Telekomunikasi 1, Dayeuhkolot Bandung

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

DAFTAR PERUBAHAN

Revisi	Deskripsi
A	
B	
C	
D	
E	
F	
G	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperiksa oleh								

Disetujui oleh								
-------------------	--	--	--	--	--	--	--	--

Daftar Halaman Perubahan

Halaman	Revisi	Halaman	Revisi

Daftar Isi

1. Pendahuluan

Tujuan Penulisan Dokumen

Lingkup Masalah

Definisi dan Istilah

Referensi

Sistematika Pembahasan

2. Deskripsi Perancangan Global

Rancangan Lingkungan Implementasi

Deskripsi Arsitektural

Deskripsi Komponen

3. Perancangan Rinci

Realisasi Use Case

Use Case <nama use case 1>

Identifikasi Kelas

Sequence Diagram

Diagram Kelas

Perancangan Detil Kelas

Kelas <nama kelas>

Kelas <nama kelas>

Diagram Kelas Keseluruhan

Algoritma/Query

Diagram Statechart

Perancangan Antarmuka

Perancangan Representasi Persistensi Kelas

4. Matriks Kerunutan

1. Pendahuluan

1.1 Tujuan Penulisan Dokumen

Tujuan penulisan dokumen ini adalah untuk digunakan sebagai referensi bagi pihak pengembang dan pengguna. Bagi pengembang dokumen ini berguna sebagai acuan untuk pengembangan dan implementasi aplikasi, sedangkan bagi pengguna dokumen ini digunakan sebagai referensi untuk mengetahui bagaimana sistem bekerja lengkap dengan spesifikasi dan kebutuhan perangkat.

1.2 Lingkup Masalah

Lingkup masalah pada penjualan event online, antara lain adalah pemaparan event-event, proses jual beli event yang disertai proses verifikasi dengan melakukan pengiriman bukti online.

1.3 Definisi dan Istilah

Semua definisi dan singkatan yang digunakan dalam dokumen ini dan penjelasannya

1.4 Referensi

Dokumen yang digunakan sebagai acuan dalam pembuatan dokumen ini adalah :

1. DPPL E-learning Universitas Telkom 2011
2. SKPL Stevent, Universitas Telkom 2019

1.5 Sistematika Pembahasan

Dokumen perancangan perangkat lunak penjualan tiket event online ini berisikan rincian dari deskripsi pengembangan perangkat lunak. Sistematika pembahasan yang akan dilakukan dikelompokkan adalah sebagai berikut:

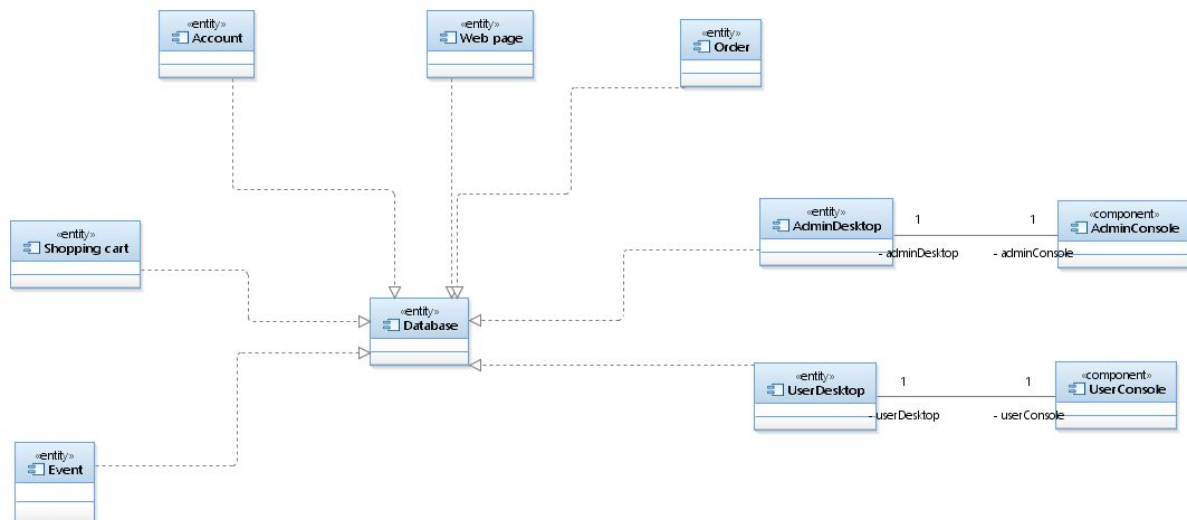
1. Bab 1 pendahuluan, berisi tujuan penulisan dokumen, lingkup masalah yang ditangani dalam perangkat lunak yang akan dibangun serta deskripsi umum dokumen.
2. Bab 2 deskripsi perancangan global, berisikan uraian singkat mengenai perangkat lunak yang akan dikembangkan dan bagaimana rancangan aplikasi itu dibuat.
3. Bab 3 rincian perancangan, berisikan detail komponen dan fitur yang akan diimplementasikan pada perangkat lunak. Bagian ini terdiri dari
4. Bab 4 matriks keruntutan

2. Deskripsi Perancangan Global

2.1 Rancangan Lingkungan Implementasi

Jenis Sistem	Spesifikasi
Sistem Operasi	Windows 7, 8, 10
DBMS	MySQL
Bahasa Pemrograman	Javascript, PHP
Browser	Google Chrome 73.0.3683 atau keatas

2.2 Deskripsi Arsitektural



Website yang kami buat menggunakan konsep MVC (Model View Controller). Inti dari MVC adalah instruksi yang diberikan pengguna melalui *view* dikirim dan di proses di controller. Jika instruksi tersebut membutuhkan database maka controller akan mengakses *Model* untuk mendapatkan data dari database yang selanjutnya akan dikirim kembali ke view sebagai respon dari instruksi yang diberikan oleh pengguna.

Berikan penjelasan singkat tentang arsitektur /L yang akan dibangun. Gambarkan dalam bentuk diagram komponen.

2.3 Deskripsi Komponen

Diisi dengan daftar modul. Daftar modul bisa dalam bentuk tabel berikut:

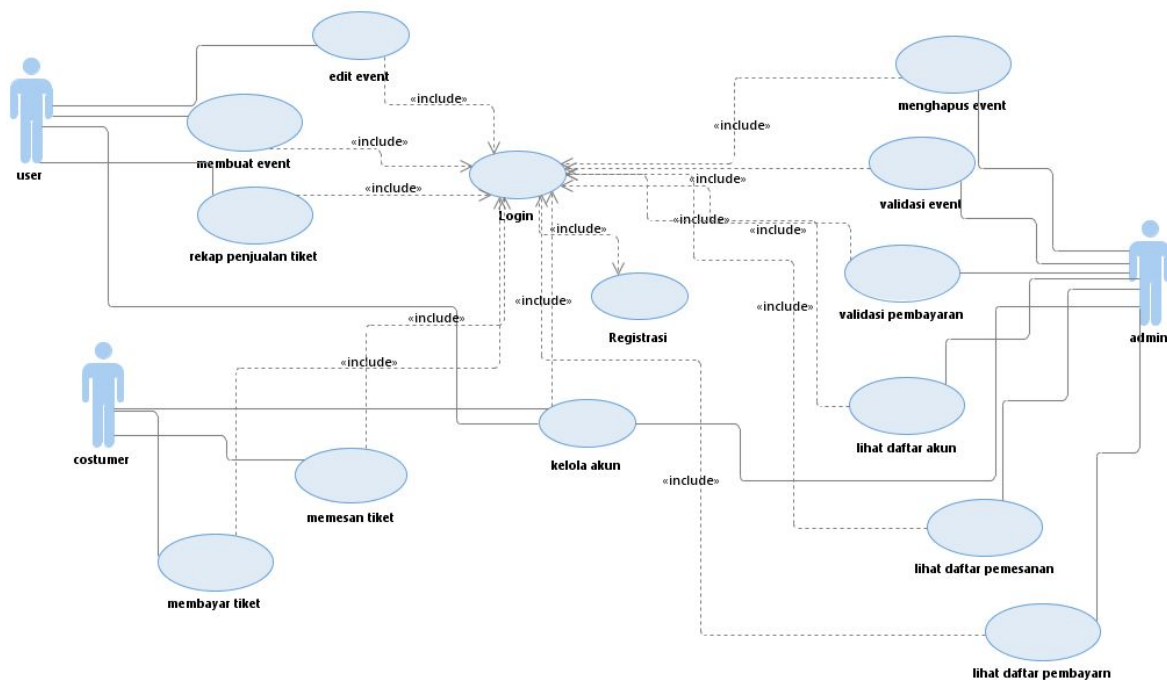
No	Nama Komponen	Keterangan
1	Account	Database yang menyimpan data akun

