

## BAB IV

### RANCANGAN SISTEM DAN PROGRAM USULAN

#### 4.1. Analisa Kebutuhan *Software*

Analisa kebutuhan *software* merupakan tahap untuk mengidentifikasi seluruh kebutuhan perangkat lunak yang diperlukan agar sistem informasi *e-Procurement* dapat berjalan secara optimal. Kebutuhan ini meliputi spesifikasi fungsional dan non-fungsional, batasan sistem, serta dependensi yang harus dipenuhi dalam proses pengembangan.

Kegiatan analisa kebutuhan *software* bertujuan untuk memastikan bahwa aplikasi yang dibangun mampu mendukung proses bisnis pengadaan barang/jasa, mulai dari usulan, persetujuan dan pengumuman, pendaftaran vendor, penawaran, hingga evaluasi dan penetapan pemenang secara efektif, terintegrasi, dan dapat digunakan oleh seluruh pihak yang terlibat.

##### 1. Kebutuhan Fungsional Administrator

- a. Administrator dapat mengelola data master seperti data barang/jasa, data vendor, data divisi, dan pengguna.
- b. Administrator dapat mengelola data pendukung seperti data KBLI (Klasifikasi Baku Lapangan Usaha Indonesia), data tahapan, dan data satuan.
- c. Administrator dapat mengelola data tender usulan.
- d. Administrator dapat membuka data penawaran dan melakukan evaluasi.
- e. Administrator dapat membuat laporan data master dan pengadaan.

##### 2. Kebutuhan Fungsional Admin Divisi

- a. Admin Divisi dapat menambah dan mengubah data barang/jasa,
- b. Admin Divisi dapat mengelola data tender usulan

##### 3. Kebutuhan Fungsional Pejabat Penandatanganan Kontrak (PPK)

- a. PPK dapat menyetujui atau menolak usulan tender
  - b. PPK dapat membuat laporan data master dan pengadaan
4. Kebutuhan Fungsional Vendor
- c. Vendor dapat melakukan registrasi
  - d. Vendor dapat melakukan perubahan data vendor
  - e. Vendor dapat mengikuti tender
  - f. Vendor dapat mengirim/memasukkan penawaran

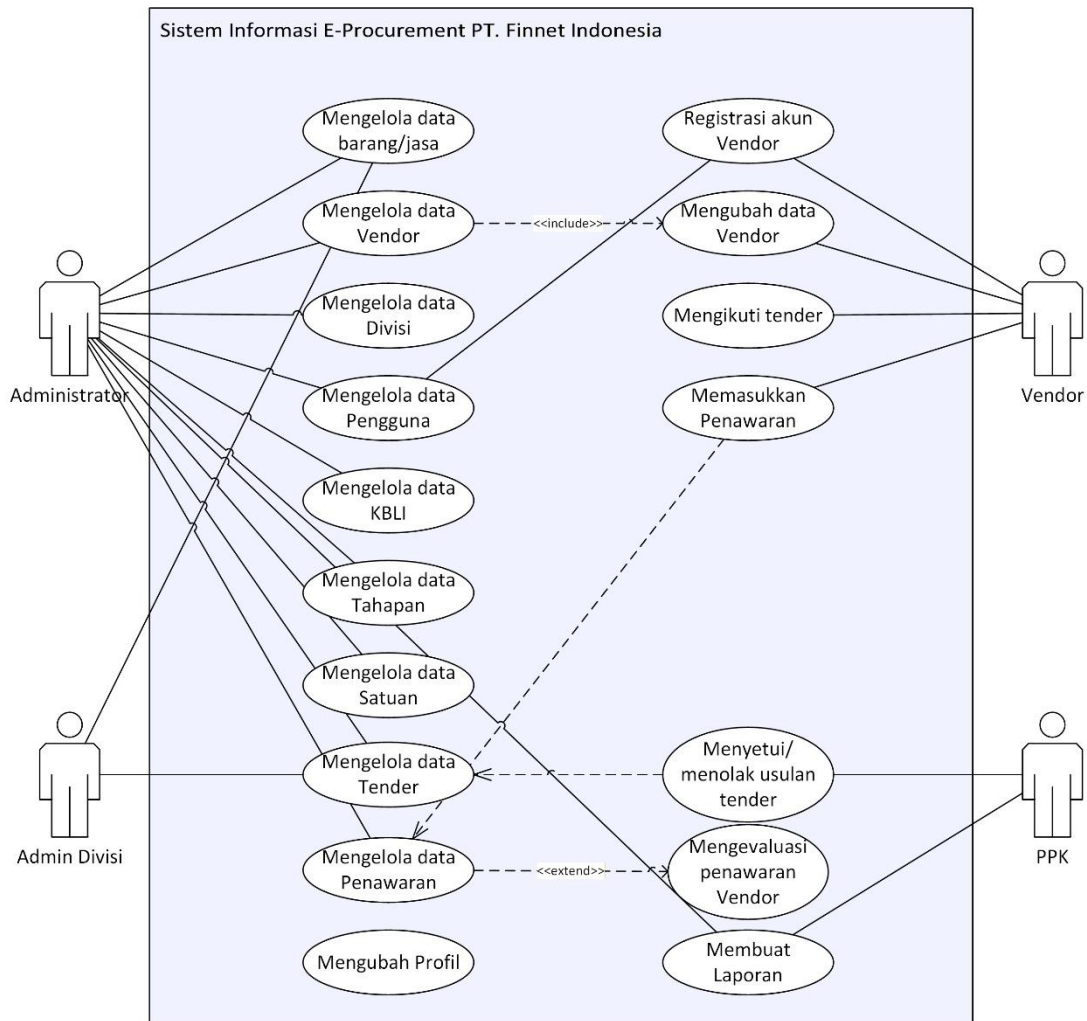
Dengan dilakukannya analisa kebutuhan software ini, seluruh aspek fungsional dalam pengembangan sistem informasi e-Procurement dapat diidentifikasi secara jelas dan terstruktur. Analisa ini memastikan bahwa sistem yang dibangun tidak hanya mampu mendukung proses bisnis pengadaan secara menyeluruh, tetapi juga memenuhi standar kualitas kinerja, keandalan, dan kemudahan penggunaan. Hasil analisa ini selanjutnya menjadi dasar dalam proses perancangan sistem dan implementasi, sehingga aplikasi yang dihasilkan mampu beroperasi secara optimal serta memberikan manfaat maksimal bagi seluruh pemangku kepentingan.

## **4.2. Desain**

### **4.2.1. Desain Pemodelan Sistem**

#### **A. Pemodelan *Use Case Diagram***

Perancangan sistem informasi *E-Procurement* ini dilaksanakan dengan pembuatan desain model *use case diagram*. Berikut ini adalah rancangan atau pemodelan *use case diagram* proses bisnis utama sistem informasi *E-Procurement* pada PT. Finnet Indonesia.



Sumber: Hasil Penelitian, 2025

Gambar IV.1. Pemodelan *Use Case Diagram*

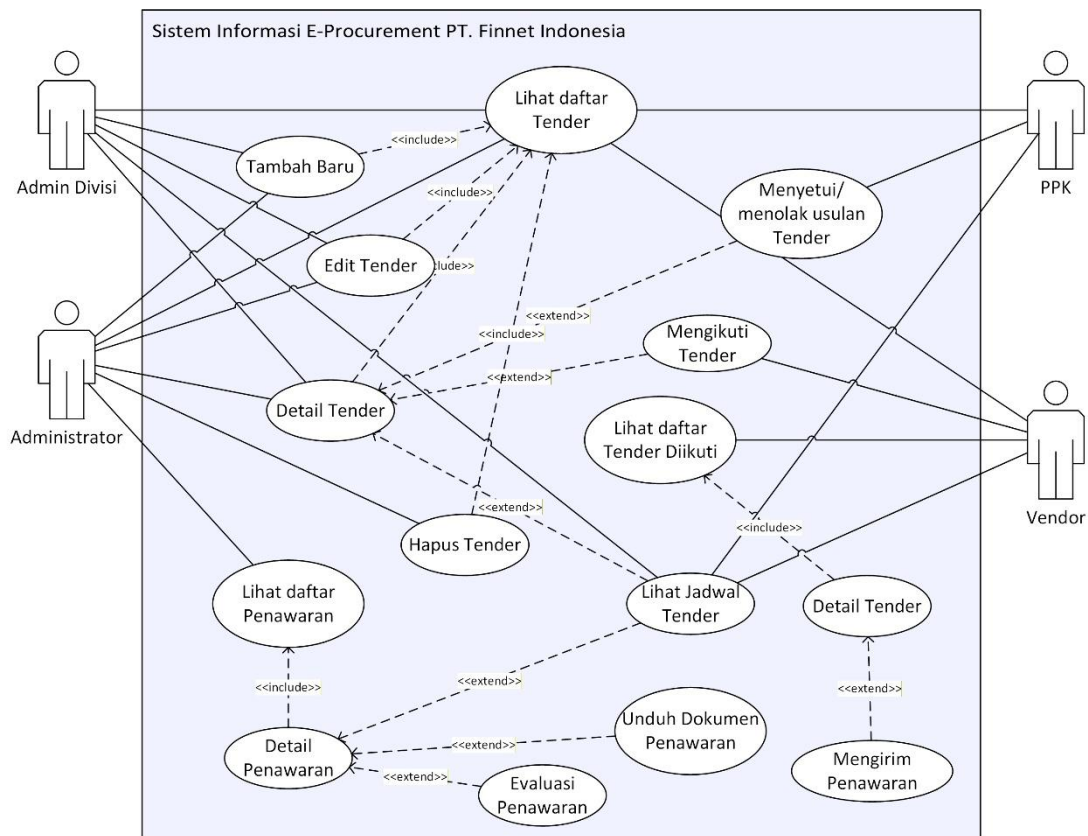
Gambar IV.1 menggambarkan *use case diagram* rancangan Sistem Informasi *E-Procurement* PT. Finnet Indonesia. Pada gambar terdapat 4 (empat) aktor pengguna sistem yaitu Administrator, Admin Divisi, Pejabat Penandatanganan Kontrak (PPK), dan Vendor. Kebutuhan fungsional setiap aktor sebagaimana dijelaskan pada analisa kebutuhan *software* telah dipenuhi pada pemodelan *use case diagram*.

Tabel IV.1. Deskripsi *Use Case Diagram*

<i>Use Case Name</i>	:	Sistem Informasi <i>E-Procurement</i> PT. Finnet Indonesia
<i>Use Case Description</i>	:	Semua fitur-fitur yang tersedia pada sistem sesuai dengan kebutuhan fungsional masing-masing aktor
<i>Actors</i>	:	Administrator, Admin Divisi, PPK, dan Vendor
<i>Pre-Condition</i>	:	1. Diakses melalui <i>web browser</i> dengan koneksi internet

		2. Setiap aktor harus memiliki akun masing-masing
<i>Post-Condition</i>	:	Beberapa fitur dapat menghasilkan keluaran seperti dokumen pengumuman dan dokumen laporan
<i>Fault-Condition</i>	:	Kesalahan atau kekurangan isian data, duplikasi data, pemrosesan data tidak sesuai ketentuan.
<i>Main Scenarios</i>	Serial No.	Step
Administrator	1	Mengelola data barang/jasa
	2	Mengelola data divisi
	3	Mengelola data vendor
	4	Mengelola data pengguna
	5	Mengelola data KBLI
	6	Mengelola data tahapan
	7	Mengelola data satuan
	8	Mengelola data tender
	9	Mengelola data penawaran
	10	Melakukan evaluasi penawaran
	11	Membuat laporan
	12	Mengubah profil
Admin Divisi	1	Mengelola data barang/jasa
	2	Mengelola data tender
	3	Mengubah profil
PPK	1	Menyetujui/menolak usulan tender
	2	Membuat laporan
	3	Mengubah profil
Vendor	1	Melakukan registrasi
	2	Mengubah data Vendor
	3	Mengikuti tender
	4	Memasukkan penawaran

Sumber: Hasil Penelitian, 2025



Sumber: Hasil Penelitian, 2025

Gambar IV.2. Pemodelan *Use Case Diagram* Pengadaan

Gambar IV.2. merupakan model *use case diagram* proses pengadaan pada Sistem Informasi E-Procurement PT. Finnet Indonesia. Proses ini adalah pokok utama dari sistem informasi pengadaan yang dibangun. Keempat aktor terlibat pada proses pengadaan dengan peran masing-masing yang telah disesuaikan pada sistem. Secara singkat dapat dijelaskan bahwa Admin Divisi membuat usulan tender, PPK menyetujui atau menolak tender yang diusulkan, Vendor mengikuti tender yang telah disetujui dan memasukkan penawaran, dan Administrator melakukan evaluasi penawaran.

Tabel IV.2. Deskripsi *Use Case Diagram* Transaksi

<i>Use Case Name</i>	:	Modul Pengadaan
<i>Use Case Description</i>	:	Proses utama sebagai inti/pokok sistem informasi <i>E-Procurement</i> PT. Finnet Indonesia. Administrator dan Admin Divisi dapat mengelola data tender usulan tetapi hanya Administrator yang dapat menghapus data.

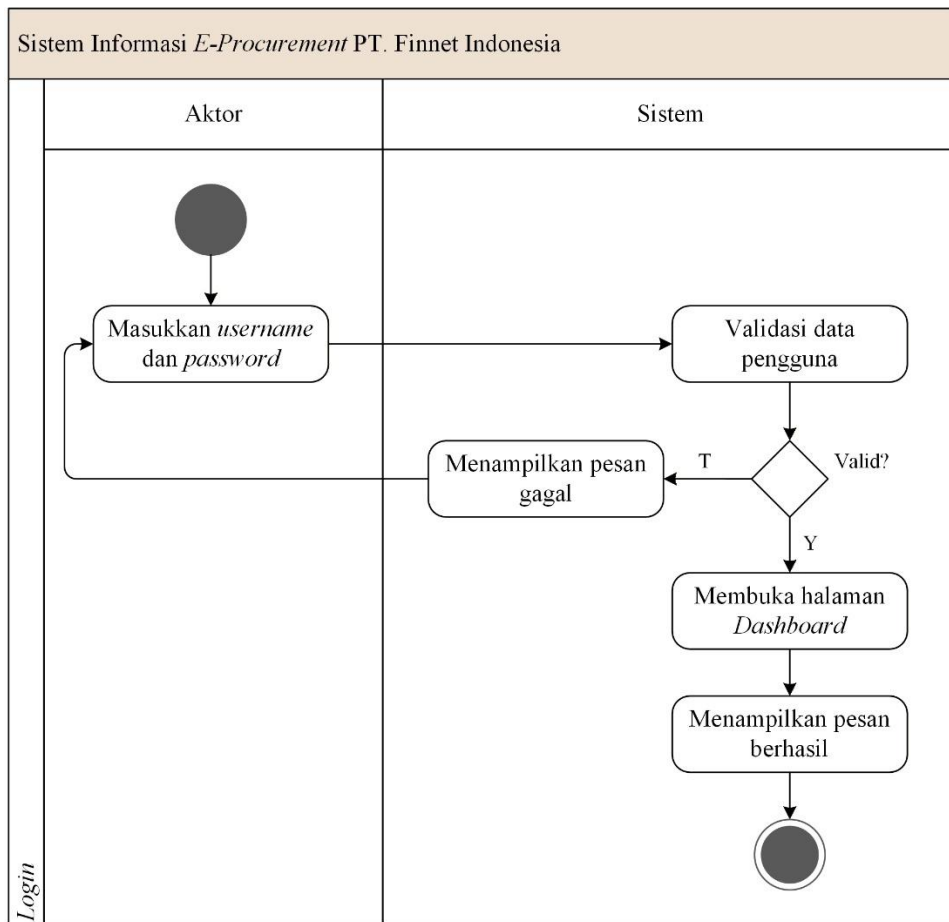
<i>Actors</i>	:	Administrator, Admin Divisi, PPK, dan Vendor
<i>Pre-Condition</i>	:	1. Diakses melalui <i>web browser</i> dengan koneksi internet 2. Aktor harus <i>login</i> terlebih dahulu
<i>Post-Condition</i>	:	<i>Auto generate</i> Jadwal dan Pengumuman Tender
<i>Fault-Condition</i>	:	Data tidak lengkap, aksi proses tidak sesuai jadwal atau kondisi tertentu
<i>Main Scenarios</i>	Serial No.	Step
Admin Divisi	1	Dari daftar tender dapat menambah atau mengubah usulan tender
PPK	1	Dari daftar tender dapat membuka detail tender
	2	Dari detail tender dapat menyetujui/menolak tender
Vendor	1	Dari daftar tender dibuka dapat membuka detail tender
	2	Dari detail tender dapat mengikuti tender
	3	Dari daftar tender diikuti dapat membuka detail tender
	4	Dari detail tender dapat mengirim penawaran
Administrator	1	Dari daftar tender dapat menambah, mengubah, atau menghapus usulan tender
	2	Dari daftar penawaran dapat membuka detail penawaran tender
	3	Dari detail penawaran dapat mengunduh dokumen penawaran
	4	Dari detail penawaran dapat mengevaluasi penawaran

Sumber: Hasil Penelitian, 2025

## B. Pemodelan *Activity Diagram*

Pemodelan *activity diagram* atau diagram aktivitas digunakan untuk menggambarkan alur aktivitas aktor dalam menggunakan dan menjalankan sistem informasi. Diagram ini menunjukkan alur kerja dari setiap fitur dalam proses bisnis mulai dari pemasukan data atau *input*, pengelolaan dan pemrosesan, hingga menghasilkan informasi sebagai *output*.