2	Event	Database yang menyimpan data Event
4	Web Page	Server / Hosting
3	Shopping Cart	Pesanan yang belum dibayar / terverifikasi
4	Order	History pesanan
5	Admin Desktop	Halaman view Admin
6	Admin Console	Controller Admin
7	User Desktop	Halaman view User
8	User Console	Controller Console
9	Customer Desktop	Halaman view Customer
10	Customer Console	Controller Customer

3. Perancangan Rinci

3.1 Realisasi Use Case

3.1.1 Use Case Diagram



3.1.2 Use Case Scenario

- Nama Use case : Login
 Aktor : User, Customer, Admin
 Pre-condition : Aktor belum masuk ke sistem
 Post-condition : Aktor masuk ke sistem dengan identitas masing-masing

Deskripsi : Aktor melakukan login ke dalam sistem

Tabel skenario

Aktor	Sistem
	1. Menampilkan laman login
2. Menginput data akun	
	3. Memvalidasi data login
4. Sudah masuk ke dalam sistem	

2. Nama Use case : Membuat event

Aktor : User

Pre-condition : Event belum terdaftar

Post-condition : Event telah terdaftar

Deskripsi : Aktor mengisi data event yang akan dijual tiketnya

Tabel skenario

Aktor	Sistem
1. Memilih menu daftar event	
	2. Menampilkan form daftar event
3. Mengisi data event	
	4. Memvalidasi data event
	5. Menampilkan event ke lama web

3. Nama Use case : Kelola akun

Aktor : User, Customer, Admin

Pre-condition : Komponen akun belum berubah

Post-condition : Komponen akun berhasil diubah

Deskripsi : Aktor mengubah poin-poin dalam akunnya sesuai *privilege* aktor

Tabel skenario

Aktor	Sistem
1. Memilih menu profil	
	2. Menampilkan informasi profil
3. Memilih opsi edit	
	4. Menampilkan form profil yang dapat diubah

5. Mengedit bagian-bagian profil yang akan dirubah	
6. Menyimpan hasil perubahan	
	7. Memperbarui <i>database</i> yang ada

4. Nama Use case :Registrasi
Aktor :User, Customer, Admin
Pre-condition :aktor belum memiliki akun
Post-condition :aktor telah memiliki akun
Deskripsi :aktor membuat akun melalui laman registrasi
Tabel skenario

Aktor	Sistem
	1. Menampilkan laman registrasi
2. Menginput data registrasi aktor	
	3. Memvalidasi data registrasi
	4. Validasi berhasil
	5. Data user masuk database
6. Aktor memiliki akun	

5. Nama Use case : Memesan tiket
Aktor : Customer
Pre-condition : Customer belum memiliki tiket
Post-condition : Customer sudah memiliki chart tiket
Deskripsi : Customer melakukan pemilihan event yang akan dibeli tiketnya
Tabel skenario

Aktor	Sistem
1. Memilih event	
	2. Menampilkan detail / deskripsi event
3. Memilih button untuk memesan tiket	
	4. Validasi data input
	5. Jika data salah, data akan dikembalikan dan memunculkan notif kepada customer

	6. Jika data benar, data akan disimpan kedalam daftar pemesanan
	7. Mengirim data tagihan kepada customer

6. Nama Use case : Validasi Event
Aktor : Admin
Pre-condition : Admin telah login dan ingin mem-validasi suatu event yang di inputkan oleh user
Post-condition : Admin telah mem-validasi suatu event
Deskripsi : Admin menentukan event yang akan di post di daftar event
Tabel skenario

Aktor	Sistem
1. Memilih event	
	2. Menampilkan detail / deskripsi event
3. Mem-validasi event	
4. Jika event ter-validasi	
	5. Data event di post kedalam daftar event
6. Jika event tidak ter-validasi	
	7. Akan ada notif balik ke user, agar memperbaiki data event tersebut.

7. Nama Use case : Melihat Event
Aktor : user, admin, customer
Pre-condition : aktor ingin mengetahui info/data event
Post-condition : aktor telah mendapatkan info/data event
Deskripsi : aktor telah melihat info/data event dan mendapatkan info
Tabel skenario

Aktor	Sistem
	1. Menampilkan laman berisi daftar event
2. Mengakses event yang sesuai	
	3. Menampilkan laman info/data event yang telah dipilih
4. Mendapatkan info/data event	

8. Nama Use case : Menghapus event
 Aktor : Admin
 Pre-condition : Event masih terdaftar
 Post-condition : Event telah dihapus
 Deskripsi : Admin menghapus event yang sudah terlaksana atau lewat masa batas
 Tabel skenario

Aktor	Sistem
1. Mengakses data event	
2. Menghapus event	
	3. Event tidak ada di sistem

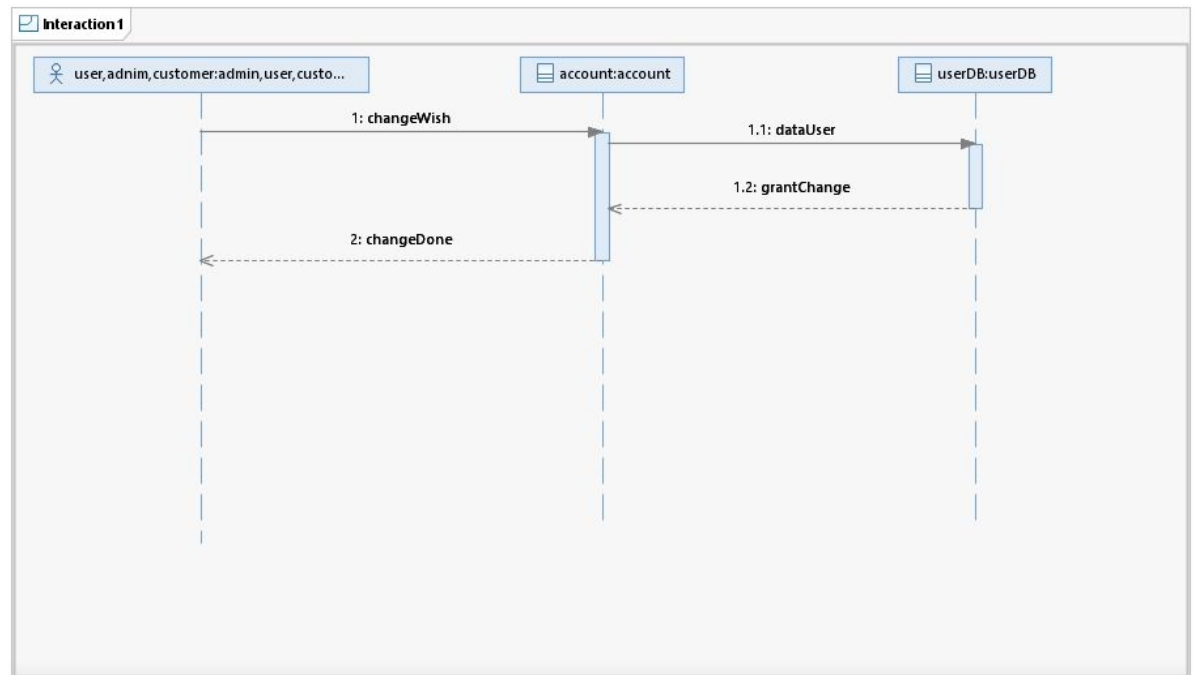
3.2 Identifikasi Kelas

No	Nama Kelas Perancangan	Tipe Kelas
1	Akun	Controller
2	Pengguna	Controller
3	Pelanggan	Controller
4	Admin	Controller
5	Pembayaran	Controller
6	Tiket	Controller
7	Event	Controller