### 1. *Login*

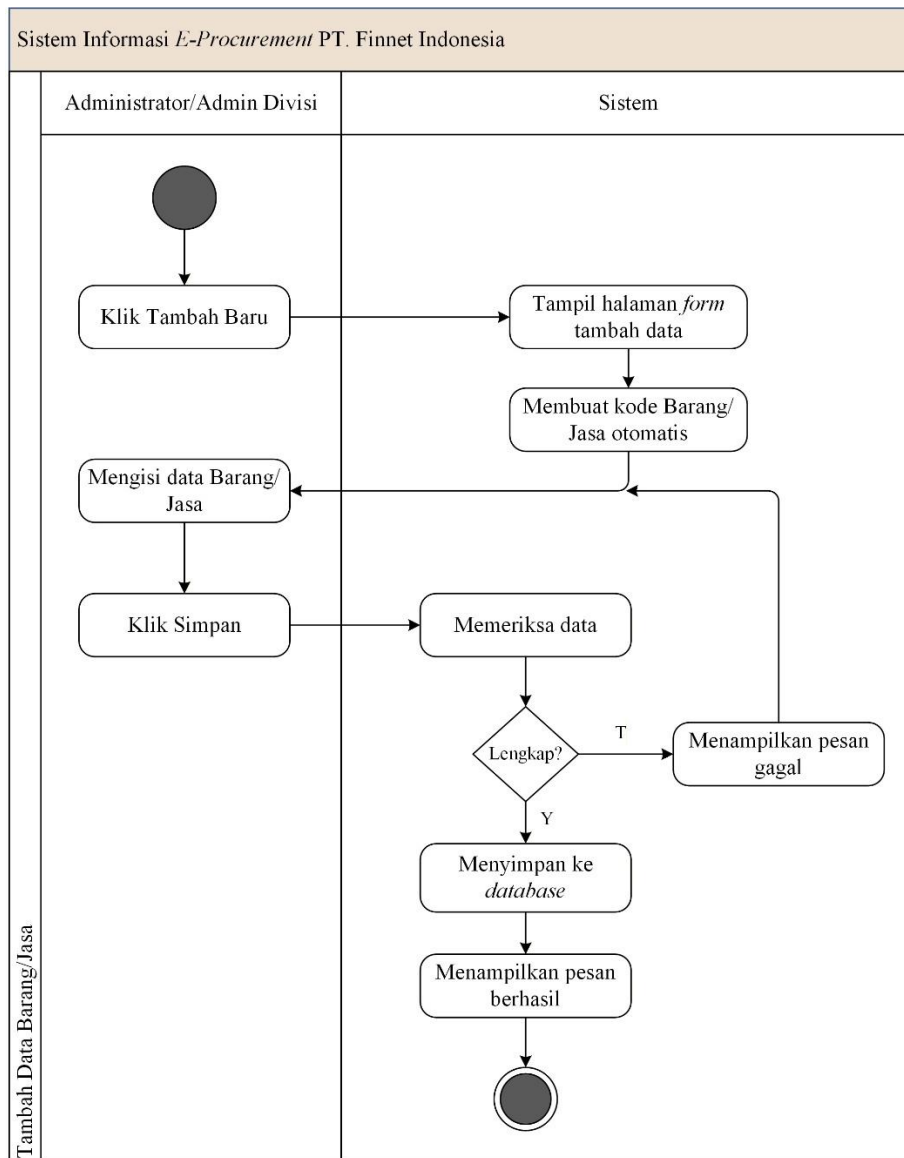


Sumber: Hasil Penelitian, 2025

Gambar IV.3. *Activity Diagram Login*

Gambar IV.3 di atas adalah aktivitas proses masuk ke sistem informasi *e-Procurement* PT. Finnet Indonesia. Pengguna melakukan *input username* dan *password* pada halaman *login*, kemudian sistem melakukan validasi data pengguna. Jika data valid maka pengguna diijinkan masuk ke dalam sistem dan tampil halaman *dashboard*. Jika data tidak valid maka sistem akan menolak dan menampilkan pesan kesalahan *username/password*. Pengguna harus memasukkan *username* dan *password* kembali.

## 2. Menambah data barang/jasa

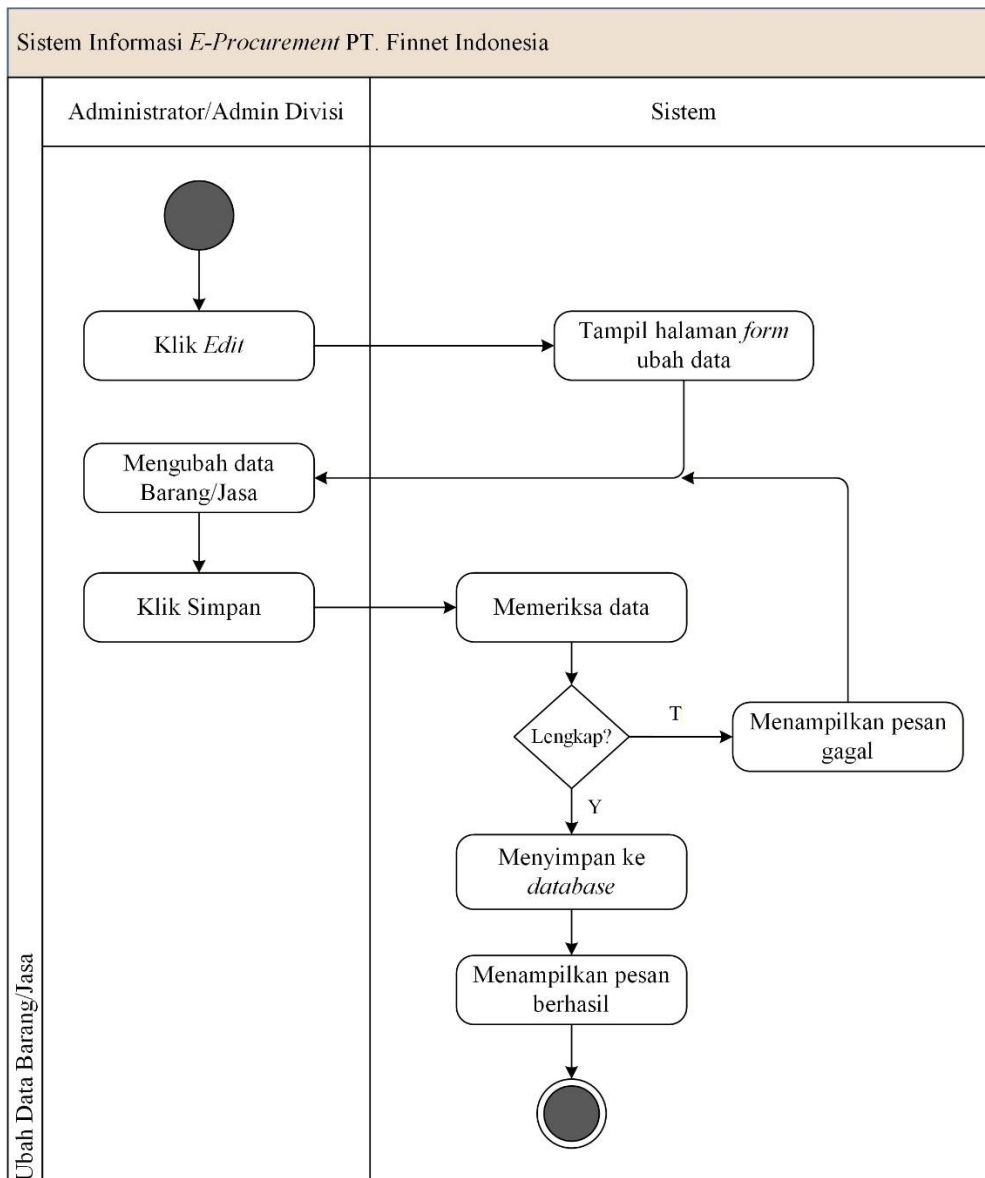


Sumber: Hasil Penelitian, 2025

Gambar IV.4. *Activity Diagram* menambah data barang/jasa

Gambar IV.4 adalah aktivitas proses menambah data barang/jasa ke dalam *database*. Aktor pengguna pada aktivitas ini adalah Administrator dan Admin Divisi. Aktivitas dimulai dari pengguna menekan tombol tambah data dari halaman master barang/jasa. Sistem membuka *form* ambah data barang/jasa lalu pengguna mengisi data. Pengguna menekan tombol simpan dan sistem memvalidasi kelengkapan data. Jika data lengkap maka disimpan dan sistem menampilkan pesan berhasil.

### 3. Mengubah data barang/jasa

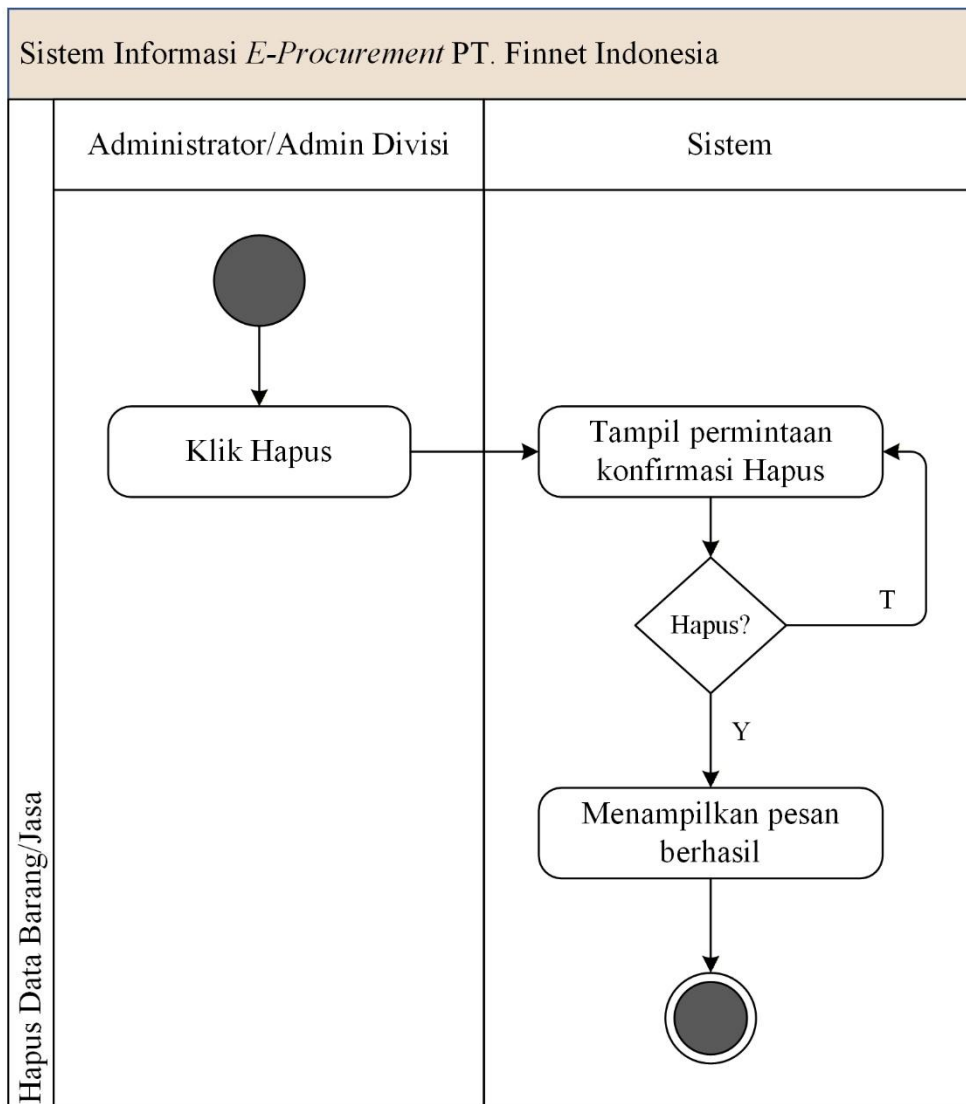


Sumber: Hasil Penelitian, 2025

Gambar IV.5. *Activity Diagram* mengubah data barang/jasa

Gambar IV.5 merupakan diagram aktivitas proses mengubah data pada sistem. Pengguna menekan tombol *Edit* pada salah satu data yang dipilih kemudian sistem akan menampilkan *form* ubah data. Pengguna melakukan perubahan data lalu menekan tombol simpan. Sistem memvalidasi kelengkapan data. Jika data lengkap maka disimpan dan sistem menampilkan pesan berhasil.

#### 4. Menghapus data barang/jasa

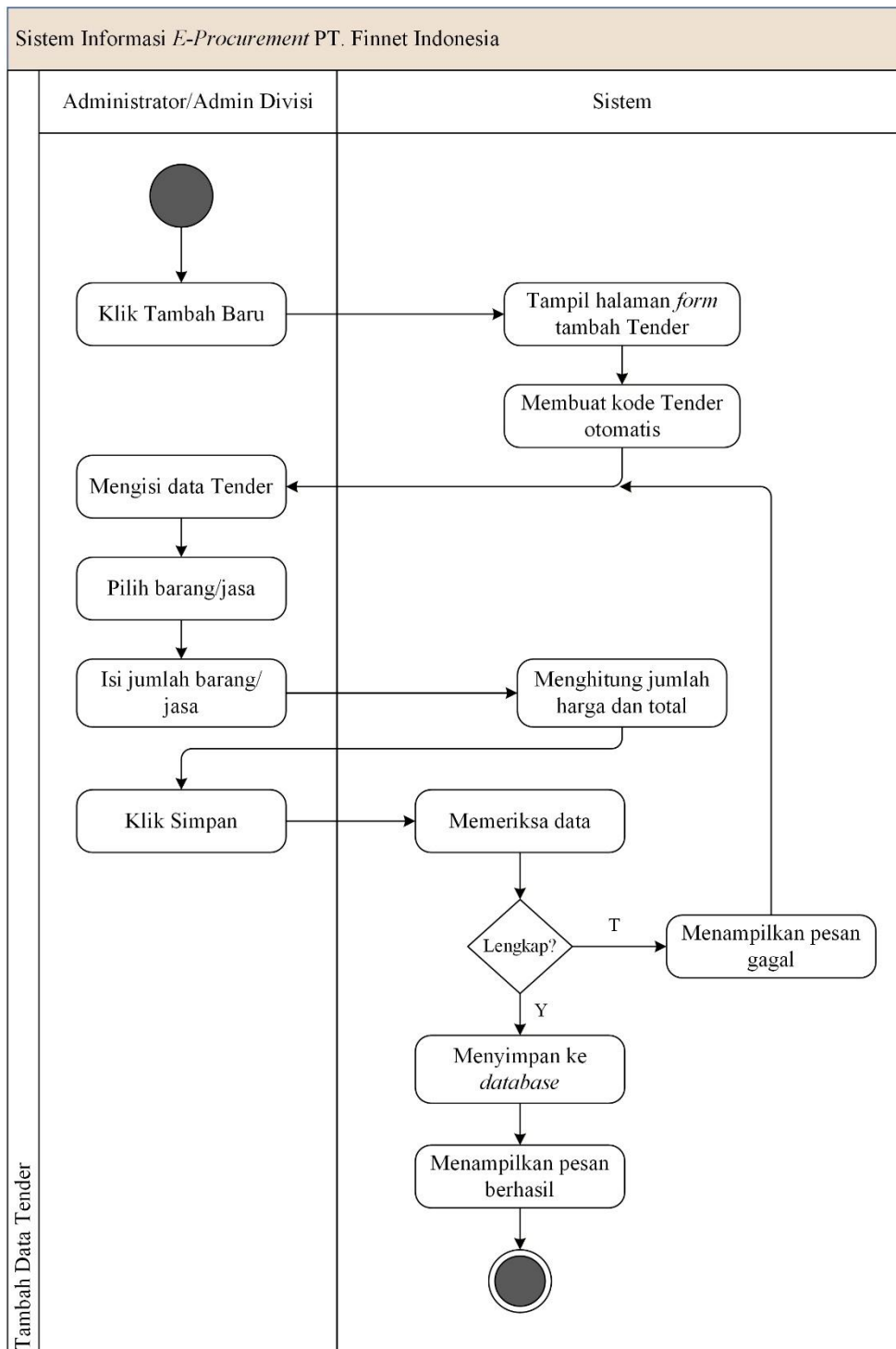


Sumber: Hasil Penelitian, 2025

Gambar IV.6. *Activity Diagram* menghapus data barang/jasa

Gambar IV.6 menampilkan aktivitas proses menghapus data pada sistem. Aktor pengguna pada aktivitas ini hanya Administrator. Pengguna menekan tombol Hapus pada data yang dipilih dan sistem akan meminta konfirmasi untuk melanjutkan proses. Jika pengguna setuju sistem akan menghapus data dan menampilkan pesan berhasil. Konfirmasi diperlukan untuk menghindari penghapusan yang tidak sengaja.