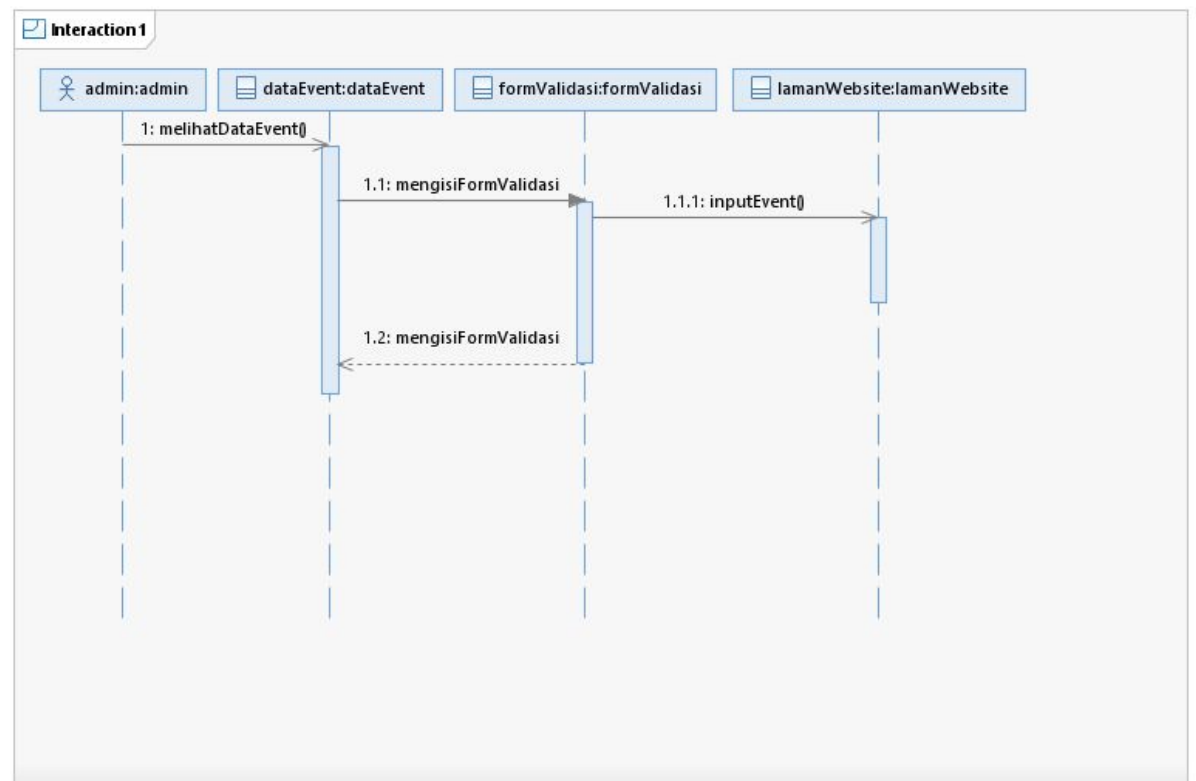
**Tipe kelas seperti Boundary(Interface), Entity(Database), Controller*