## 5. Menambah data Tender



Sumber: Hasil Penelitian, 2025

Gambar IV.7. Activity Diagram Tambah data Tender

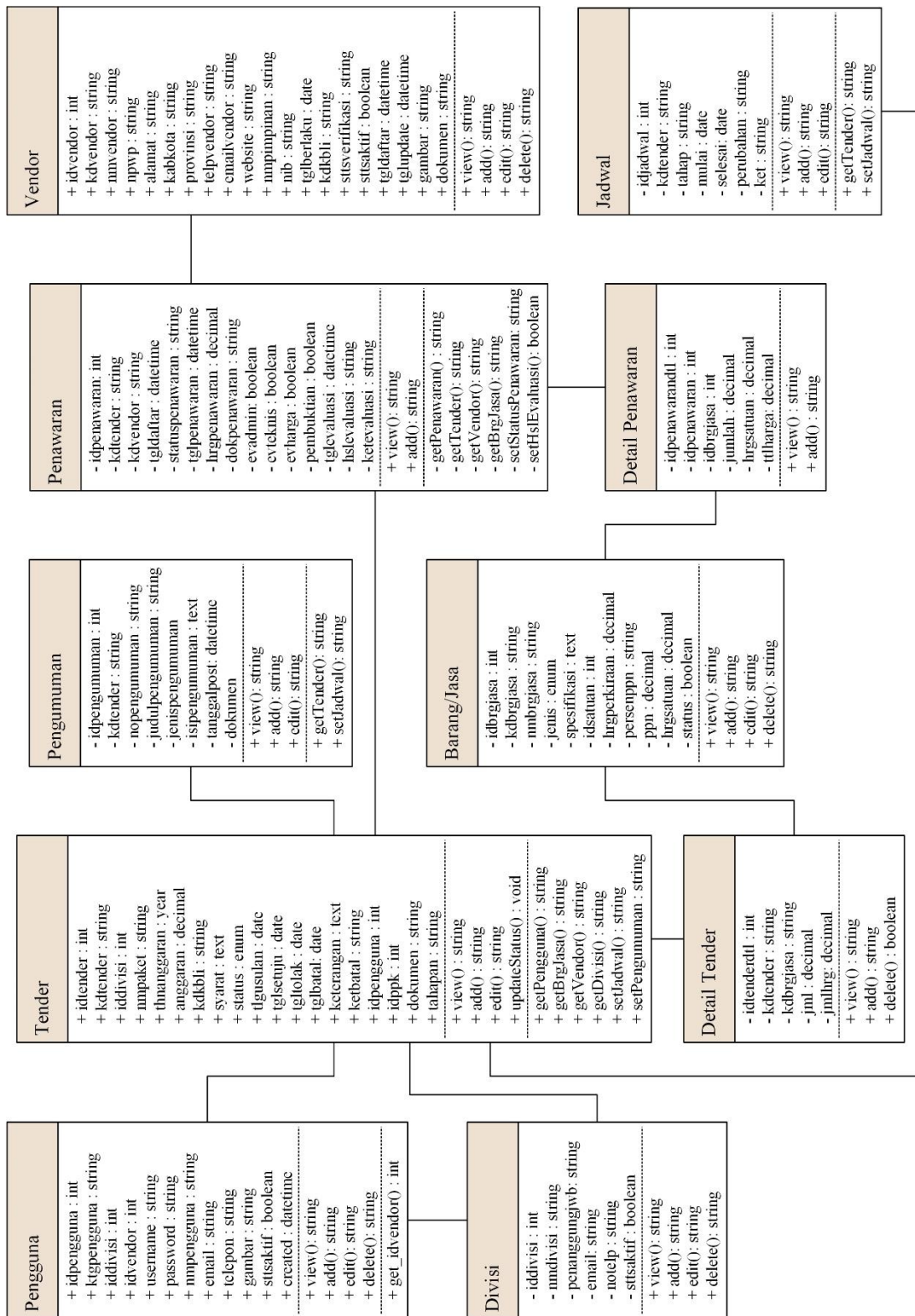
Gambar IV.7 menunjukkan diagram aktivitas proses menambah data tender ke dalam sistem informasi *E-Procurement*. Aktor pengguna pada proses ini adalah Administrator atau Admin Divisi. Pengguna menekan tombol Tambah Data dari halaman indeks tender lalu sistem

akan membuka form tambah tender dan membuat kode tender otomatis berdasarkan tanggal dan urutan tender. Pengguna kemudian mengisi data tender dan data detail barang/jasa yang akan ditenderkan. Sistem akan otomatis menghitung jumlah dan total seluruh barang/jasa yang di-*input* sebagai total anggaran. Kemudian pengguna menyimpan data dan sistem akan memeriksa jika data telah lengkap akan dilanjutkan menyimpan data tender ke *database*.

### **C. Pemodelan *Class Diagram***

Untuk menggambarkan struktur sistem secara lebih terperinci, digunakan pemodelan *class diagram* yang merepresentasikan kelas-kelas yang ada dalam sistem beserta atribut, metode, dan relasi untuk membantu memahami desain sistem dari sisi pemrograman berorientasi objek serta menjadi dasar implementasi kode program.

Class Diagram Sistem Informasi *E-Procurement* PT. Finnet Indonesia menggambarkan struktur statis sistem yang terdiri dari beberapa class utama seperti Pengguna, Vendor, Tender, Penawaran, dan Barang/Jasa. Setiap class merepresentasikan entitas data yang diimplementasikan dalam basis data MySQL dan saling berinteraksi melalui relasi asosiasi sesuai dengan proses bisnis pengadaan barang/jasa. Class Tender berperan sebagai pusat proses pengadaan yang menghubungkan data divisi pengusul, vendor peserta, penawaran, serta jadwal dan pengumuman tender.

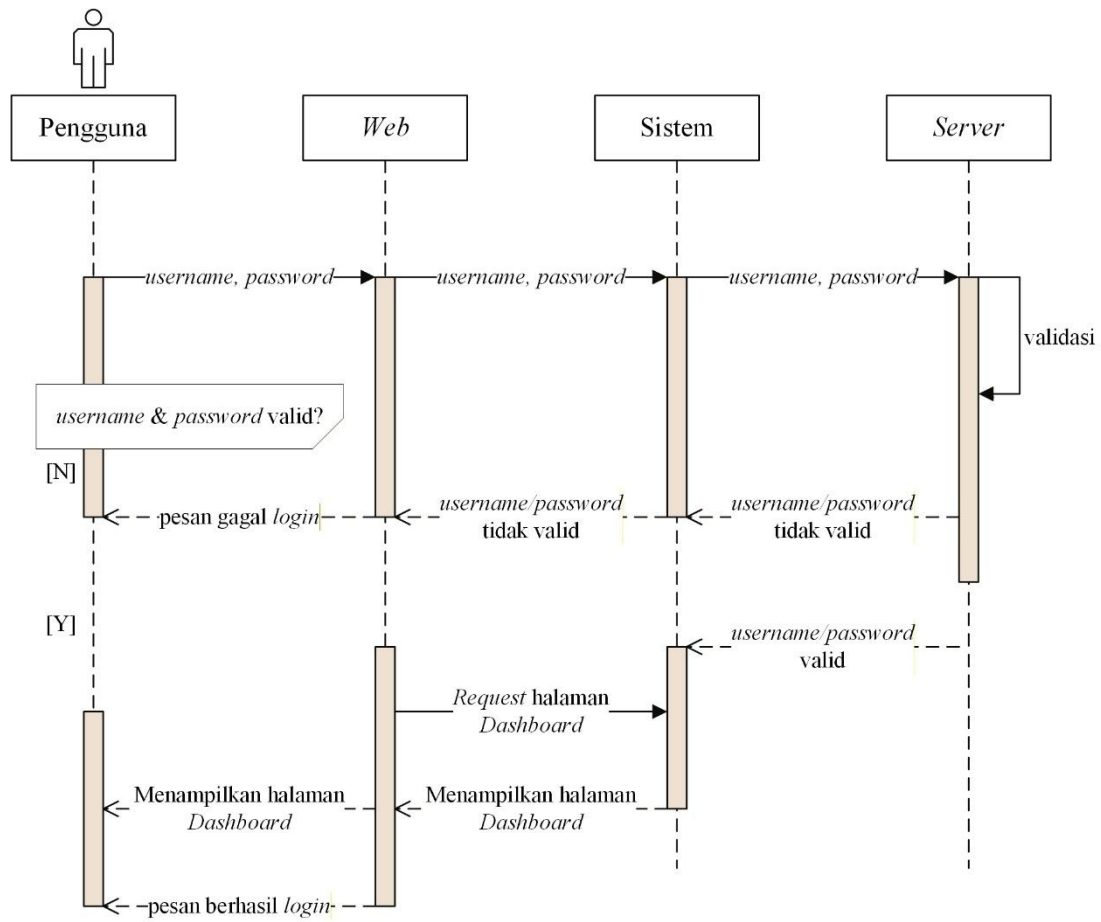


Sumber: Hasil Penelitian, 2025

Gambar IV.8. *Class Diagram* Sistem Informasi E-Procurement PT. Finnet Indonesia

## D. Pemodelan *Sequence Diagram*

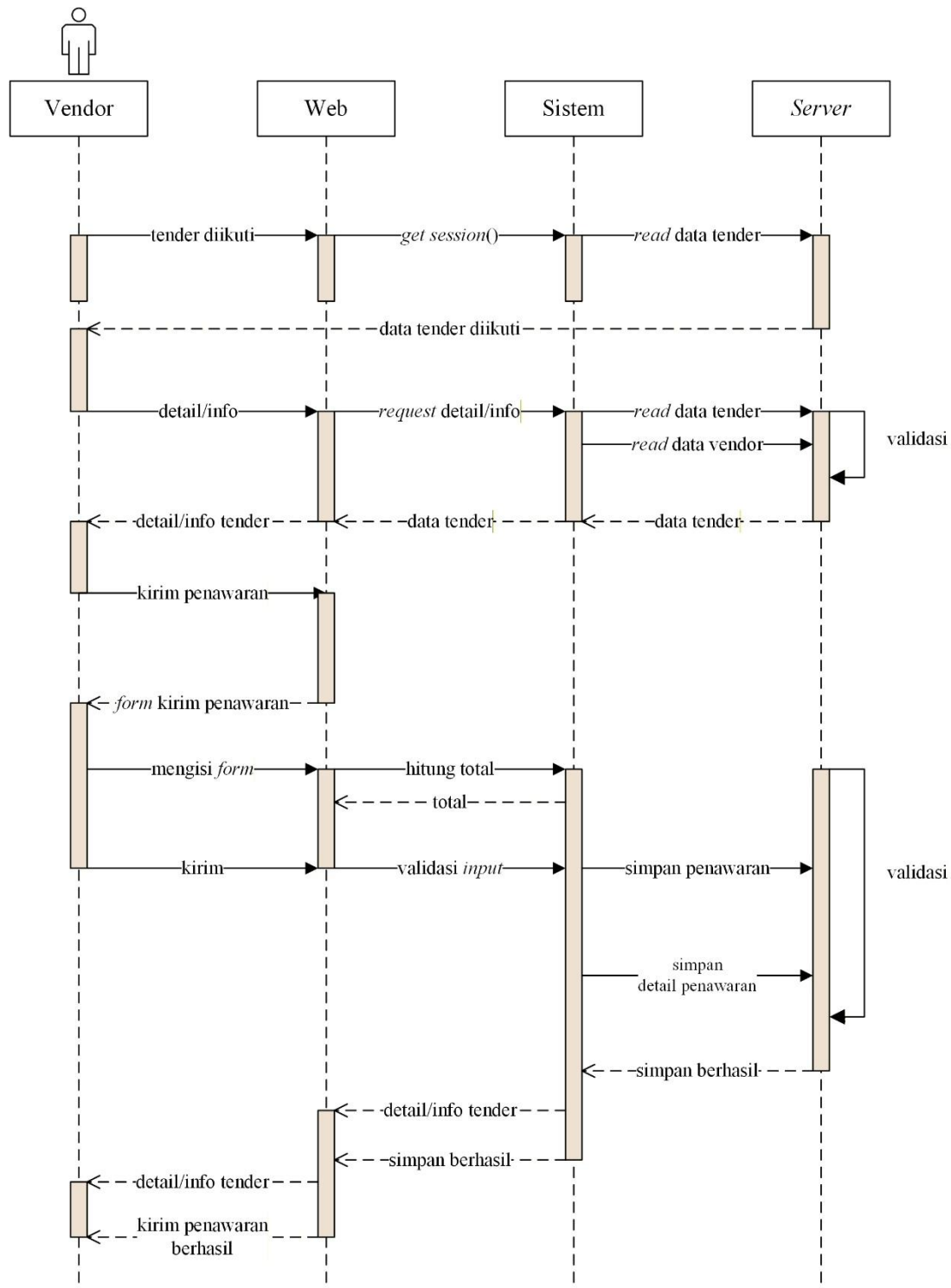
### 1. Login



Sumber: Hasil Penelitian, 2025

Gambar IV.9. *Sequence Diagram Login*

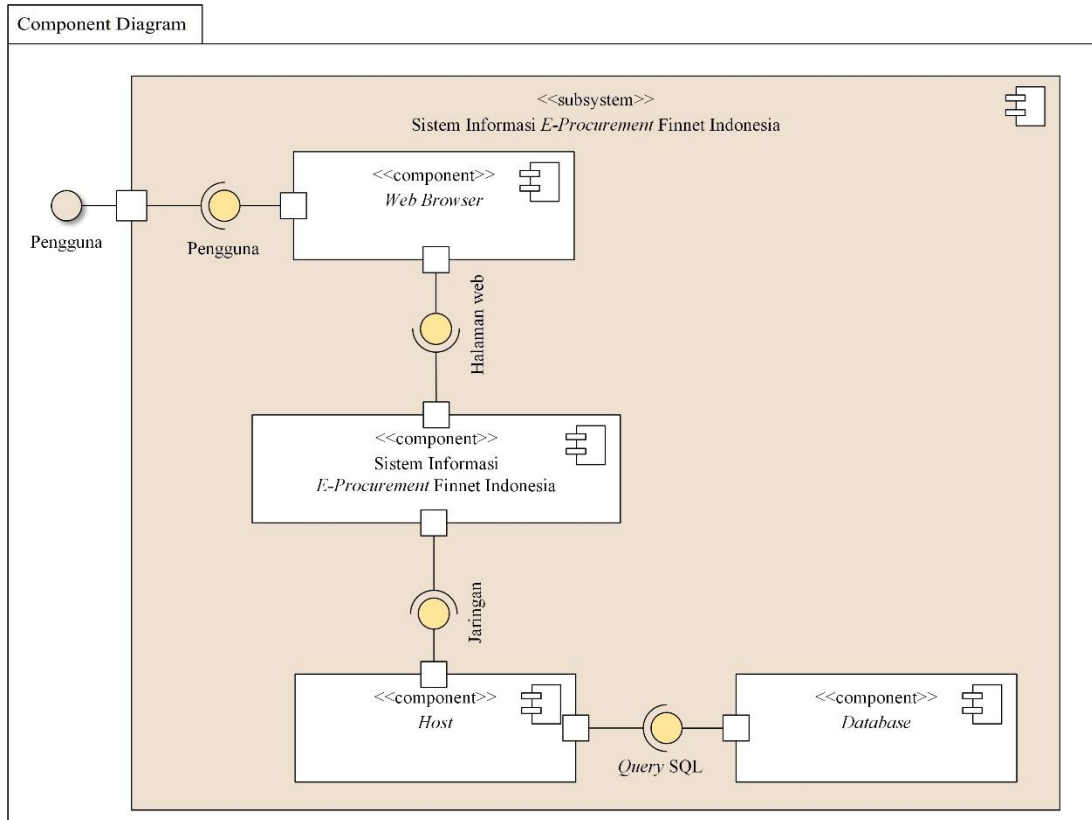
## 2. Kirim Penawaran



Sumber: Hasil Penelitian, 2025

Gambar IV.10. Sequence Diagram Kirim Penawaran

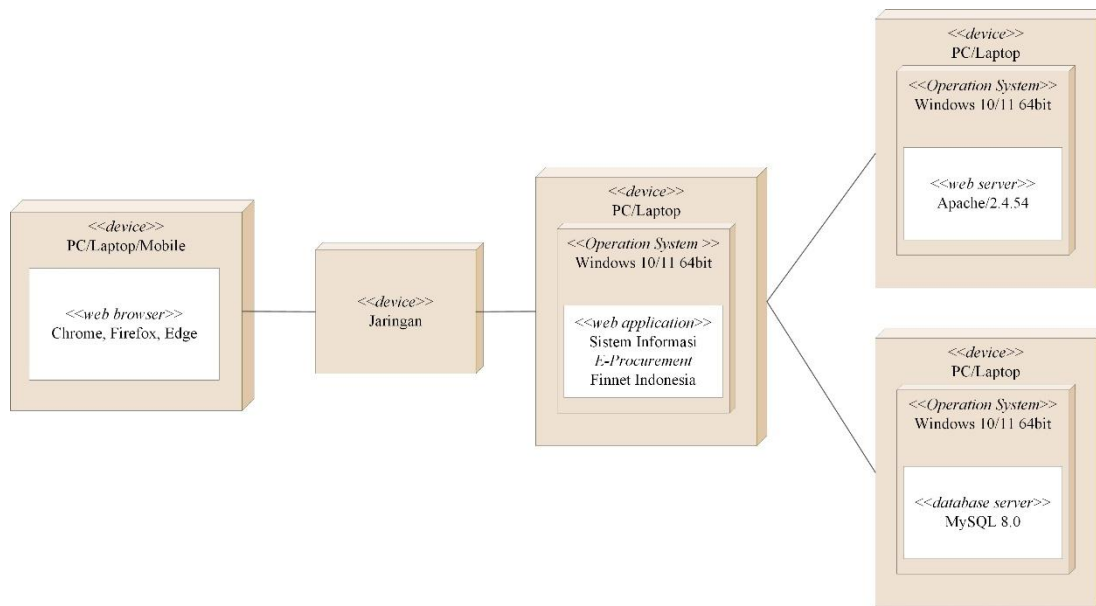
## E. Pemodelan Component Diagram



Sumber: Hasil Penelitian, 2025

Gambar IV.11. *Component Diagram* Sistem Informasi *E-Procurement* Finnet Indonesia

## F. Pemodelan *Deployment Diagram*

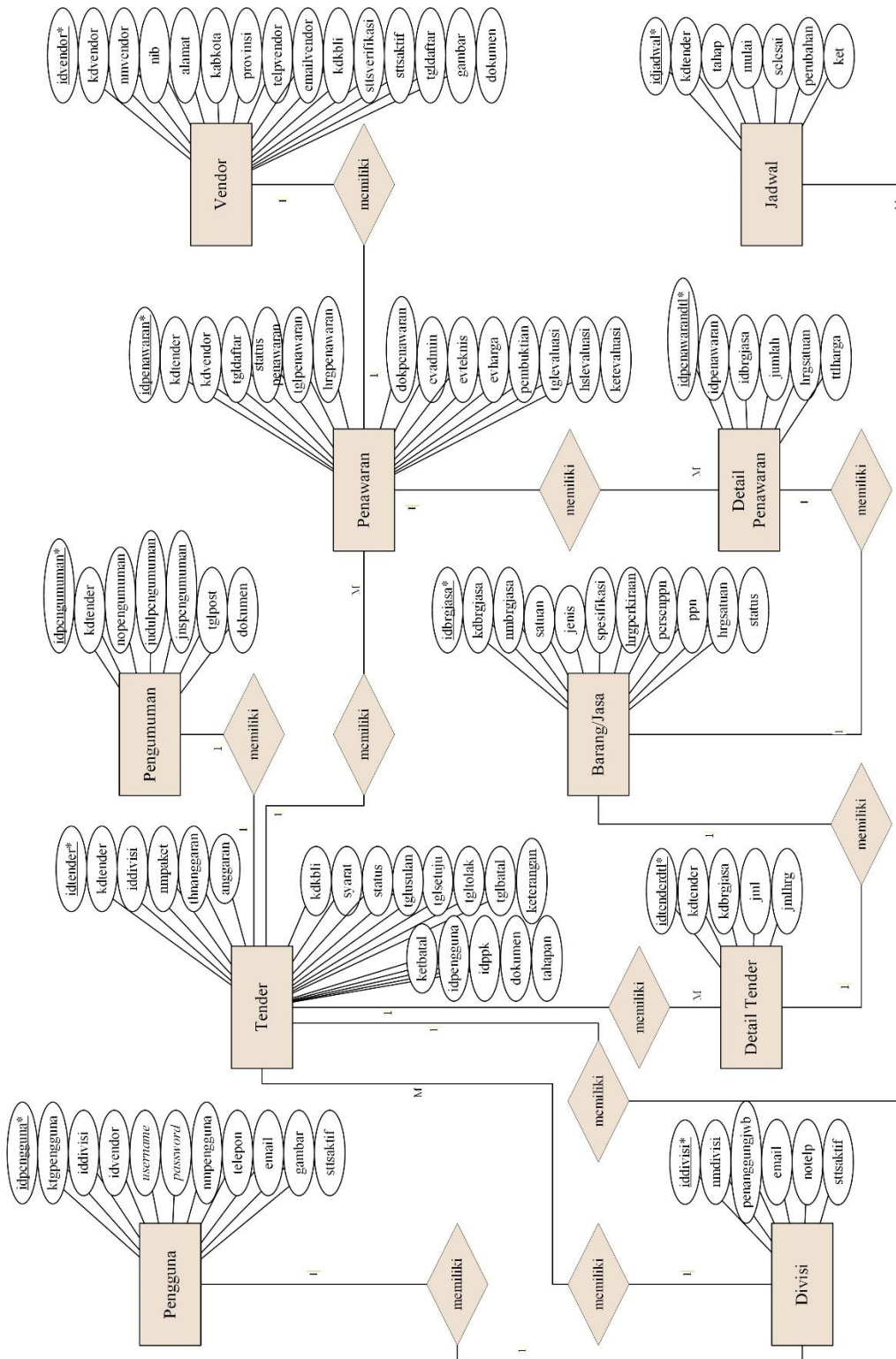


Sumber: Hasil Penelitian, 2025

Gambar IV.12. *Deployment Diagram* Sistem Informasi *E-Procurement* Finnet Indonesia

#### 4.2.2. Desain Pemodelan Data

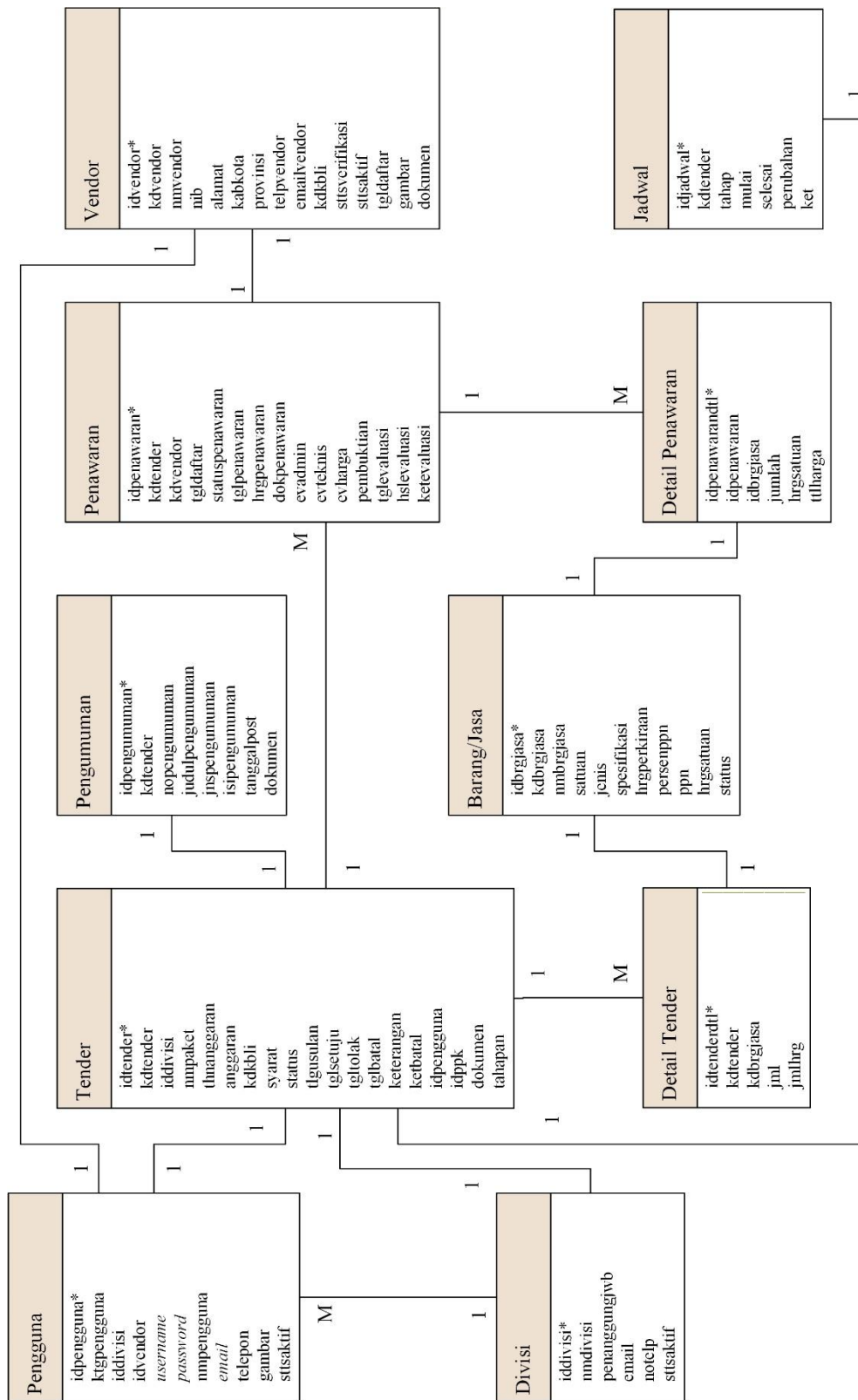
##### A. *Entity Relationship Diagram*



Sumber: Hasil Penelitian, 2025

Gambar IV.13. ERD Sistem Informasi E-Procurement Finnet Indonesia

**B. Logical Record Structure (LRS)**



Sumber: Hasil Penelitian, 2025

Gambar IV.14. LRS System Informasi *E-Procurement* Finnet Indonesia

Gambar *Logical Record Structure* (LRS) di atas menggambarkan struktur data dan relasi antar entitas yang dibuat pada sistem informasi ini. Setiap tabel entitas ditampilkan dengan atribut kolom penyusun tabel pada *database*. Tanda bintang (\*) pada atribut menandakan bahwa kolom tersebut adalah *primary key* pada tabel entitas. Garis-garis relasi antar tabel menunjukkan hubungan dan keterkaitan antara satu dengan yang lain dan bagaimana model atau jenis relasinya seperti 1-1 (satu ke satu) atau 1-M (satu ke banyak).

### C. Spesifikasi File Database

#### 1. Spesifikasi File Tabel Pengguna

Nama *Database* : eproc  
 Nama *File* : Tabel Pengguna  
 Akronim : mst\_pengguna  
 Tipe *File* : *File* Master  
 Akses *File* : Random  
 Panjang *Record* : 320 byte  
 Kunci *Field* : idpengguna

Tabel IV.3. Spesifikasi File Tabel Pengguna

No.	Nama <i>Field</i>	Tipe Data	Ukuran <i>Field</i>	Jenis <i>Key</i>
1	idpengguna	int		<i>Primary Key</i>
2	ktgpengguna	varchar	20	
3	iddivisi	int		<i>Foreign Key</i>
4	idvendor	int		<i>Foreign Key</i>
5	username	varchar	15	
6	password	varchar	60	
7	nmpengguna	varchar	30	
8	email	varchar	30	
9	telepon	varchar	15	
10	gambar	varchar	150	
11	sttsaktif	tinyint		

#### 2. Spesifikasi File Tabel Barang/Jasa

Nama *Database* : eproc  
 Nama *File* : Tabel Barang/Jasa  
 Akronim : mst\_brgjasa  
 Tipe *File* : *File* Master  
 Akses *File* : Random  
 Panjang *Record* : 364 byte  
 Kunci *Field* : idbrgjasa

Tabel IV.4. Spesifikasi File Tabel Barang/Jasa

No.	Nama <i>Field</i>	Tipe Data	Ukuran <i>Field</i>	Jenis <i>Key</i>
1	idbrgjasa	int		<i>Primary Key</i>
2	kdbrgjasa	varchar	6	
3	nmbrgjasa	varchar	300	
4	satuan	varchar	20	
5	jenis	enum		
6	spesifikasi	text		
7	hrgperkiraan	decimal	11	
8	persenppn	varchar	5	
9	ppn	decimal	11	
10	hrgsatuan	decimal	11	
11	status	tinyint		

### 3. Spesifikasi File Tabel Vendor

Nama *Database* : eproc  
 Nama *File* : Tabel Vendor  
 Akronim : mst\_vendor  
 Tipe *File* : *File* Master  
 Akses *File* : Random  
 Panjang *Record* : 786 byte  
 Kunci *Field* : idvendor

Tabel IV.5. Spesifikasi File Tabel Vendor

No.	Nama <i>Field</i>	Tipe Data	Ukuran <i>Field</i>	Jenis <i>Key</i>
1	idvendor	int		<i>Primary Key</i>
2	kdvendor	varchar	6	
3	nmvendor	varchar	60	
4	nib	varchar	20	
5	alamat	text		
6	kabkota	varchar	50	
7	provinsi	varchar	30	
8	telpvendor	varchar	15	
9	emailvendor	varchar	30	
10	kdkbli	varchar	50	
11	sttsverifikasi	varchar	25	
12	sttsaktif	tinyint		
13	tgldaftar	datetime		
14	gambar	varchar	250	
15	dokumen	varchar	250	

#### 4. Spesifikasi File Tabel Divisi

Nama *Database* : eproc

Nama *File* : Tabel Divisi

Akronim : mst\_divisi

Tipe *File* : *File* Master

Akses *File* : Random

Panjang *Record* : 105 byte

Kunci *Field* : iddivisi

Tabel IV.6. Spesifikasi File Tabel Divisi

No.	Nama <i>Field</i>	Tipe Data	Ukuran <i>Field</i>	Jenis <i>Key</i>
1	iddivisi	int		<i>Primary Key</i>
2	nmdivisi	varchar	30	
3	penanggungjwb	varchar	30	
4	email	varchar	30	
5	notelp	varchar	15	
6	sttsaktif	tinyint		

#### 5. Spesifikasi File Tabel Tender

Nama *Database* : eproc  
 Nama *File* : Tabel Tender  
 Akronim : tr\_tender  
 Tipe *File* : *File* Master  
 Akses *File* : Random  
 Panjang *Record* : 803 byte  
 Kunci *Field* : idtender

Tabel IV.7. Spesifikasi File Tabel Tender

No.	Nama <i>Field</i>	Tipe Data	Ukuran <i>Field</i>	Jenis Key
1	idtender	int		<i>Primary Key</i>
2	kdtender	varchar	10	
3	iddivisi	int		<i>Foreign Key</i>
4	nmpaket	varchar	200	
5	thnanggaran	year		
6	anggaran	decimal	18	
7	kdkbli	varchar	50	
8	syarat	text		
9	status	enum		
10	tglusulan	date		
11	tglsetuju	date		
12	tgltolak	date		
13	tglbatal	date		
14	keterangan	text		
15	ketbatal	varchar	250	
16	idpengguna	int		<i>Foreign Key</i>
17	idppk	int		
18	dokumen	varchar	250	
19	tahapan	varchar	25	

#### 6. Spesifikasi File Tabel Detail Tender

Nama *Database* : eproc  
 Nama *File* : Tabel Detail Tender  
 Akronim : tr\_tenderdtl  
 Tipe *File* : *File* Transaksi  
 Akses *File* : Random

Panjang *Record* : 42 byte

Kunci *Field* : idtenderdtl

Tabel IV.8. Spesifikasi File Tabel Detail Tender

No.	Nama <i>Field</i>	Tipe Data	Ukuran <i>Field</i>	Jenis <i>Key</i>
1	idtenderdtl	int		<i>Primary Key</i>
2	kdtender	varchar	10	
3	kdbrgjasa	varchar	6	<i>Foreign Key</i>
4	jml	decimal	8	
5	jmlhrng	decimal	18	

## 7. Spesifikasi File Tabel Penawaran

Nama *Database* : eproc

Nama *File* : Tabel Penawaran

Akronim : tr\_penawaran

Tipe *File* : *File* Transaksi

Akses *File* : Random

Panjang *Record* : 409 byte

Kunci *Field* : idpenawaran

Tabel IV.9. Spesifikasi File Tabel Penawaran

No.	Nama <i>Field</i>	Tipe Data	Ukuran <i>Field</i>	Jenis <i>Key</i>
1	idpenawaran	int		<i>Primary Key</i>
2	kdtender	varchar	10	<i>Foreign Key</i>
3	kdvendord	varchar	6	<i>Foreign Key</i>
4	tgldaftar	datetime		
5	statuspenawaran	varchar	15	
6	tglpenawaran	datetime		
7	hrngpenawaran	decimal	18	
8	dokpenawaran	varchar	250	
9	evadmin	varchar	15	
10	evteknis	varchar	15	
11	evharga	varchar	15	
12	pembuktian	varchar	15	
13	tglevaluasi	datetime		
14	hslevaluasi	varchar	50	
15	ketevaluasi	text		

## 8. Spesifikasi File Tabel Detail Penawaran

Nama *Database* : eproc  
 Nama *File* : Tabel Detail Penawaran  
 Akronim : tr\_penawarandtl  
 Tipe *File* : *File* Transaksi  
 Akses *File* : Random  
 Panjang *Record* : 44 byte  
 Kunci *Field* : idpenawarandtl

Tabel IV.9. Spesifikasi File Tabel Penawaran

No.	Nama <i>Field</i>	Tipe Data	Ukuran <i>Field</i>	Jenis Key
1	idpenawarandtl	int		<i>Primary Key</i>
2	idpenawaran	int		<i>Foreign Key</i>
3	idbrgjasa	int		<i>Foreign Key</i>
4	jumlah	decimal	8	
5	hrgsatuan	decimal	18	
6	ttlharga	decimal	18	

## 9. Spesifikasi File Tabel Jadwal

Nama *Database* : eproc  
 Nama *File* : Tabel Jadwal  
 Akronim : tr\_jadwal  
 Tipe *File* : *File* Transaksi  
 Akses *File* : Random  
 Panjang *Record* : 289 byte  
 Kunci *Field* : idjadwal

Tabel IV.9. Spesifikasi File Tabel Jadwal

No.	Nama <i>Field</i>	Tipe Data	Ukuran <i>Field</i>	Jenis Key
1	idjadwal	int		<i>Primary Key</i>
2	kdtender	varchar	10	<i>Foreign Key</i>

3	tahap	varchar	20	
4	mulai	date		
5	selesai	date		
6	perubahan	varchar	9	
7	ket	varchar	250	

#### 10. Spesifikasi File Tabel Pengumuman

Nama *Database* : eproc  
 Nama *File* : Tabel Pengumuman  
 Akronim : tr\_pengumuman  
 Tipe *File* : *File* Transaksi  
 Akses *File* : Random  
 Panjang *Record* : 560 byte  
 Kunci *Field* : idpengumuman

Tabel IV.9. Spesifikasi File Tabel Pengumuman

No.	Nama <i>Field</i>	Tipe Data	Ukuran <i>Field</i>	Jenis <i>Key</i>
1	idpengumuman	int		<i>Primary Key</i>
2	kdtender	varchar	10	<i>Foreign Key</i>
3	nopengumuman	varchar	30	
4	judulpengumuman	varchar	250	
5	jnspengumuman	varchar	20	
6	isipengumuman	text		
7	tglpost	datetime		
8	dokumen	varchar	250	