3.3 Sequence Diagram

1. Kelola akun



2. Validasi Event



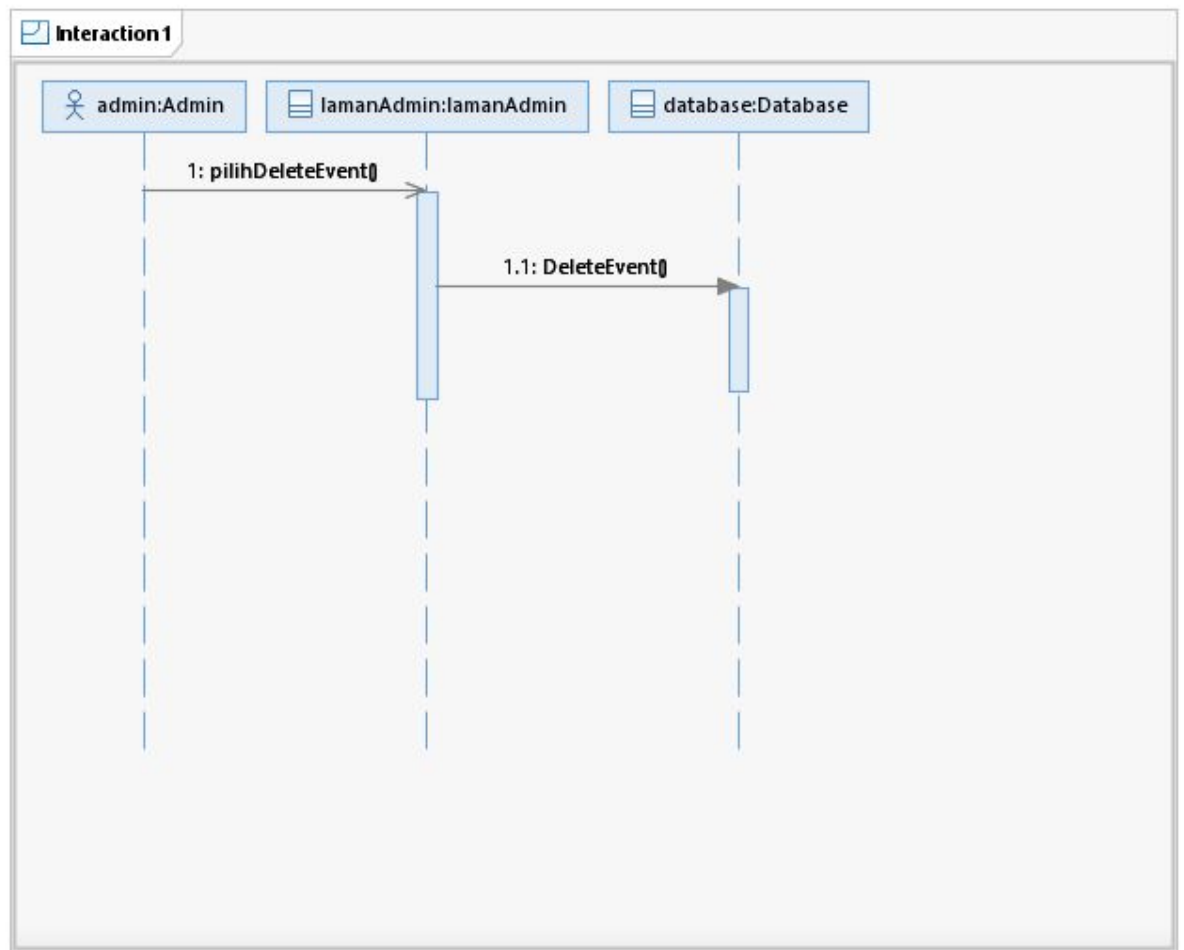
3. Memesan Tiket



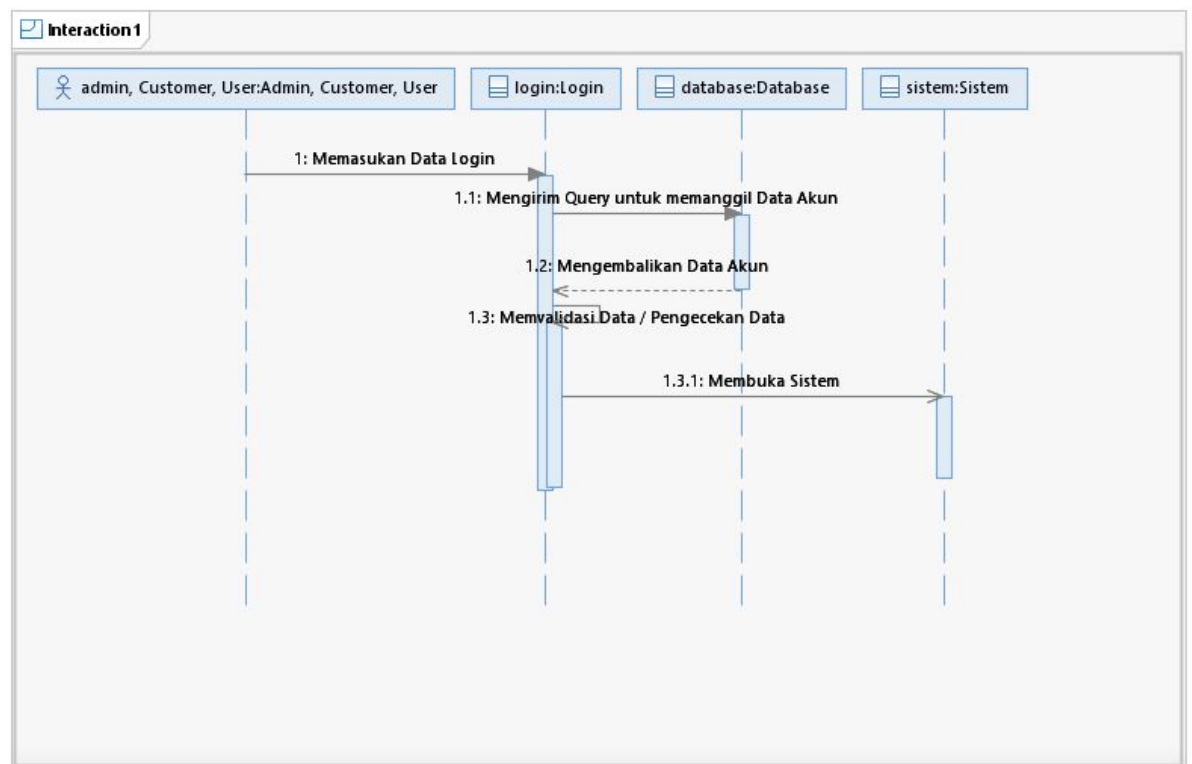
4. Buat Event



5. Hapus Event

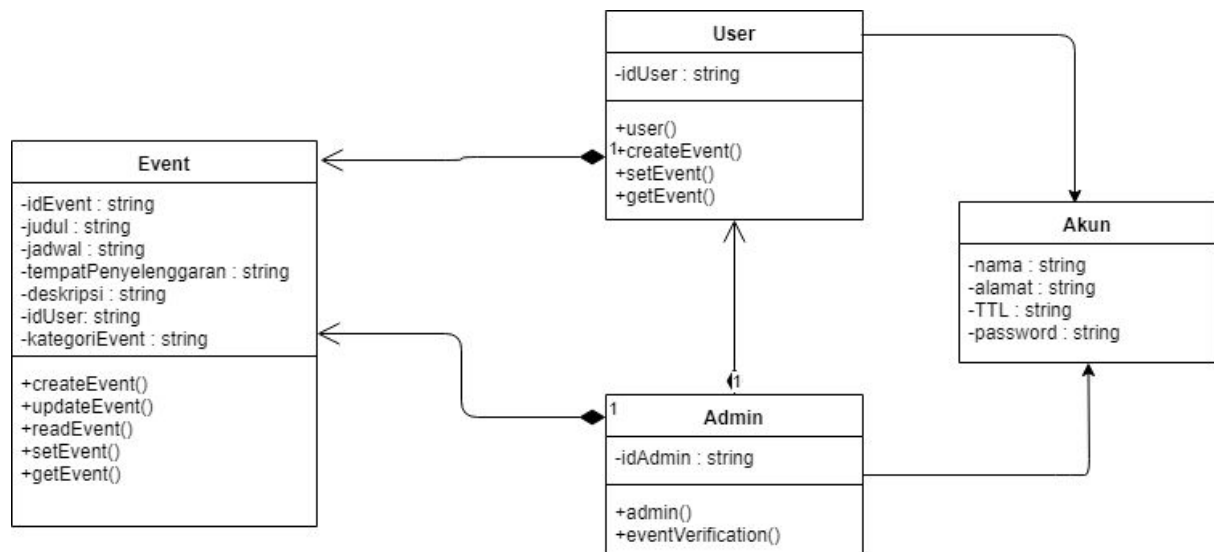


6. Login

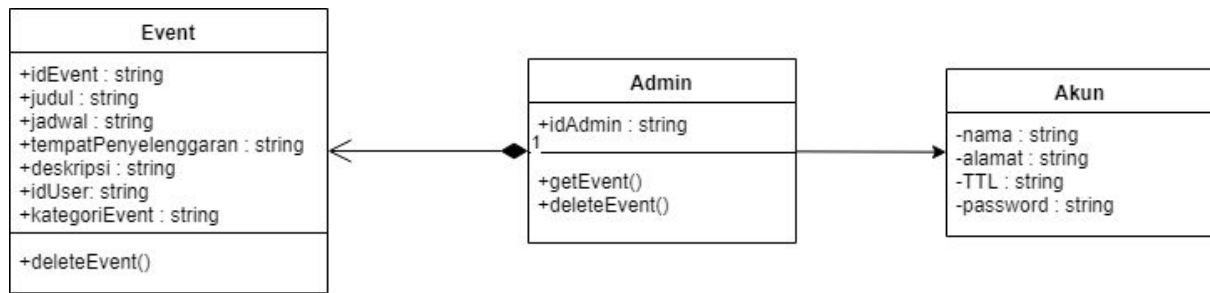


3.4 Diagram Kelas

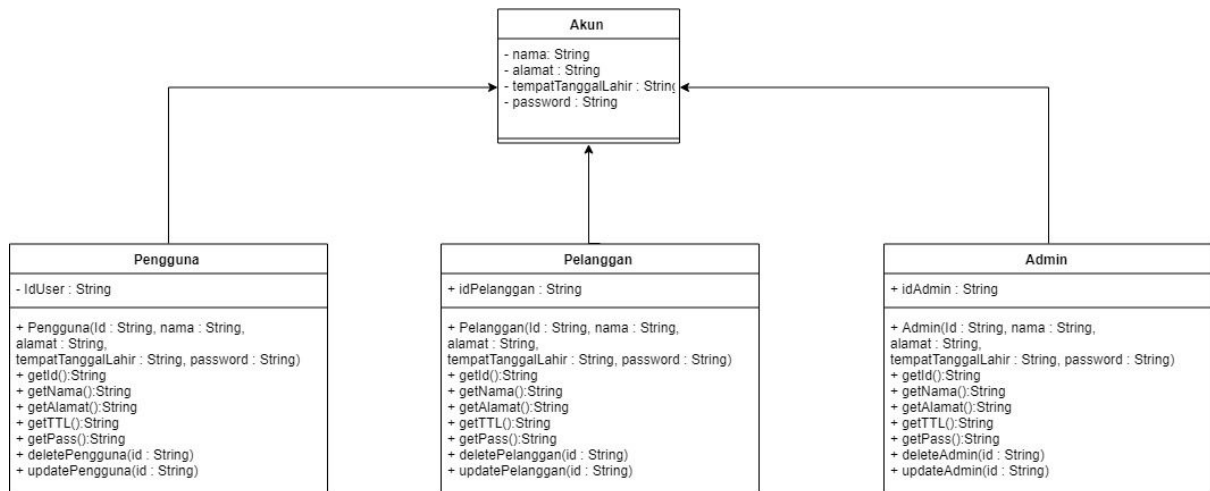
1. Create Event



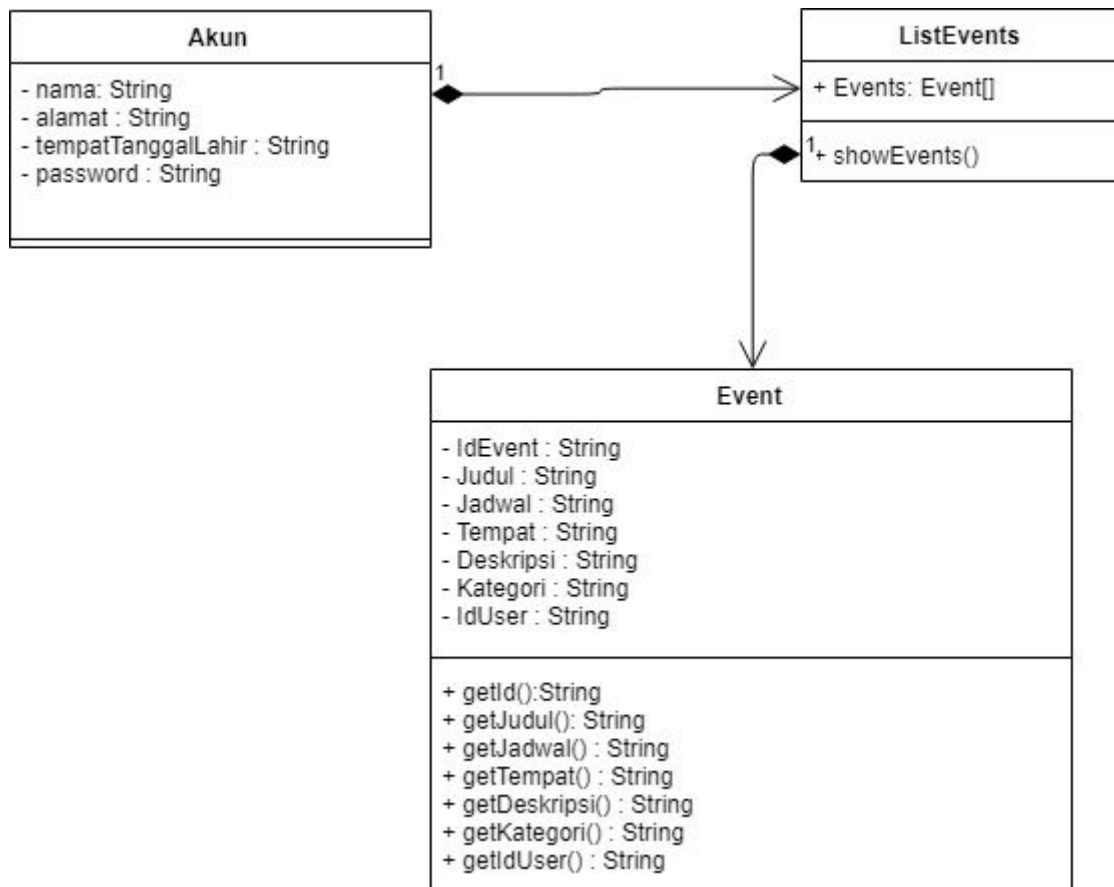
2. Delete Event



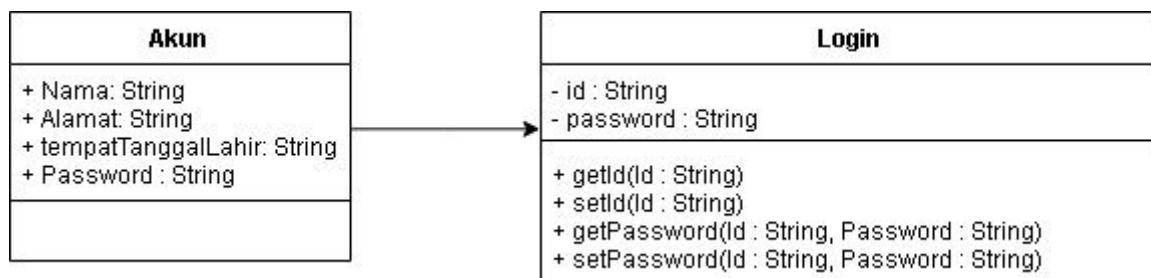
3. Kelola Akun



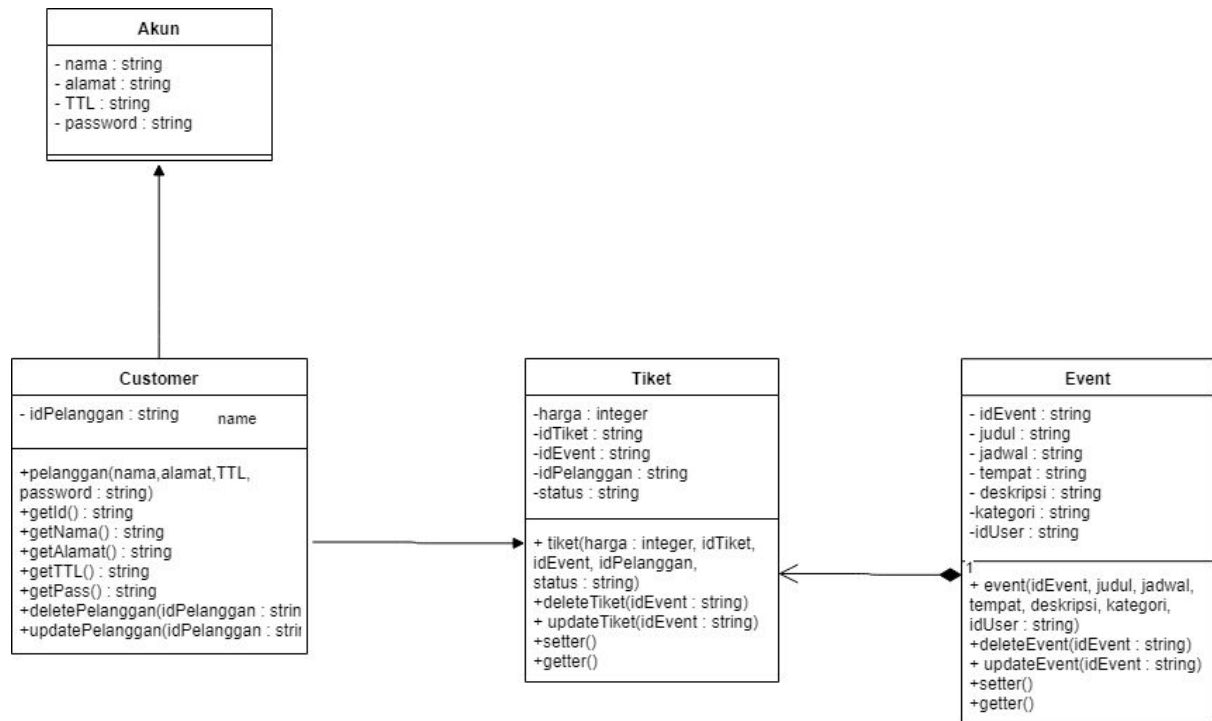
4. Lihat Event



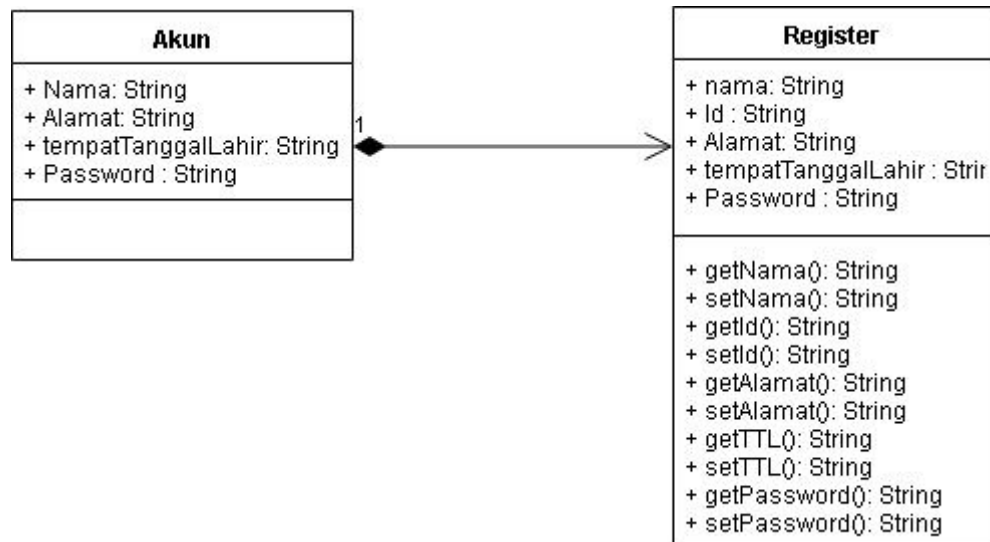
5. Login



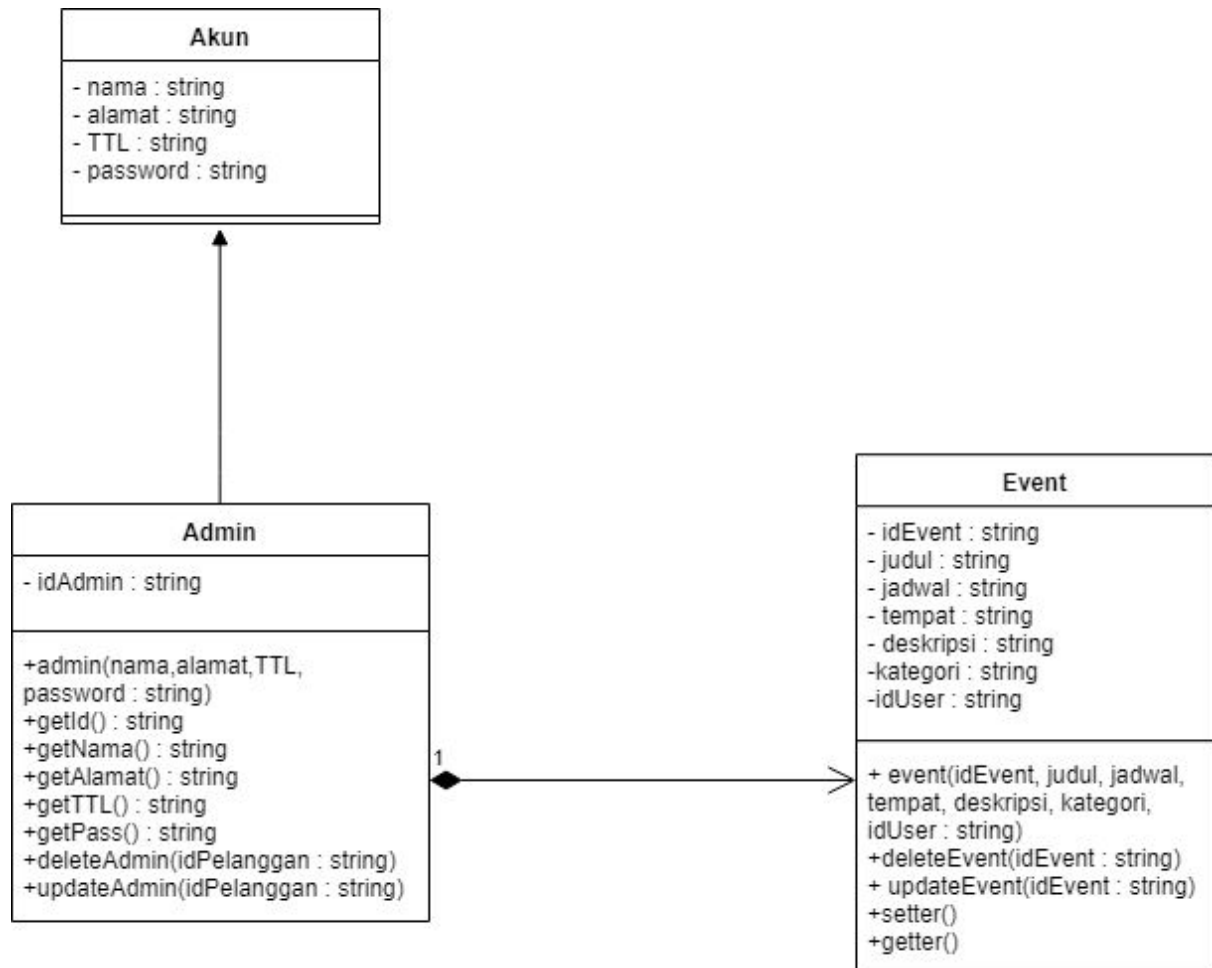
6. Order Tiket



7. Registrasi



8. Validasi



3.5 Perancangan Detail Kelas

No	Nama Kelas Perancangan	Nama Kelas Analisis Terkait
1	Akun	Akun
2	Pengguna	Pengguna
3	Pelanggan	Pelanggan
4	Admin	Admin
5	Event	Event
6	Tiket	Tiket
7	Pembayaran	Pembayaran

1. Kelas Akun

Nama Kelas : Akun

<i>Nama Operasi</i>	<i>Visibility (private, public)</i>	<i>Keterangan</i>
<i>Diisi dengan signature operasi</i>		
<i>Nama Atribut</i>	<i>Visibility (private, public)</i>	<i>Tipe</i>
nama	private	String
alamat	private	String
tempatTanggalLahir	private	String
password	private	String

2. Kelas Pengguna

Nama Kelas : Pengguna

<i>Nama Operasi</i>	<i>Visibility (private, public)</i>	<i>Keterangan</i>
Pengguna()	public	
getId()	public	Method untuk memanggil Id Pengguna
getNama()	public	Method untuk memanggil Nama Pengguna
getAlamat()	public	Method untuk memanggil Alamat Pengguna
getTTL()	public	Method untuk memanggil Tempat Tanggal Lahir Pengguna