#### 11. Spesifikasi File Tabel KBLI

Nama *Database* : eproc  
 Nama *File* : Tabel KBLI

Akronim : mst\_kbli  
 Tipe File : File Support  
 Akses File : Random  
 Panjang Record : 255 byte  
 Kunci Field : idkbli

Tabel IV.9. Spesifikasi File Tabel KBLI

No.	Nama Field	Tipe Data	Ukuran Field	Jenis Key
1	idkbli	int		Primary Key
2	kdkbli	varchar	5	
3	nmkbli	varchar	250	

## 12. Spesifikasi File Tabel Tahapan

Nama Database : eproc  
 Nama File : Tabel Tahapan  
 Akronim : mst\_tahapan  
 Tipe File : File Support  
 Akses File : Random  
 Panjang Record : 275 byte  
 Kunci Field : idtahapan

Tabel IV.9. Spesifikasi File Tabel Tahapan

No.	Nama Field	Tipe Data	Ukuran Field	Jenis Key
1	idtahapan	int		Primary Key
2	nmtahapan	varchar	25	
3	wkttahapan	int		
4	kettahapan	varchar	250	

## 13. Spesifikasi File Tabel Satuan

Nama Database : eproc  
 Nama File : Tabel Satuan

Akronim : mst\_satuan  
 Tipe File : File Support  
 Akses File : Random  
 Panjang Record : 255 byte  
 Kunci Field : idkbli

Tabel IV.9. Spesifikasi File Tabel Satuan

No.	Nama Field	Tipe Data	Ukuran Field	Jenis Key
1	idsatuan	int		Primary Key
2	nmsatuan	varchar	50	

### 4.2.3. Desain User Interface

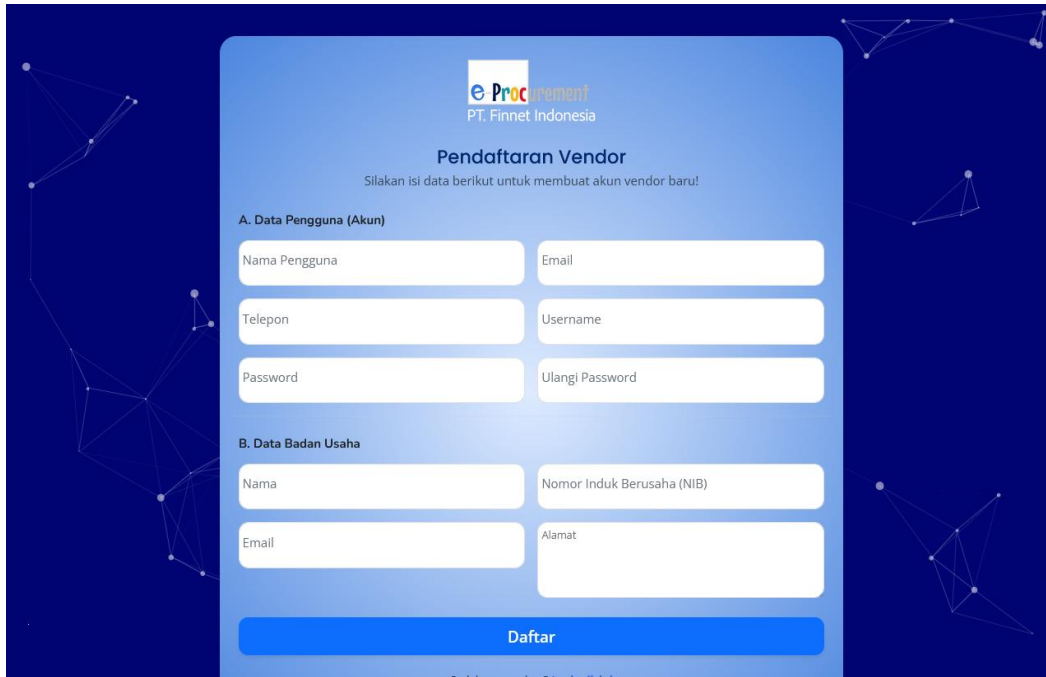
#### 1. Halaman Depan



Sumber : Hasil Penelitian, 2025

Gambar IV.15. Desain Halaman Depan

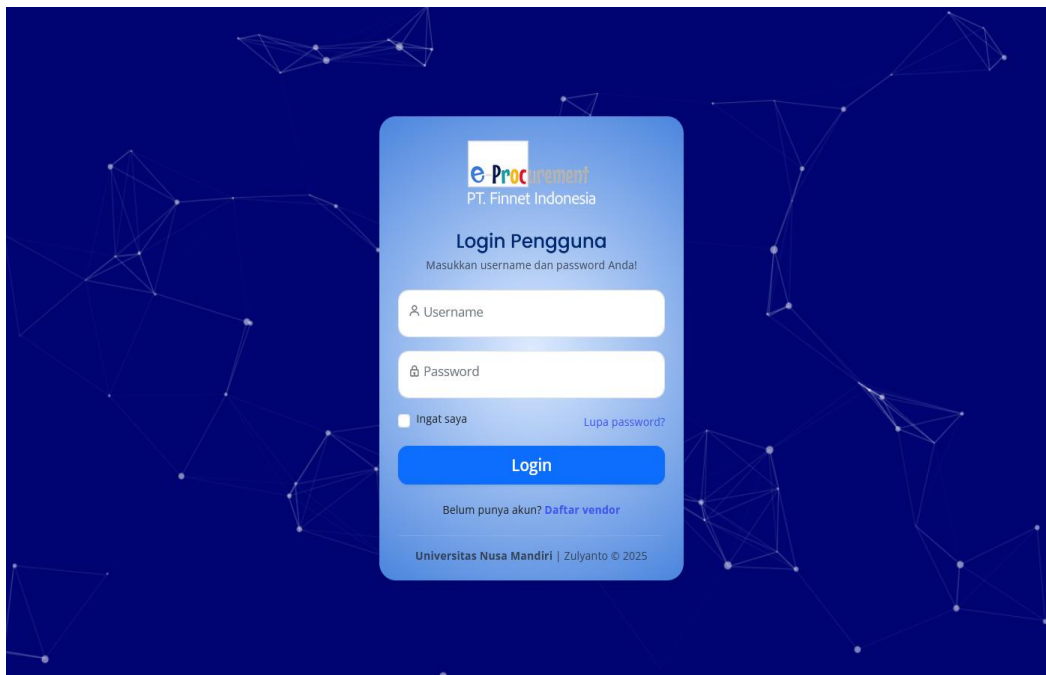
#### 2. Halaman Daftar Vendor



Sumber : Hasil Penelitian, 2025

Gambar IV.16. Desain Halaman Daftar Vendor

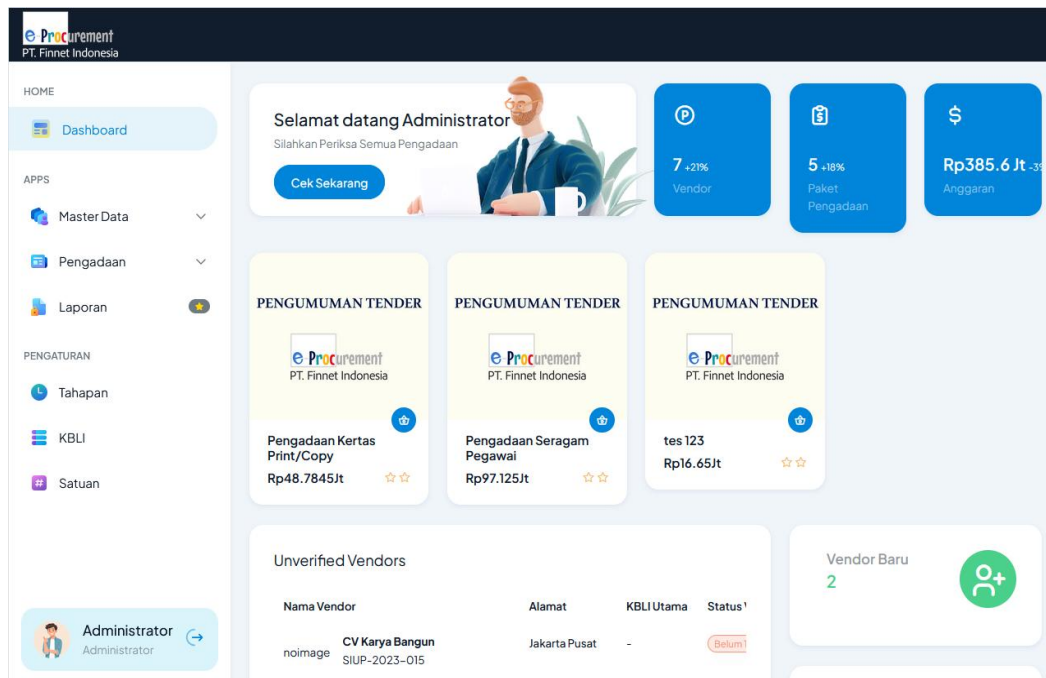
### 3. Halaman *Login* Pengguna



Sumber : Hasil Penelitian, 2025

Gambar IV.17. Desain Halaman *Login* Pengguna

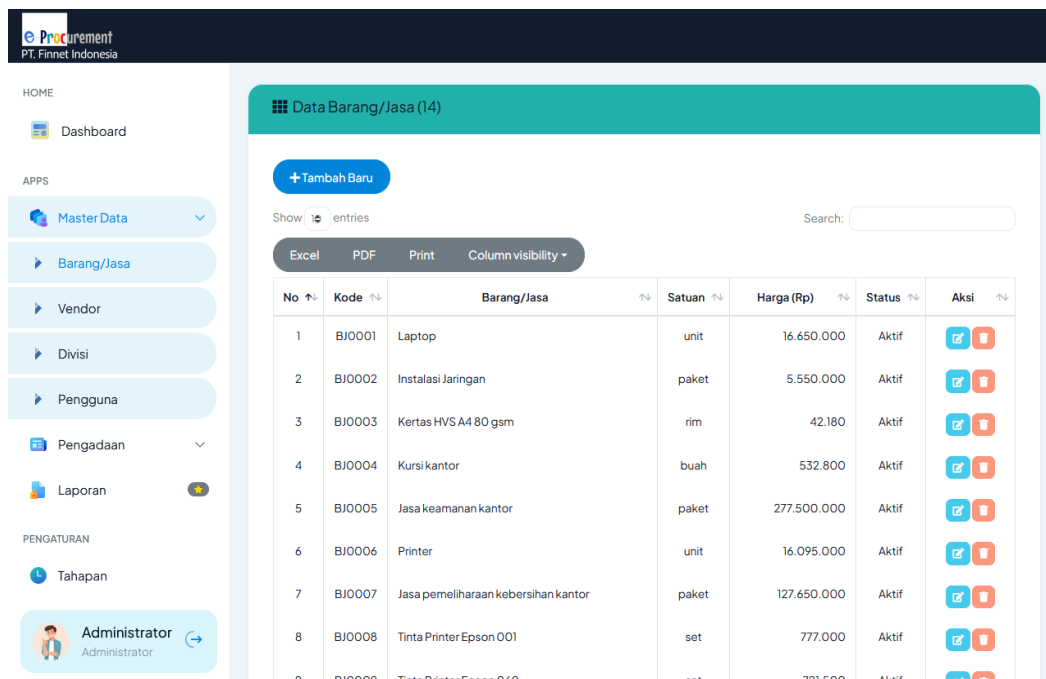
### 4. Halaman *Dashboard* Administrator