getPass()	public	Method untuk memanggil Password Pengguna

deletePengguna()	public	Method untuk menghapus pengguna
updatePengguna()	public	Method untuk meng- <i>update</i> pengguna
<i>Nama Atribut</i>	<i>Visibility (private, public)</i>	<i>Tipe</i>
idUser	private	<i>String</i>

3. Kelas Pelanggan

Nama Kelas : Pelanggan

<i>Nama Operasi</i>	<i>Visibility (private, public)</i>	<i>Keterangan</i>
Pelanggan()	public	
getId()	public	Method untuk memanggil Id Pelanggan
getNama()	public	Method untuk memanggil Nama Pelanggan
getAlamat()	public	Method untuk memanggil Alamat Pelanggan
getTTL()	public	Method untuk memanggil Tempat Tanggal Lahir Pelanggan
getPass()	public	Method untuk memanggil Password Pelanggan
deletePelanggan()	public	Method untuk menghapus pelanggan

updatePelanggan()	public	Method untuk meng- <i>update</i> pelanggan
<i>Nama Atribut</i>	<i>Visibility (private, public)</i>	<i>Tipe</i>
idPelanggan	private	<i>String</i>

4. Kelas Admin

Nama Kelas : Admin

<i>Nama Operasi</i>	<i>Visibility (private, public)</i>	<i>Keterangan</i>
Admin()	public	
getId()	public	Method untuk memanggil Id Admin
getNama()	public	Method untuk memanggil Nama Admin
getAlamat()	public	Method untuk memanggil Alamat Admin
getTTL()	public	Method untuk memanggil Tempat Tanggal Lahir Admin
getPass()	public	Method untuk memanggil Password Admin
deleteAdmin()	public	Method untuk menghapus Admin
updateAdmin()	public	Method untuk meng- <i>update</i> Admin

<i>Nama Atribut</i>	<i>Visibility (private, public)</i>	<i>Tipe</i>
idAdmin	private	String

5. Kelas Event

Nama Kelas : Event