Sumber : Hasil Penelitian, 2025

Gambar IV.18. Desain Halaman *Dashboard* Administrator

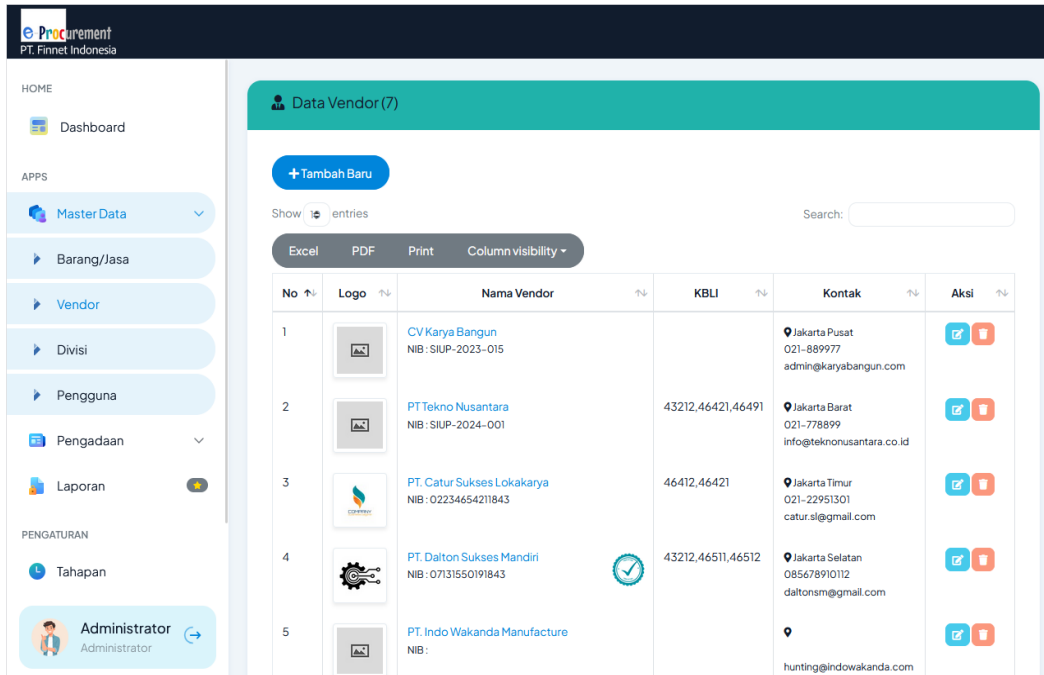
## 5. Halaman Data Barang/Jasa



Sumber : Hasil Penelitian, 2025

Gambar IV.19. Desain Halaman Data Barang/Jasa

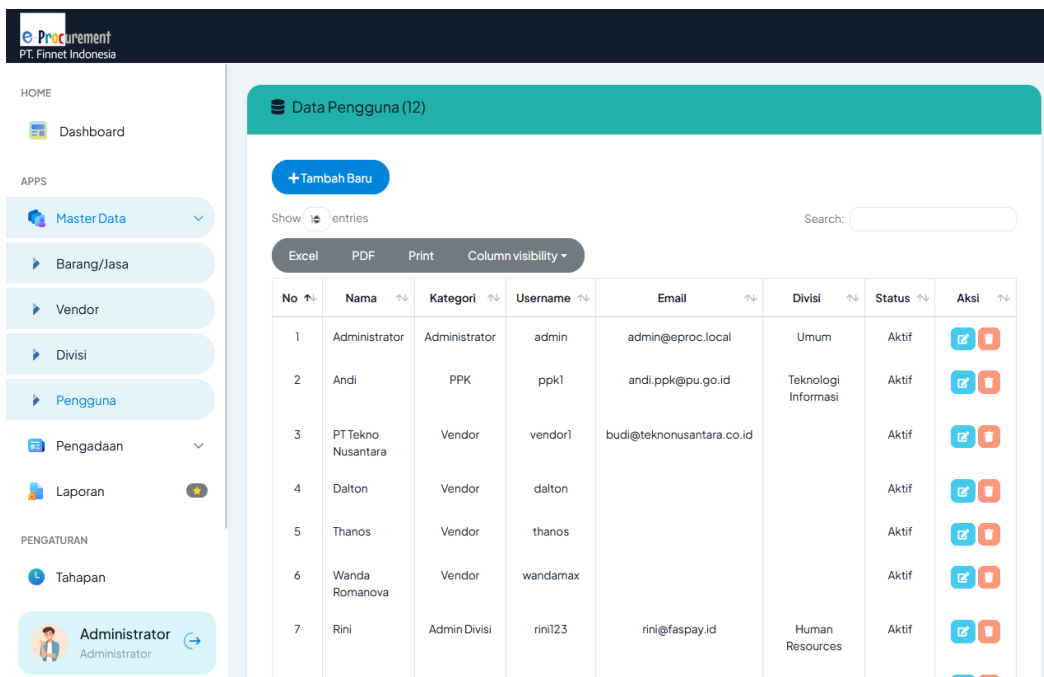
## 6. Halaman Data Vendor



Sumber : Hasil Penelitian, 2025

Gambar IV.20. Desain Halaman Data Vendor

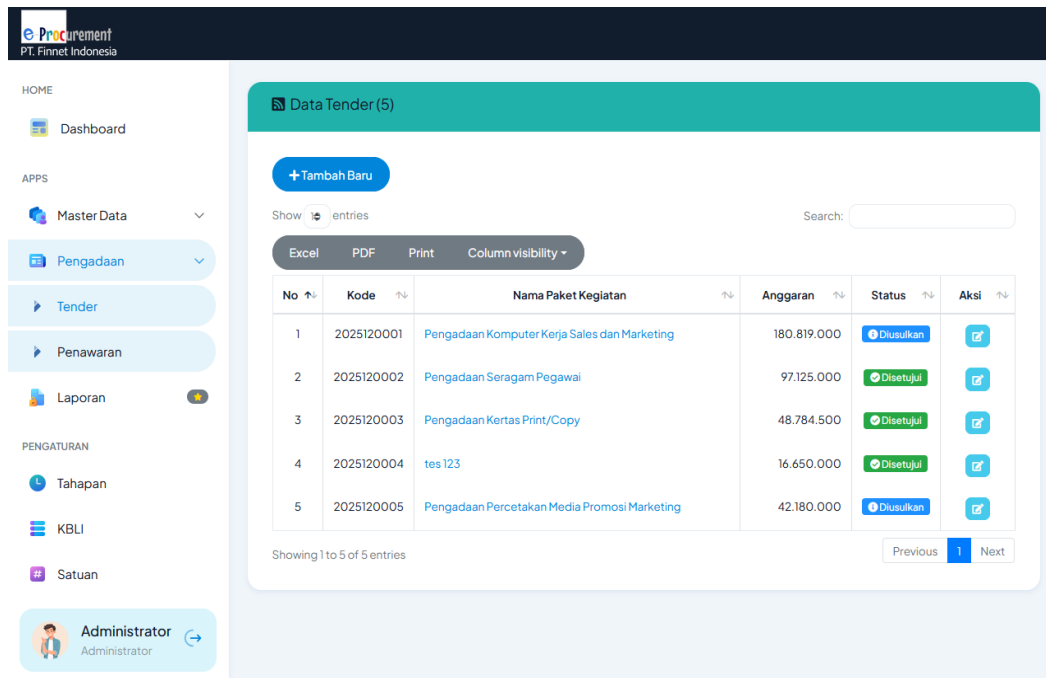
## 7. Halaman Data Pengguna



Sumber : Hasil Penelitian, 2025

Gambar IV.21. Desain Halaman Data Pengguna

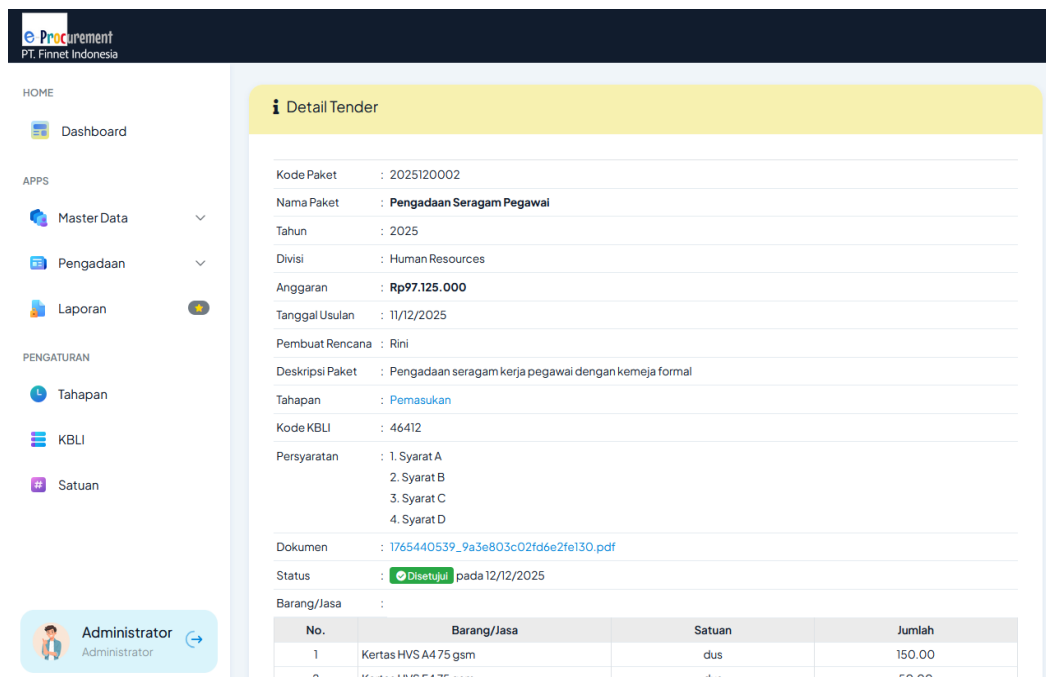
## 8. Halaman Data Tender



Sumber : Hasil Penelitian, 2025

Gambar IV.22. Desain Halaman Data Tender

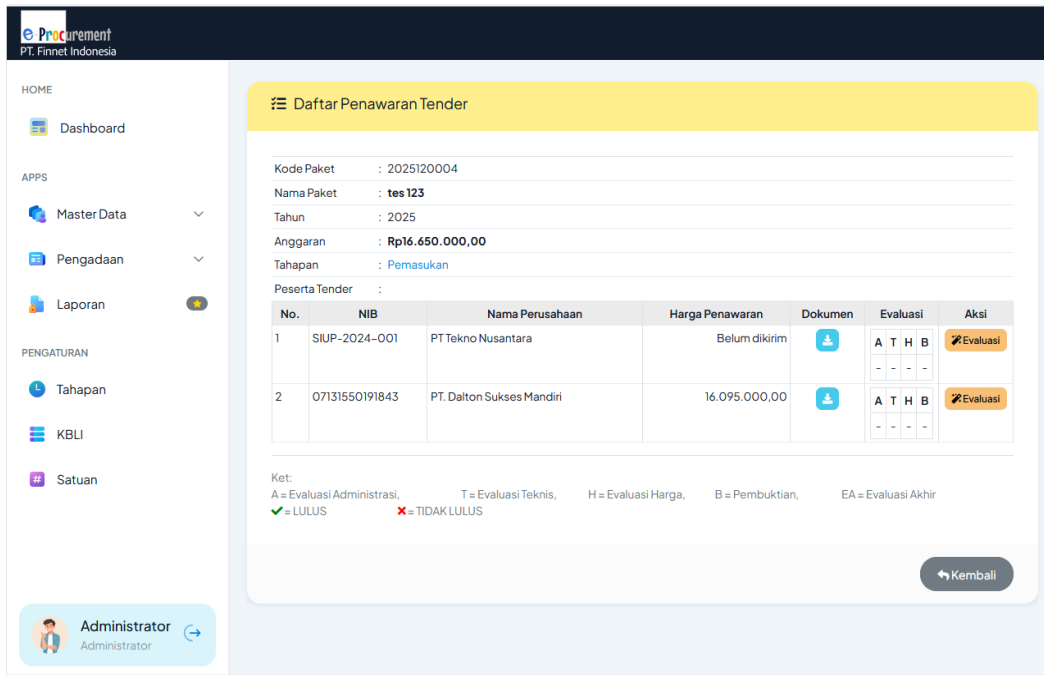
## 9. Halaman Detail Tender



Sumber : Hasil Penelitian, 2025

Gambar IV.23. Desain Halaman Detail Tender

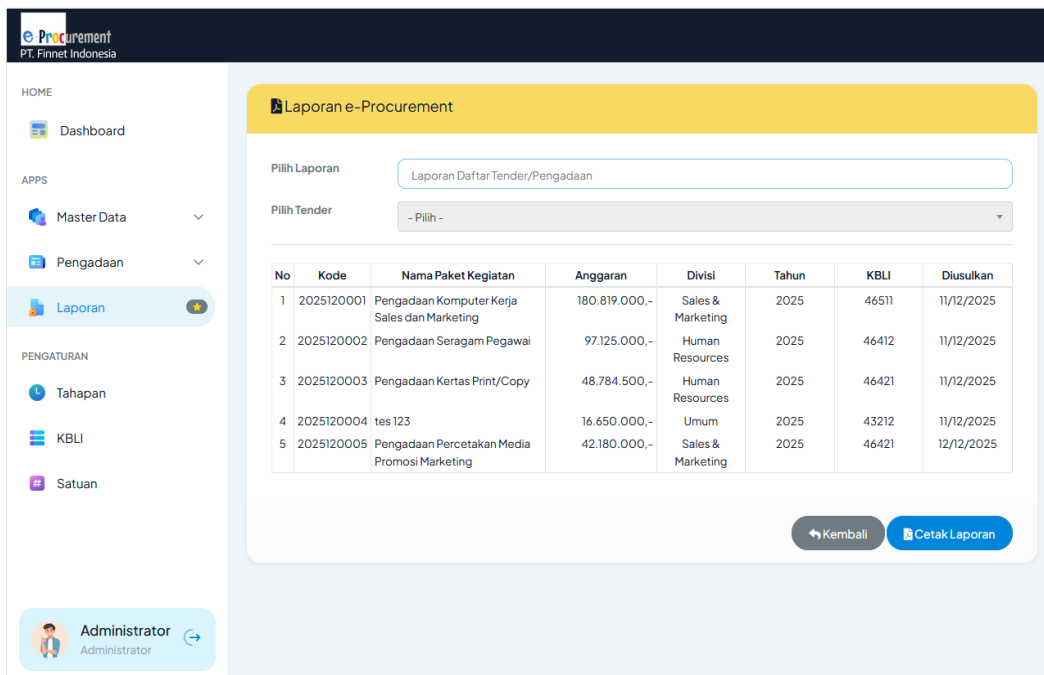
## 10. Halaman Daftar Penawaran Tender



Sumber : Hasil Penelitian, 2025

Gambar IV.24. Desain Halaman Daftar Penawaran Tender

### 11. Halaman Laporan E-Procurement



Sumber : Hasil Penelitian, 2025

Gambar IV.25. Desain Halaman Daftar Penawaran Tender

### 4.3. Code Generation

Tahap code generation merupakan proses penerjemahan hasil perancangan sistem ke dalam bentuk kode program yang dapat dijalankan oleh komputer. Pada Sistem Informasi *E-Procurement* PT. Finnet Indonesia, proses ini dilakukan dengan mengimplementasikan rancangan UML, struktur basis data, serta alur bisnis ke dalam kode program menggunakan *framework* CodeIgniter yang menerapkan arsitektur *Model-View-Controller* (MVC). Setiap modul utama seperti pengelolaan data barang/jasa, vendor, tender, penawaran, pengguna, serta laporan dikembangkan secara terpisah sesuai fungsinya untuk meningkatkan keteraturan kode dan kemudahan pemeliharaan sistem.

Dalam proses *code generation*, bahasa pemrograman PHP digunakan untuk membangun logika aplikasi pada bagian *controller* dan *model*, sedangkan view dikembangkan menggunakan HTML, CSS, dan JavaScript untuk menghasilkan antarmuka pengguna yang interaktif dan responsif. Pengelolaan data dilakukan melalui sistem manajemen basis data MySQL yang terintegrasi dengan model CodeIgniter untuk mendukung operasi *Create, Read, Update, dan Delete* (CRUD).

## 1. Tambah Data Pengguna

```
public function tambah()
{
    checklogin();
    $m_pengguna = new MdlPengguna();
    $m_divisi = new MdlDivisi();
    $pengguna = $m_pengguna->listing();
    $divisi = $m_divisi->optionlist();
    $total = $m_pengguna->total();

    if($this->request->getMethod() === 'post' && $this->validate(
        [
            'nmpengguna'=>'required',
            'username'=>'required',
            'password'=>'required',
            'gambar' => [
                'mime_in[gambar,image/jpg,image/jpeg,image/gif,image/png]',
                'max_size[gambar,4096]',
            ],
        ])) {
```

```

        $existing_pengguna = $m_pengguna->where('username', $this->request-
>getPost('username'))->first();
        if ($existing_pengguna) {
            $this->session->setFlashdata('warning', 'User sudah terdaftar.');
```

```

            return redirect()->to(base_url('admin/pengguna/tambah'));
        }

        if(!empty($_FILES['gambar']['name'])) {
            $avatar          = $this->request->getFile('gambar');
            $namabarbaru    = $avatar->getRandomName('gambar');
            $avatar->move(WRITEPATH
'../assets/upload/pengguna/', $namabarbaru);
            $image = \Config\Services::image()
->withFile(WRITEPATH . '../assets/upload/pengguna/'. $namabarbaru)
->fit(100, 100, 'center')
->save(WRITEPATH
'../assets/upload/pengguna/thumbs/'. $namabarbaru);
            $data = [
                'iddivisi' => $this->request->getPost('iddivisi'),
                'nmpengguna' => $this->request->getPost('nmpengguna'),
                'email' => $this->request->getPost('email'),
                'telepon' => $this->request->getPost('telepon'),
                'username' => $this->request->getPost('username'),
                'password' => sha1($this->request->getPost('password')),
                'gambar' => $namabarbaru,
                'ktgpengguna' => $this->request->getPost('ktgpengguna'),
            ];
        }else{
            $data = [
                'iddivisi' => $this->request->getPost('iddivisi'),
                'nmpengguna' => $this->request->getPost('nmpengguna'),
                'email' => $this->request->getPost('email'),
                'telepon' => $this->request->getPost('telepon'),
                'username' => $this->request->getPost('username'),
                'password' => sha1($this->request->getPost('password')),
                'ktgpengguna' => $this->request->getPost('ktgpengguna'),
            ];
        }
        $m_pengguna->tambah($data);
        $this->session->setFlashdata('sukses', 'Data telah ditambah');
        return redirect()->to(base_url('admin/pengguna'));
    }else{
        $data = [
            'title' => 'Tambah User',
            'pengguna' => $pengguna,
            'divisi' => $divisi,
            'content' => 'admin/pengguna/tambah'
        ];
        echo view('admin/layout/wrapper', $data);
    }
}
}

```

Potongan kode program di atas merupakan implementasi fungsi tambah() pada *controller* pengguna yang digunakan untuk mengelola proses penambahan data pengguna pada Sistem Informasi E-Procurement PT. Finnet Indonesia. Proses diawali dengan pengecekan autentikasi melalui fungsi checklogin() untuk memastikan bahwa hanya pengguna yang memiliki hak akses yang dapat menambahkan data pengguna baru. Selanjutnya, sistem memanggil *model* MdlPengguna dan MdlDivisi untuk mengambil data pengguna, daftar divisi, serta total pengguna yang diperlukan sebagai informasi pendukung pada tampilan formulir penambahan pengguna.

Pada saat permintaan data dilakukan menggunakan metode POST dan seluruh data input telah lolos proses validasi, sistem akan memeriksa ketersediaan *username* untuk mencegah duplikasi data pengguna. Apabila *username* belum terdaftar, sistem akan memproses data yang diinputkan, termasuk pengunggahan foto pengguna apabila tersedia. File gambar yang diunggah akan disimpan ke dalam direktori sistem dan diproses lebih lanjut untuk menghasilkan *thumbnail* berukuran 100×100 piksel. Data pengguna kemudian disimpan ke dalam basis data dengan menerapkan enkripsi pada kata sandi menggunakan fungsi *hashing*. Setelah proses penyimpanan berhasil, sistem memberikan notifikasi keberhasilan kepada pengguna dan mengarahkan kembali ke halaman manajemen pengguna. Apabila validasi data tidak terpenuhi, sistem akan menampilkan kembali halaman formulir penambahan pengguna beserta data pendukung yang diperlukan.

## 2. Daftar Vendor

```
public function daftar()
{
    $session = \Config\Services::session();
    $m_pengguna = new MdlPengguna();
    $m_vendor = new MdlVendor();
    $kodeOto = $m_vendor->getKodeOtomatis();

    if($this->request->getMethod() === 'post' && $this->validate(
        [
            'nmpengguna'=>'required',
            'username'=>'required',
            'password'=>'required',
```

```

        'passwordknf'=>'required|matches[password]',
        'nmvendor'=>'required',
        'npwp'=>'required',
        'emailvendor'=>'required|valid_email',
        'alamat'=>'required',
    ))) {

        $existing_pengguna = $m_pengguna->where('username', $this->request->getPost('username'))->first();
        if ($existing_pengguna) {
            $this->session->setFlashdata('warning', 'Username sudah digunakan. ');
            return redirect()->to(base_url('vendor/daftar'));
        }
        $datavendor = [
            'kdvendor' => $kodeOto,
            'nmvendor' => $this->request->getPost('nmvendor'),
            'npwp' => $this->request->getPost('npwp'),
            'emailvendor' => $this->request->getPost('emailvendor'),
            'alamat' => $this->request->getPost('alamat'),
        ];
        $m_vendor->tambah($datavendor);
        $vendor = $m_vendor->detail($kodeOto);
        $data = [
            'nmpengguna' => $this->request->getPost('nmpengguna'),
            'username' => $this->request->getPost('username'),
            'password' => sha1($this->request->getPost('password')),
            'ktgpengguna' => 'Vendor',
            'idvendor' => $vendor['idvendor']
        ];
        $simpan = $m_pengguna->tambah($data);
        return redirect()->to(base_url('login'))->with('sukses', 'Pendaftaran berhasil!');
    }

    echo view('vendor/daftar/index', ['session' => $session,]);
}

```

Potongan kode program di atas merupakan implementasi fungsi `daftar()` yang digunakan untuk menangani proses pendaftaran vendor pada Sistem Informasi E-Procurement PT. Finnet Indonesia. Proses diawali dengan inialisasi sesi serta pemanggilan *model* `MdlPengguna` dan `MdlVendor` untuk mengelola data pengguna dan vendor. Sistem juga menghasilkan kode vendor secara otomatis melalui fungsi `getKodeOtomatis()` yang bertujuan untuk memastikan setiap vendor memiliki identitas unik di dalam basis data. Selanjutnya, sistem melakukan validasi terhadap data masukan yang dikirimkan melalui metode POST, meliputi data pengguna, data perusahaan vendor, serta kesesuaian kata sandi dan format alamat surat elektronik.

Apabila seluruh data telah lolos proses validasi, sistem akan melakukan pengecekan ketersediaan *username* untuk mencegah terjadinya duplikasi akun. Setelah dipastikan *username* belum digunakan, data vendor akan disimpan terlebih dahulu ke dalam basis data, kemudian sistem mengambil kembali data vendor tersebut untuk memperoleh identitas vendor yang akan dikaitkan dengan akun pengguna. Data pengguna selanjutnya disimpan dengan kategori pengguna sebagai *Vendor* dan kata sandi yang telah diamankan menggunakan metode *hashing*. Setelah seluruh proses pendaftaran berhasil, sistem akan mengarahkan pengguna ke halaman *login* disertai notifikasi bahwa proses pendaftaran telah berhasil dilakukan. Apabila tidak terdapat data yang dikirimkan, sistem akan menampilkan halaman formulir pendaftaran vendor.