<i>Nama Operasi</i>	<i>Visibility (private, public)</i>	<i>Keterangan</i>
Event()	public	
deleteEvent()	public	Method untuk menghapus Event
updateEvent()	public	Method untuk meng- <i>update</i> Event
getId()	public	Method untuk memanggil Id Event
getJudul()	public	Method untuk memanggil Judul Event
getJadwal()	public	Method untuk memanggil Jadwal Event
getTempat()	public	Method untuk memanggil tempat Event
getDeskripsi()	public	Method untuk memanggil deskripsi Event
getKategori()	public	Method untuk memanggil kategori Event

getIdUser()	public	Method untuk memanggil Id User
<i>Nama Atribut</i>	<i>Visibility (private, public)</i>	<i>Tipe</i>
idEvent	Private	<i>String</i>
Judul	Private	<i>String</i>
Jadwal	Private	<i>String</i>
Tempat	Private	<i>String</i>
Deskripsi	Private	<i>String</i>
Kategori	Private	<i>String</i>
IdUser	Private	<i>String</i>

6. Kelas Tiket

Nama Kelas : Tiket

<i>Nama Operasi</i>	<i>Visibility (private, public)</i>	<i>Keterangan</i>
Tiket()	public	
deleteTiket()	public	Method untuk menghapus Tiket
updateTiket()	public	Method untuk meng- <i>update</i> Tiket

getIdTiket()	public	Method untuk memanggil Id Tiket
getIdEvent()	public	Method untuk memanggil Id Event
getIdPelanggan()	public	Method untuk memanggil Id Pelanggan
getHarga()	public	Method untuk memanggil harga
Status()	public	Method untuk memanggil status
<i>Nama Atribut</i>	<i>Visibility (private, public)</i>	<i>Tipe</i>
harga	Private	<i>Integer</i>
IdTiket	Private	<i>String</i>
IdEvent	Private	<i>String</i>
IdPelanggan	Private	<i>String</i>
Status	Private	<i>String</i>

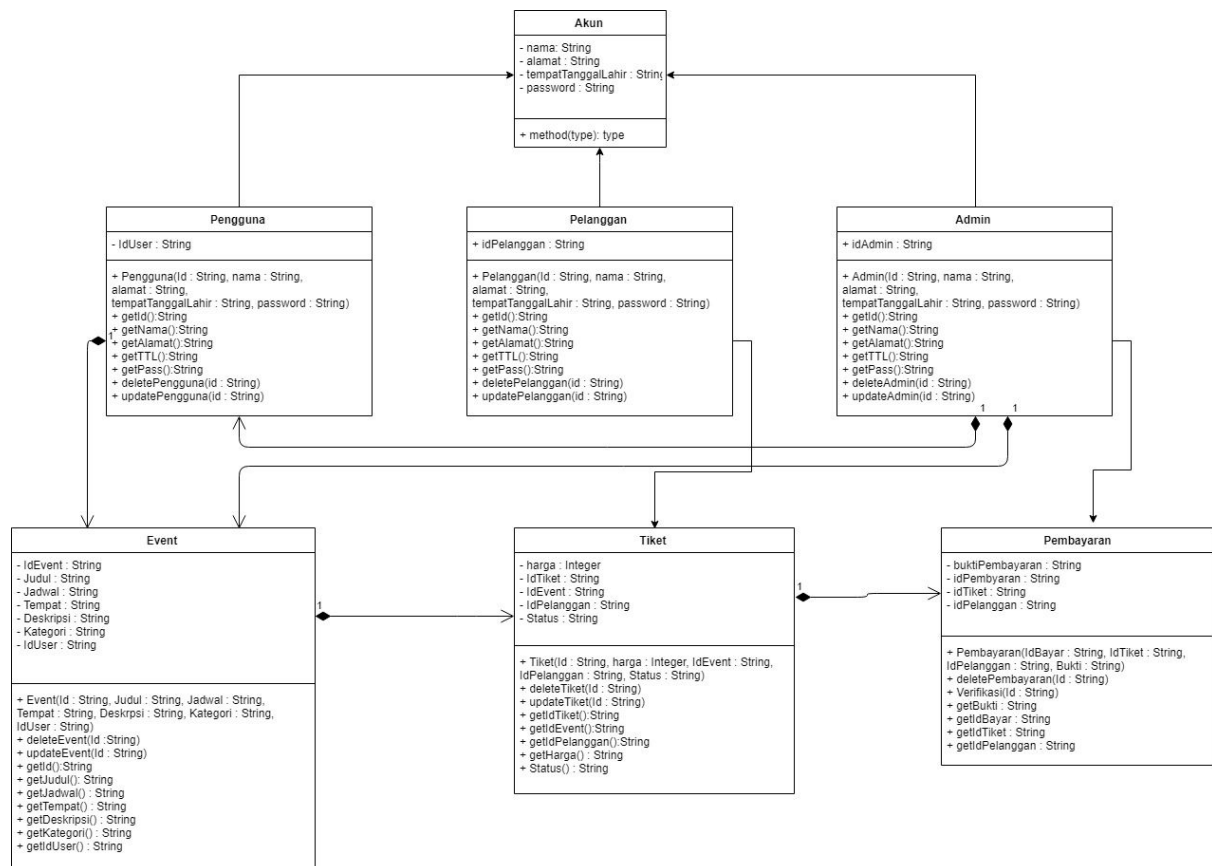
7. Kelas Pembayaran

Nama Kelas : Pembayaran

<i>Nama Operasi</i>	<i>Visibility (private, public)</i>	<i>Keterangan</i>
---------------------	---	-------------------

Pembayaran()	public	
deletePembayaran()	public	Method untuk menghapus Tiket
Verifikasi()	public	Method untuk meng- <i>update</i> Tiket
getIdBukti()	public	Method untuk memanggil Id Tiket
getIdBayar()	public	Method untuk memanggil Id Event
getIdTiket()	public	Method untuk memanggil Id Pelanggan
getIdPelanggan()	public	Method untuk memanggil harga
<i>Nama Atribut</i>	<i>Visibility (private, public)</i>	<i>Tipe</i>
buktiPembayaran	Private	<i>String</i>
IdPembayaran	Private	<i>String</i>
IdTiket	Private	<i>String</i>
IdPelanggan	Private	<i>String</i>

3.6 Diagram Kelas Keseluruhan



3.7 Algoritma/Query

Nama Kelas : Tiket

Nama Operasi : Memesan Tiket

Algoritma : (Algo-001)

1. Saat event dipilih, sistem memproses pesanan bersamaan dengan mengirim ID User yang memesan.
2. Pemesanan akan di *insert* ke *database* dengan status = “belum terverifikasi”.
3. Sistem akan mengirimkan alamat rekening untuk menyelesaikan proses pembayaran.

Query :

No Query	Query	Keterangan
Q-001	Insert into tbTiket (Id_tiket, Id_Event, Id_Customer, Harga, Status) Value (\$Id_tiket,	Memasukan data Tiket ke <i>database</i> dengan status “Belum Terverifikasi”

	\$Id_Event, \$Id_Customer, \$Harga, “Belum Terverifikasi”);	
--	---	--

Nama Kelas : Tiket

Nama Operasi : Verifikasi Tiket

Algoritma : (*Algo-002*)

1. Admin akan memilih data Tiket yang memiliki status “Belum Terverifikasi”.
2. Saat Admin menganggap data bisa diverifikasi pembayarannya, Admin akan mengklik tombol “Verifikasi”.
3. Sistem akan mengubah status pada database menjadi “Terverifikasi”

Query :

No Query	Query	Keterangan
Q-002	Update tbTiket set status = “Terverifikasi” where Id_Tiket = \$Id_tiket;	Mengubah status pada tbTiket menjadi “Terverifikasi”

Nama Kelas : Event

Nama Operasi : Membuat Event

Algoritma : (*Algo-003*)

1. User memasukan data Event yang ingin dibuat dengan lengkap dan benar.
2. Saat User mengklik tombol Submit, sistem akan meng-*insert* data tersebut ke database dengan status = “Belum Terverifikasi”

Query :

No Query	Query	Keterangan
Q-003	Insert into tbEvent (Id_Event, Judul, Jadwal, Tempat, Deskripsi, Kategori, Id_User, Harga) Value (\$Id_Event, \$Judul, \$Jadwal, \$Tempat, \$Deskripsi, \$Kategori, \$Id_User, \$Harga);	Memasukan data Event ke <i>database</i> dengan status “Belum Terverifikasi”

Nama Kelas : Event

Nama Operasi : Verifikasi Event

Algoritma : (*Algo-004*)

1. Admin akan memilih data Event yang memiliki status “Belum Terverifikasi”.
2. Saat Admin menganggap data bisa diverifikasi, Admin akan mengklik tombol “Verifikasi”.

3. Sistem akan mengubah status pada database menjadi “Terverifikasi”

Query :

No Query	Query	Keterangan
Q-004	Update tbEvent set status = “Terverifikasi” where Id_Event = \$Id_event;	Mengubah status pada tbEvent menjadi “Terverifikasi”

Nama Kelas : Event

Nama Operasi : List Event

Algoritma : (Algo-005)

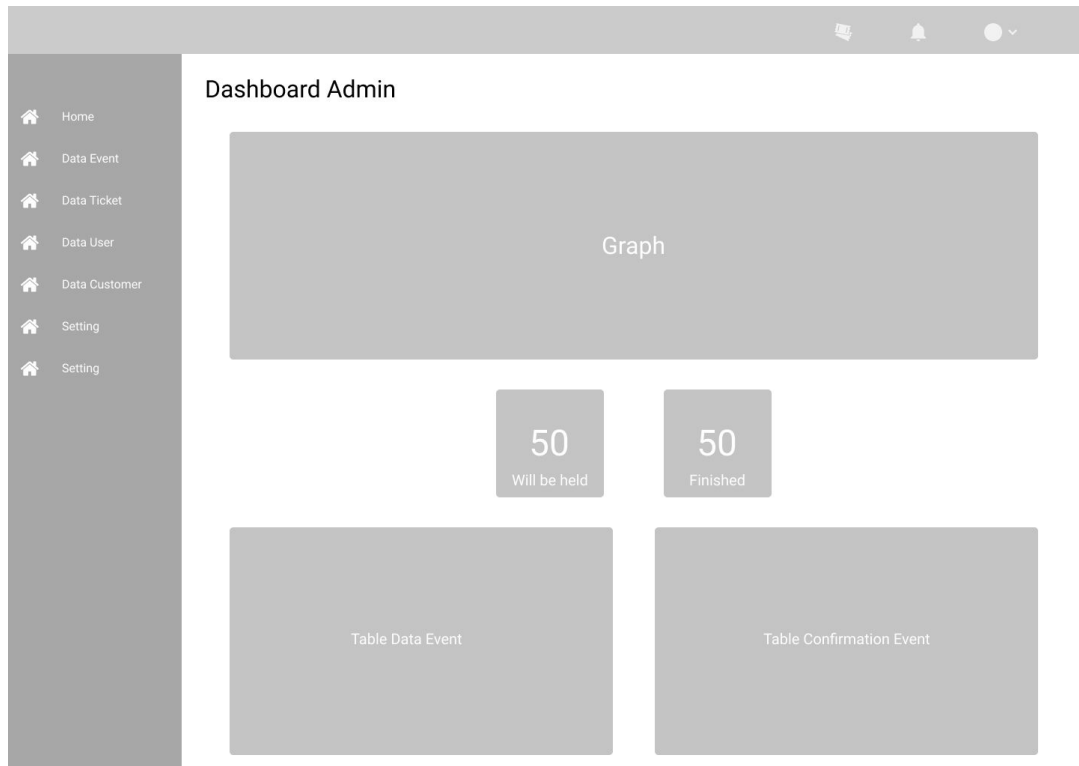
1. Saat User atau Customer memasuki halaman list event, event yang ditampilkan hanyalah event yang sudah terverifikasi saja.