### 3. Tambah Penawaran

```
public function tambah($kdtender)
{
    checklogin();
    $session = \Config\Services::session();
    $m_penawaran = new MdlPenawaran();
    $m_penawarandtl = new MdlPenawaranDtl();
    $m_tender = new MdlTender();
    $tender = $m_tender->detail($kdtender);
    $brgjasa = $m_tender->brgjasa($kdtender);

    if($this->request->getMethod() === 'post' && $this->validate(
        [
            'kdtender'=>'required',
            'kdvendor'=>'required',
            'hrgpenawaran_input'=>'required',
        ])) {
        $kdtender = $this->request->getPost('kdtender');
        $kdvendor = $this->request->getPost('kdvendor');
        $penawaran = $m_penawaran->detailByVendor($kdtender);
        $idpenawaran = $penawaran['idpenawaran'];
        $dataHeader = [
            'idpenawaran' => $idpenawaran,
            'kdtender' => $kdtender,
            'kdvendor' => $kdvendor,
            'statuspenawaran' => "Sudah dikirim",
            'tglpenawaran' => date('Y-m-d H:i:s'),
            'hrgpenawaran' => $this->request->getPost('hrgpenawaran_input')
        ];

        $file = $this->request->getFile('dokpenawaran');
```

```

if ($file->isValid() && !$file->hasMoved()) {
    $nama = $file->getRandomName();
    $file->move(WRITEPATH . '../assets/upload/dokumen', $nama);
    $dataHeader['dokpenawaran'] = $nama;
}

$m_penawaran->kirim($dataHeader);

$idbrgjasa = $this->request->getPost('idbrgjasa');
$jumlah = $this->request->getPost('jumlah');
$hrgsatuan = $this->request->getPost('hrgsatuan');
$ttharga = $this->request->getPost('ttharga_input');

for ($i = 0; $i < count($idbrgjasa); $i++) {
    $m_penawarandtl->tambah([
        'idpenawaran' => $idpenawaran,
        'idbrgjasa' => $idbrgjasa[$i],
        'jumlah' => $jumlah[$i],
        'hrgsatuan' => $hrgsatuan[$i],
        'ttharga' => $ttharga[$i],
    ]);
}
$this->session->setFlashdata('sukses','Penawaran Anda telah dikirim');
return redirect()->to(base_url('admin/paket/vendor/'.$session->get('kdvendor')));
}else{
    $data = [
        'title' => 'Kirim Penawaran',
        'tender' => $tender,
        'brgjasa' => $brgjasa,
        'content' => 'admin/penawaran/tambah'
    ];
    echo view('admin/layout/wrapper',$data);
}
}

```

Potongan kode program di atas merupakan implementasi fungsi tambah() pada *controller* penawaran yang berperan dalam proses pengiriman data penawaran oleh vendor pada Sistem Informasi E-Procurement PT. Finnet Indonesia. Fungsi ini diawali dengan proses pengecekan autentikasi pengguna melalui checklogin() untuk memastikan hanya pengguna yang berwenang yang dapat mengakses fitur tersebut. Selanjutnya, sistem menginisialisasi beberapa *model* yang berkaitan dengan penawaran, detail penawaran, dan tender. Data tender serta daftar barang/jasa yang terkait dengan kode tender (\$kdtender) diambil dari basis data dan digunakan sebagai informasi pendukung pada tampilan formulir pengajuan penawaran.

Pada saat metode permintaan (*request*) berupa POST dan seluruh data input telah lolos proses validasi, sistem akan memproses data penawaran yang dikirimkan oleh vendor. Data utama penawaran disimpan dalam bentuk *header* yang mencakup informasi tender, vendor, status penawaran, tanggal pengiriman, nilai total penawaran, serta dokumen pendukung apabila diunggah. Selanjutnya, sistem memproses data detail penawaran berupa rincian barang/jasa, jumlah, harga satuan, dan total harga yang disimpan secara berulang menggunakan perulangan (*looping*). Setelah seluruh data berhasil disimpan, sistem memberikan notifikasi keberhasilan kepada pengguna dan mengarahkan kembali ke halaman paket vendor. Apabila kondisi validasi tidak terpenuhi, sistem akan menampilkan kembali halaman formulir pengiriman penawaran beserta data tender dan barang/jasa terkait.

#### 4. Kode View Detail Paket

```
<main id="main">
<section class="content">
  <div class="container-fluid">
    <div class="row">
      <div class="col">
        <div class="card">
          <div class="card-header" style="background-color: #F7F1B0;">
            <h3 class="card-title"><i class="fas fa-info"></i>
              &nbsp;   <?= $title ?></h3>
          </div>
          <div class="card-body">
            <!-- <table class="table table-bordered" id="example1"> -->
            <table class="table">
              <tr><td width="14%">Kode Paket</td><td width="1%">:</td><td><?=
                $tender['kdtender']; ?></td></tr>
              <tr><td>Nama Paket</td><td>:</td><td><b><?= $tender['nmpaket'];
                ?></b></td></tr>
              <tr><td>Tahun</td><td>:</td><td><?= $tender['thnanggaran'];
                ?></td></tr>
              <tr><td>Divisi</td><td>:</td><td><?= $tender['nmdivisi'];
                ?></td></tr>
              <tr><td>Anggaran</td><td>:</td><td><b>Rp<?=
                number_format($tender['anggaran'],0,',','.')> ?></b></td></tr>
              <tr><td>Tanggal Usulan</td><td>:</td><td><?=
                tanggal_id($tender['tglusulan']); ?></td></tr>
              <tr><td>Pembuat Rencana</td><td>:</td><td><?= $tender['username'];
                ?></td></tr>
            </table>
          </div>
        </div>
      </div>
    </div>
  </div>
</section>
</main>
```

```

        <tr><td>Deskripsi Paket</td><td>:</td><td><?=$tender['keterangan'];
?></td></tr>
        <tr><td>Rencana Tender</td><td>:</td><td>
        <table class="table table-sm table-bordered">
            <tr><th style="background-color: #EBEBEB;">Mulai</th><th
style="background-color: #EBEBEB;">Selesai</th></tr>
            <tr><td><?=$tender['mulaitender'];?></td><td><?=$tender['selesaitender'];?></td></tr>
        </table>
        <tr><td>Rencana Pelaksanaan</td><td>:</td><td>
        <table class="table table-sm table-bordered">
            <tr><th style="background-color: #EBEBEB;">Mulai</th><th
style="background-color: #EBEBEB;">Selesai</th></tr>
            <tr><td><?=$tender['mulaipelaksanaan'];?></td><td><?=$tender['selesaipelaksanaan'];?></td></tr>
        </table>
        <tr><td>Dokumen</td><td>:</td><td><?=$tender['dokumen'];
?></td></tr>
        <tr><td>Persyaratan</td><td>:</td><td><?=$tender['syarat'];
?></td></tr>
<?php
$session = \Config\Services::session();
if ($session->get('ktgpengguna') == 'Administrator'){
    ?>
        <tr><td>Status</td><td>:</td><td>
        <?php
            if ($tender['status'] == 'Diusulkan'): ?>
                <span class="badge badge-primary"><i class="fas fa-info-
circle"></i> <?=$tender['status'] ?></span>
                pada <?=$tanggal_id($tender['tglusulan']) ?>
                <?php elseif ($tender['status'] == 'Disetujui'): ?>
                <span class="badge badge-success"><i class="fas fa-check-
circle"></i> <?=$tender['status'] ?></span>
                pada <?=$tanggal_id($tender['tglsetuju']) ?>
                <?php elseif ($tender['status'] == 'Ditolak'): ?>
                <span class="badge badge-danger"><i class="fas fa-times-
circle"></i> <?=$tender['status'] ?></span>
                pada <?=$tanggal_id($tender['tgltolak']) ?>
                <?php elseif ($tender['status'] == 'Paket'): ?>
                <span class="badge badge-warning"><i class="fas fa-times-
circle"></i> <?=$tender['status'] ?></span>
                pada <?=$tanggal_id($tender['tglpaket']) ?>
                <?php else: ?>
                <span class="badge badge-info"><i class="fas fa-exclamation-
circle"></i><?=$tender['status'] ?></span>
                <?php endif; ?>
            </td></tr>
        <?php
            if ($tender['status'] == 'Diusulkan'): ?>
        <tr><td></td><td></td><td>
            <button type="reset" onclick="goBack()" class="btn btn-success
btn-custom"><i class="fa fa-check-circle"></i> &nbsp;&nbsp;Setujui</button>

```

```

        <button type="reset" onclick="goBack()" class="btn btn-danger"><i
class="fa fa-times-circle"></i> &nbsp;&nbsp;&nbsp;Tolak</button>
    </td></tr>
    <?php elseif ($tender['status'] == 'Disetujui'): ?>
    <tr><td></td><td></td><td>
        <button type="reset" onclick="goBack()" class="btn btn-
warning"><i class="fas fa-gavel"></i> &nbsp;&nbsp;&nbsp;Buat Paket</button>
    </td></tr>
    <?php endif; ?>
<?php } ?>
</table>
</div>

<div class="card-footer" align="right">
    <button type="reset" onclick="goBack()" class="btn btn-secondary"><i
class="fa fa-reply"></i> Kembali</button>
</div>

```

Potongan kode di atas merupakan implementasi antarmuka pengguna (*view*) yang berfungsi untuk menampilkan informasi detail paket tender pada Sistem Informasi E-Procurement PT. Finnet Indonesia. Tampilan disusun menggunakan komponen *card* dan *table* berbasis HTML dan Bootstrap untuk menghasilkan tata letak yang rapi dan mudah dipahami oleh pengguna. Informasi yang ditampilkan meliputi kode paket, nama paket, tahun anggaran, divisi, nilai anggaran, tanggal usulan, pembuat rencana, deskripsi paket, serta jadwal rencana tender dan pelaksanaan. Selain itu, sistem juga menampilkan dokumen dan persyaratan tender yang telah disimpan sebelumnya, sehingga pengguna dapat memperoleh gambaran lengkap terkait paket yang sedang ditinjau.

Pada bagian bawah tampilan, sistem menerapkan logika kondisional berdasarkan kategori pengguna dan status tender yang sedang berlangsung. Apabila pengguna memiliki hak akses sebagai administrator, maka status tender akan ditampilkan dalam bentuk *badge* dengan warna dan ikon yang berbeda sesuai kondisi, seperti diusulkan, disetujui, ditolak, atau telah menjadi paket. Sistem juga menyediakan tombol aksi yang bersifat dinamis, seperti menyetujui, menolak, atau membuat paket tender, yang hanya muncul pada status tertentu. Pendekatan ini bertujuan untuk meningkatkan kejelasan informasi, mencegah kesalahan

tindakan, serta memastikan bahwa setiap proses pengelolaan tender berjalan sesuai dengan alur bisnis yang telah ditetapkan dalam Sistem Informasi E-Procurement PT. Finnet Indonesia.

#### **4.4. Testing**

Tahap *testing* pada penelitian ini dilakukan melalui pengujian aplikasi web untuk memastikan Sistem Informasi E-Procurement PT. Finnet Indonesia dapat berjalan dengan baik pada lingkungan berbasis web. Pengujian difokuskan pada aspek fungsionalitas halaman, alur navigasi, proses input dan pengolahan data, serta kesesuaian tampilan antarmuka pada browser. Selain itu, pengujian juga mencakup pemeriksaan hak akses pengguna, respon sistem terhadap kesalahan input, serta kestabilan sistem saat digunakan dalam skenario operasional. Melalui pengujian web ini, diharapkan sistem mampu memberikan kinerja yang optimal, mudah digunakan, dan siap diterapkan sebagai solusi pendukung proses pengadaan barang dan jasa di PT. Finnet Indonesia.

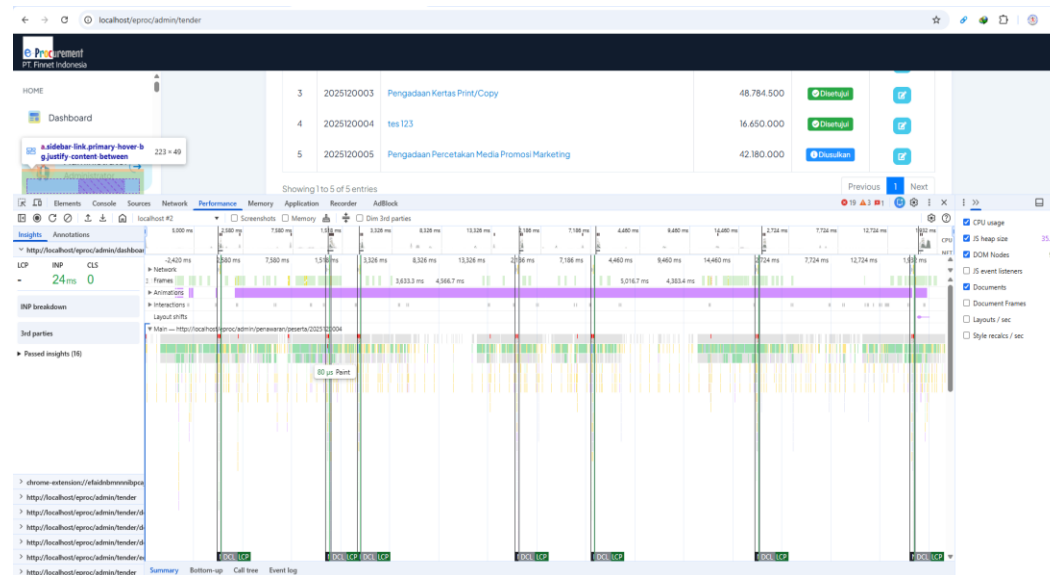
##### **4.4.1. Tahap Pengujian Aplikasi**

###### **a. Pengujian *Performance***

Pengujian performa dilakukan untuk memastikan bahwa Sistem Informasi E-Procurement PT. Finnet Indonesia mampu memberikan respon yang cepat dan stabil dalam mendukung proses pengadaan barang dan jasa. Pengujian ini difokuskan pada akses halaman dan fitur utama sistem, seperti proses login pengguna, pengelolaan data tender, pendaftaran dan pengiriman penawaran oleh vendor, serta penampilan detail paket tender. Pengujian dilakukan secara manual dengan menjalankan aplikasi pada lingkungan pengembangan berbasis web (*localhost*) dan mengamati waktu respon sistem saat memproses permintaan pengguna.

Berdasarkan hasil pengamatan, sistem mampu menampilkan halaman dan memproses data tender serta penawaran dengan waktu respon yang relatif cepat pada skenario penggunaan

normal. Proses penyimpanan dan penampilan data berjalan dengan stabil tanpa ditemukan kendala yang signifikan, sehingga dapat disimpulkan bahwa Sistem Informasi E-Procurement yang dikembangkan telah memenuhi kebutuhan performa dasar sebagai aplikasi web pendukung kegiatan pengadaan di PT. Finnet Indonesia.



Sumber: (Hasil Penelitian, 2025)

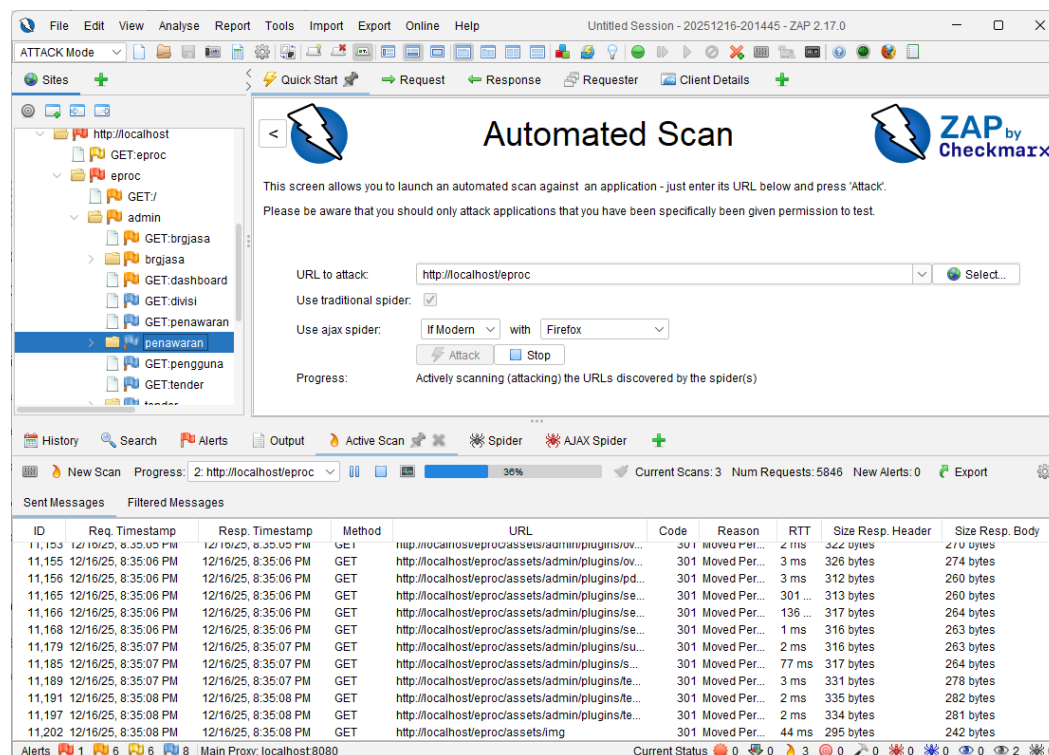
Gambar IV.26. Pengujian Performance menggunakan DevTools

Gambar tersebut menunjukkan hasil pengujian performa aplikasi web Sistem Informasi E-Procurement PT. Finnet Indonesia menggunakan fitur *Performance* pada peramban Google Chrome. Pengujian dilakukan dengan mengakses halaman manajemen tender pada lingkungan pengembangan berbasis *localhost*. Dari hasil pengujian terlihat bahwa halaman dapat dimuat dengan baik, dengan nilai *Largest Contentful Paint* (LCP) sebesar 24 ms, *Interaction to Next Paint* (INP) sebesar 24 ms, serta nilai *Cumulative Layout Shift* (CLS) sebesar 0, yang menunjukkan bahwa tampilan halaman stabil dan tidak mengalami pergeseran elemen saat proses pemuatan berlangsung. Aktivitas pemrosesan halaman didominasi oleh proses *rendering* dan *script execution* dalam durasi yang relatif singkat, sehingga dapat disimpulkan bahwa sistem mampu memberikan respon yang cepat dan stabil pada skenario penggunaan normal. Hasil ini menunjukkan bahwa aplikasi E-

Procurement telah memenuhi aspek performa dasar sebagai aplikasi web pendukung proses pengadaan barang dan jasa.