Query :

No Query	Query	Keterangan
Q-005	Select * from tbEvent where Status = “Terverifikasi”;	Memunculkan data event yang telah terverifikasi oleh admin.

3.8 Perancangan Antarmuka

Dashboard Admin :



Id_Objek	Jenis	Nama	Keterangan
Nav1	Link	Home Link	Link yang membawa ke halaman home
NAV2	Link	Data Event	Link yang membawa ke halaman kumpulan data event
NAV3	Link	Data Ticket	Link yang membawa ke halaman kumpulan ticket yang dipesan oleh Customer
NAV4	Link	Data User	Link yang membawa ke halaman kumpulan data user yang telah mendaftar
NAV5	Link	Data Customer	Link yang membawa ke halaman kumpulan data customer yang telah mendaftar
NAV6	Link	Setting	Link yang membawa ke halaman setting
HEAD1	Link	Profile	Link yang membawa ke halaman profile
HEAD2	Dropdown	Notification	Memberitahu tentang hal yang terjadi saat ini
HEAD3	Link	Ticket	Link yang membawa ke halaman pemesanan tiket
GRAPH	Graph	Graph	Grafik yang menunjukkan perbandingan antara event yang akan diadakan dan yang sudah diadakan dari hari ke hari.
CARD1	Card	Will be held	Menunjukkan jumlah terakhir event yang akan diadakan.
CARD2	Card	Finished	Menunjukkan jumlah terakhir event yang sudah diadakan
TABLE1	Table	Table data event	Berisi table yang menunjukkan data event secara singkat

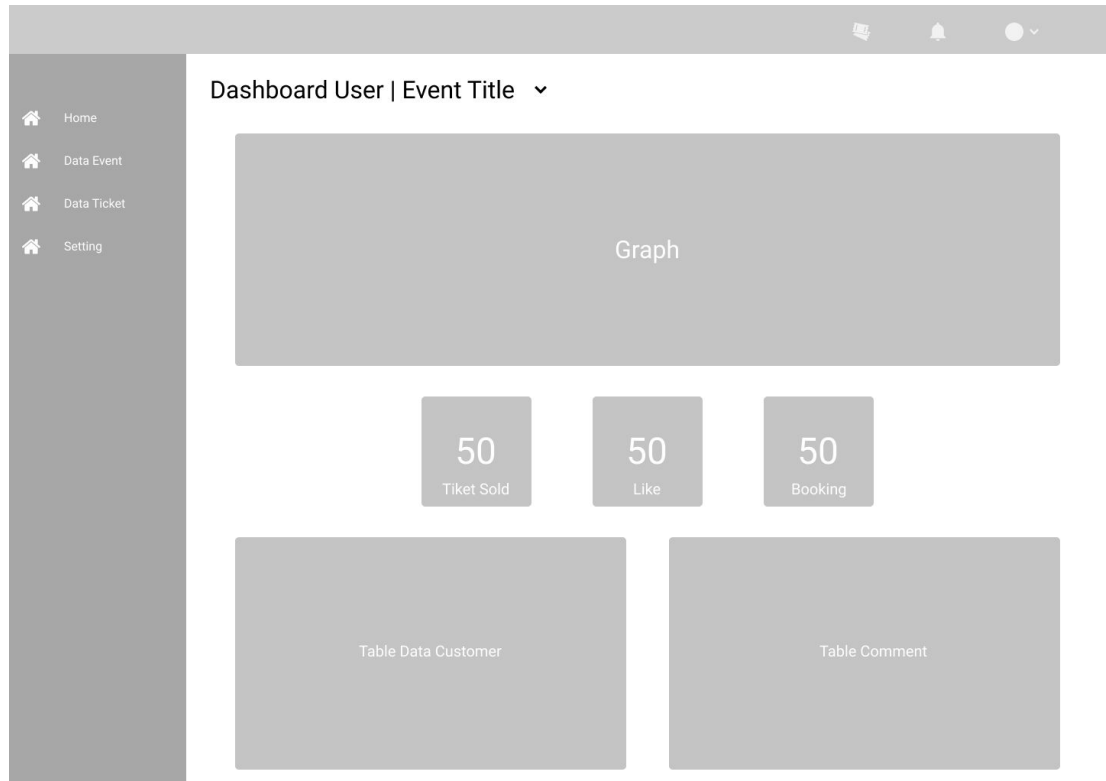
TABLE2	Table	Table Confirmation Event	Berisi table yang menunjukan data event yang belum di konfirmasi
--------	-------	--------------------------	--

Dashboard Customer :

The image shows a dashboard for a customer. At the top, there is a header bar with a search icon, a notification bell, and a user profile icon. Below the header, there is a search section titled "Search Event". It contains a dropdown menu labeled "Category", a text input field labeled "Value", and a "Search" button. Below the search section, there is a "Result" section displaying a grid of event cards. Each card features a placeholder image, a "Title of Event", a date "21 April 2019", a location pin icon, and a short description: "Short description goes here. Lorem ipsum dolor sit amet and bla bla bla".

Id_Objek	Jenis	Nama	Keterangan
HEAD1	Link	Profile	Link yang membawa ke halaman profile
HEAD2	Dropdown	Notification	Memberitahu tentang hal yang terjadi saat ini
HEAD3	Link	Ticket	Link yang membawa ke halaman pemesanan tiket
SEARCH1	Dropdown	Category	Untuk memilih kategori untuk kepentingan pencarian data
SEARCH2	Text box	Value	Kolom pencarian untuk mencari data berdasarkan kategori yang telah dipilih
SEARCH3	Button	Search	Tombol yang digunakan untuk memulai pencarian
CARD	Card	Event	Hasil pencarian data, yang berupa daftar event dalam bentuk card.

Dashboard User :



Id_Objek	Jenis	Nama	Keterangan
Nav1	Link	Home Link	Link yang membawa ke halaman home
NAV2	Link	Data Event	Link yang membawa ke halaman kumpulan data event
NAV3	Link	Data Ticket	Link yang membawa ke halaman kumpulan ticket yang dipesan oleh Customer
NAV4	Link	Setting	Link yang membawa ke halaman setting
HEAD1	Link	Profile	Link yang membawa ke halaman profile
HEAD2	Dropdown	Notification	Memberitahu tentang hal yang terjadi saat ini
HEAD3	Link	Ticket	Link yang membawa ke halaman pemesanan tiket
GRAPH	Graph	Graph	Grafik yang menunjukkan perbandingan pemesanan tiket dari hari ke hari
CARD1	Card	Ticket Sold	Menunjukkan jumlah terakhir ticket yang telah konfirmasi
CARD2	Card	Like	Menunjukkan banyaknya customer yang menyukai acara ini
CARD3	Card	Booking	Menunjukkan jumlah terakhir ticket yang telah dipesan namun belum dikonfirmasi

TABLE1	Table	Table data customer	Berisi tabel yang menunjukkan data customer yang memesan tiket
TABLE2	Table	Table Comments	Berisi table yang menunjukkan komentar terhadap event ini.
DROP1	Dropdown	Event Title	Untuk memilih event yang ingin dikelola

4. Matriks Kerunutan

Requirement	Usecase Terkait	Kelas
FR-01	Pembayaran, buatEvent, hapusEvent, Registrasi, login, kelolaAkun, verifikasiEvent, lihatEvent	Admin
FR-02	buatEvent, kelolaAkun, registrasi, login, kelolaAkun	User
FR-03	pemesanan, lihatEvent, kelolaAkun, registrasi, login	Customer
FR-04	buatEvent, hapusEvent, verifikasiEvent	Event
FR-05	pembayaran, pemesanan	Ticket
FR-06	pembayaran	Payment