## b. Pengujian Keamanan Website

Pengujian keamanan dilakukan untuk memastikan Sistem Informasi E-Procurement PT. Finnet Indonesia memiliki tingkat perlindungan yang memadai terhadap potensi ancaman keamanan pada aplikasi berbasis web. Pengujian difokuskan pada pemeriksaan kerentanan umum seperti *SQL Injection*, *Cross-Site Scripting (XSS)*, serta penyalahgunaan input pada formulir sistem. Proses pengujian dilakukan secara manual dengan memanfaatkan fitur *Inspect Element* pada peramban web serta pengujian langsung pada halaman penting, seperti halaman login, pendaftaran vendor, pengelolaan tender, dan pengiriman penawaran.

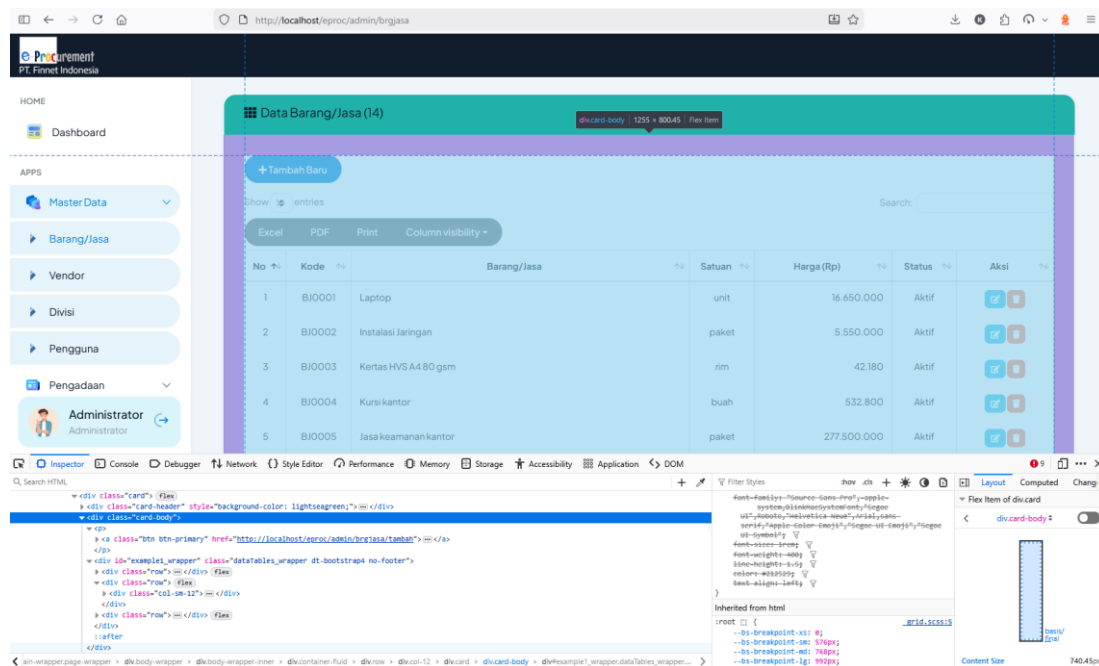


Sumber: (Hasil Penelitian, 2025)

Gambar IV.27. Pengujian Keamananan menggunakan ZAP

Hasil pengujian menunjukkan bahwa sistem telah menerapkan mekanisme keamanan dasar, seperti validasi dan penyaringan data masukan pada sisi server serta penggunaan sesi (*session*) untuk mengatur autentikasi dan hak akses pengguna. Mekanisme ini membantu

mencegah akses tidak sah dan kesalahan input data. Meskipun demikian, sebagai pengembangan lebih lanjut, sistem masih dapat ditingkatkan dengan penambahan perlindungan tambahan, seperti penerapan token *Cross-Site Request Forgery* (CSRF) dan pembatasan akses langsung ke URL tertentu tanpa proses login, guna meningkatkan keamanan Sistem Informasi E-Procurement PT. Finnet Indonesia secara keseluruhan.



Sumber: (Hasil Penelitian, 2025)

Gambar IV.28. Pengujian Keamanan menggunakan *Inspect Element*

#### 4.4.2. Tahap Pengujian Penerimaan Sistem (UAT)

Tahap pengujian penerimaan sistem (*User Acceptance Test/UAT*) bertujuan untuk mengetahui sejauh mana Sistem Informasi *E-Procurement* dapat diterima dan digunakan oleh pengguna utama, yaitu administrator, admin divisi, ppk, dan vendor. Pengujian dilakukan dengan memberikan akses langsung ke sistem yang telah diimplementasikan pada lingkungan pengujian, kemudian pengguna diminta menjalankan skenario proses bisnis yang umum dilakukan dalam *e-procurement*.

Skenario pengujian meliputi pengelolaan data barang/jasa, pengelolaan data vendor, pembuatan dan publikasi tender, proses pemasukan penawaran oleh vendor, evaluasi penawaran oleh panitia, serta pembuatan laporan hasil tender. Seluruh proses tersebut diuji untuk memastikan sistem berjalan sesuai dengan alur pengadaan yang telah ditetapkan.

Setelah melakukan pengujian, pengguna diminta mengisi formulir evaluasi yang mencakup kriteria kemudahan penggunaan sistem, kelengkapan fitur, kejelasan alur proses pengadaan, serta kecepatan dan stabilitas akses sistem. Hasil pengujian menunjukkan bahwa sistem e-procurement secara umum telah memenuhi kebutuhan pengguna dan mendukung proses pengadaan secara lebih terstruktur dan transparan. Meskipun demikian, terdapat beberapa masukan dari pengguna, seperti penambahan fitur filter status tender, peningkatan fungsi pencarian data vendor dan penawaran, serta optimalisasi tampilan informasi pada dashboard. Masukan tersebut menjadi bahan pertimbangan untuk pengembangan dan penyempurnaan sistem pada tahap selanjutnya.

Tabel IV.4. *User Acceptance Testing*

<b>Dokumen <i>User Acceptance Testing</i></b>	
Nama Proyek	: Perancangan Sistem Informasi <i>E-Procurement</i> Berbasis Web Dengan Metode MVC
Studi Kasus/Mitra	: PT. Finnet Indonesia
<b>Proses Pengujian</b>	

No	Use Case	Hasil Uji	Nama Penguji	Tanggal Pengujian	Catatan
1	<p>Use Case uji : Login            Deskripsi : Verifikasi pengguna terdaftar</p> <p>Kasus pengujian:            username : admin            password : admin</p> <p>Hasil yang diharapkan:            - Jika pengguna terverifikasi maka berhasil masuk sistem            - Jika pengguna tidak terverifikasi maka gagal masuk sistem</p>	Berhasil	Rudi  Dedi	12-11-2025  12-11-2025	
2	<p>Use Case uji : Tambah data pengguna            Deskripsi : Menguji validasi <i>input</i></p> <p>Kasus pengujian:            Nama pengguna : admineproc            Email : admineproc@finnet.id            Telepon : 02123456789            Username : admineproc            Password : admin0987</p> <p>Hasil yang diharapkan:            - Jika data valid dan lengkap maka simpan berhasil            - Jika data tidak valid atau tidak lengkap maka simpan gagal</p>	Berhasil	Dedi  Ayu	12-11-2025  12-11-2025	
3	<p>Use Case uji : Daftar Vendor            Deskripsi : Menguji validasi <i>input</i> dan penyimpanan data vendor sekaligus pendaftaran pengguna kategori vendor.</p> <p>Kasus pengujian:            Nama pengguna : Rudi            Email : rudi@finnet.id            Telepon : 02123456789            Username : rudi            Password : rudi0987            Nama vendor : PT. ABC Konoha            NIB : 123456789            Alamat : Jl. Air Mancur</p> <p>Hasil yang diharapkan:            - Jika data valid dan lengkap maka simpan berhasil            - Sistem menyimpan data vendor sekaligus membuat data pengguna baru untuk vendor            - Jika data tidak valid atau tidak lengkap maka simpan gagal</p>	Berhasil	Rudi  Ayu	12-11-2025  12-11-2025	
4	<p>Use Case uji : Menyetujui Tender            Deskripsi : Menguji fungsionalitas sistem ketika ppk menyetujui tender.</p>	Berhasil	Andi  Dedi	12-11-2025  12-11-2025	

	Kasus pengujian: Kode Tender : 2025120001 Nama Paket : Pengadaan Komputer Kerja Sales dan Marketing  Hasil yang diharapkan: - Jika tender disetujui maka sistem otomatis membuat jadwal tender dan membuat sekaligus mempublikasi dokumen pengumuman tender				
5	Use Case uji : Mengirim Penawaran Deskripsi : Menguji fungsionalitas sistem ketika mengirim penawaran tender.  Kasus pengujian: Kode Tender : 2025120003 Nama Paket : Pengadaan Kertas Print/Copy Dokumen : dokumen.pdf Ttl Penawaran : 47.730.000  Hasil yang diharapkan: - Jika data valid dan lengkap maka simpan berhasil - Jika data tidak valid atau tidak lengkap maka simpan gagal	Berhasil	Hanif  Sulaiman	12-11-2025  12-11-2025	

Sumber: (Hasil Penelitian, 2025)

#### 4.5. Support

Setelah Sistem Informasi E-Procurement diimplementasikan dan digunakan oleh mitra, sistem tersebut memerlukan dukungan (support) dan pemeliharaan secara berkelanjutan. Hal ini dikarenakan perangkat lunak yang telah diterapkan tidak terlepas dari kemungkinan terjadinya perubahan, baik akibat ditemukannya kesalahan (*bug*), perubahan lingkungan eksternal seperti pembaruan sistem operasi dan infrastruktur teknologi informasi, maupun adanya kebutuhan penambahan atau penyempurnaan fitur sesuai dengan kebijakan dan proses pengadaan yang berlaku.

Dukungan sistem dilakukan dengan cara monitoring kinerja aplikasi, perbaikan kesalahan yang ditemukan selama operasional, serta penyesuaian sistem terhadap perubahan regulasi pengadaan barang dan jasa. Selain itu, dilakukan pula pemeliharaan adaptif dan

penyempurnaan fungsional, seperti peningkatan fitur pengelolaan tender, penawaran vendor, dan pelaporan, guna memastikan sistem tetap relevan dan optimal dalam mendukung proses e-procurement.

#### 4.5.1. Publikasi Web

Setelah Sistem Informasi E-Procurement selesai dibangun dan melalui tahap pengujian, langkah selanjutnya adalah melakukan publikasi sistem ke server hosting agar dapat digunakan secara operasional. Sistem dipublikasikan menggunakan layanan server hosting atau Virtual Private Server (VPS) yang telah ditentukan oleh pihak mitra, serta menggunakan domain resmi agar dapat diakses secara luas dan aman oleh pengguna yang berwenang.

Proses deployment dilakukan dengan memindahkan seluruh file aplikasi dan basis data dari lingkungan pengembangan (localhost) ke server produksi. Pengunggahan file aplikasi dilakukan melalui layanan FTP atau file manager hosting, sedangkan proses impor basis data dilakukan menggunakan tools manajemen basis data seperti phpMyAdmin.

Adapun tahapan publikasi sistem yang dilakukan adalah:

1. Ekspor basis data dari server lokal.
2. Mengunggah seluruh file aplikasi berbasis web ke server hosting.
3. Melakukan konfigurasi untuk menyesuaikan pengaturan koneksi basis data dan lingkungan server.
4. Melakukan pengujian akses sistem melalui URL publik untuk memastikan seluruh fitur *e-procurement* berjalan dengan baik.

Dengan dilakukannya proses publikasi web ini, Sistem Informasi E-Procurement dapat diakses melalui browser, baik menggunakan perangkat desktop maupun perangkat mobile yang terhubung ke jaringan internet.

#### 4.5.2. Spesifikasi Hardware dan Software

Untuk dapat berjalan dengan baik, sistem informasi *e-procurement* ini membutuhkan perangkat pendukung dengan kinerja optimal. Spesifikasi dibawah ini adalah kapasitas minimum yang dibutuhkan:

##### a. Spesifikasi *Hardware*

Monitor	:	14 inch HD
Prosesor	:	Core i3 atau setara
Memory	:	4 GB RAM
Penyimpanan	:	125 GB HDD/SSD
Printer	:	Printer A4

##### b. Spesifikasi *Software*

Sistem Operasi	:	Windows 8
Web Server	:	Laragon/XAMPP
Bahasa Pemrograman	:	PHP 8.1
Database	:	MySQL
Browser	:	Chrome/Firefox/Edge

#### 4.6. Spesifikasi Dokumen Sistem Usulan

Dokumen sistem usulan mencakup seluruh formulir dan dokumen elektronik yang digunakan dalam sistem informasi *e-procurement* untuk mendukung proses pengadaan barang dan jasa. Dokumen-dokumen tersebut memiliki peran penting dalam setiap tahapan pengadaan, mulai dari pendaftaran vendor, pengajuan usulan tender, pengelolaan dokumen tender, hingga proses penyampaian penawaran oleh vendor.

Spesifikasi dokumen sistem usulan disusun untuk memberikan gambaran yang terstruktur mengenai jenis, fungsi, serta alur penggunaan dokumen dalam sistem. Dokumen formulir pendaftaran vendor digunakan sebagai sarana input data identitas dan legalitas

perusahaan penyedia. Dokumen usulan tender berfungsi untuk mencatat kebutuhan pengadaan yang diajukan oleh unit atau divisi terkait. Selanjutnya, dokumen tender digunakan sebagai media penyampaian informasi resmi pengadaan, sedangkan dokumen penawaran menjadi sarana bagi vendor untuk mengajukan harga dan persyaratan sesuai dengan ketentuan tender.

Setiap dokumen dalam sistem usulan dirancang dalam format digital berbasis web dan telah disesuaikan dengan kebutuhan proses e-procurement. Dokumen-dokumen tersebut dilengkapi dengan mekanisme validasi data untuk meminimalkan kesalahan input serta memastikan kelengkapan informasi. Seluruh dokumen diimplementasikan dalam antarmuka web berbasis framework CodeIgniter dan terintegrasi langsung dengan basis data, sehingga mendukung pengelolaan data yang efisien, terpusat, dan terdokumentasi dengan baik.

Berikut adalah spesifikasi dokumen sistem usulan:

1. Nama Dokumen : Formulir Pendaftaran Vendor  
Fungsi : Untuk mencatat data identitas dan legalitas vendor sebagai penyedia barang/jasa  
Sumber : Vendor  
Tujuan : *Database*  
Media : Tampilan Web  
Frekuensi : Setiap ada vendor yang melakukan pendaftaran  
Format : Lampiran B1
2. Nama Dokumen : Formulir Data Vendor  
Fungsi : Untuk mencatat data vendor baru oleh Administrator  
Sumber : Administrator  
Tujuan : *Database*  
Media : Tampilan Web  
Frekuensi : Setiap ada vendor baru

- Format : Lampiran B2
3. Nama Dokumen : Formulir Data Barang/Jasa
- Fungsi : Untuk mencatat data barang/jasa baru
- Sumber : Administrator
- Tujuan : *Database*
- Media : Tampilan Web
- Frekuensi : Setiap ada barang/jasa baru
- Format : Lampiran B3
4. Nama Dokumen : Formulir Data Pengguna
- Fungsi : Untuk mencatat data pengguna baru
- Sumber : Administrator
- Tujuan : *Database*
- Media : Tampilan Web
- Frekuensi : Setiap ada pengguna baru
- Format : Lampiran B4
5. Nama Dokumen : Formulir Usulan Tender
- Fungsi : Untuk mencatat usulan kebutuhan pengadaan barang/jasa
- Sumber : Admin Divisi
- Tujuan : *Database*
- Media : Tampilan Web
- Frekuensi : Setiap ada pengajuan usulan tender
- Format : Lampiran B5
6. Nama Dokumen : Formulir Kirim Penawaran
- Fungsi : Untuk mencatat data penawaran baru
- Sumber : Vendor

- Tujuan : *Database*
- Media : Tampilan Web
- Frekuensi : Setiap vendor mengirim penawaran
- Format : Lampiran B6
7. Nama Dokumen : Dokumen Pengumuman Tender
- Fungsi : Sebagai pengumuman tender resmi
- Sumber : Sistem
- Tujuan : Publik
- Media : File PDF
- Frekuensi : Setiap PPK menyetujui tender baru
- Format : Lampiran B7
8. Nama Dokumen : Dokumen Pengumuman Pemenang Tender
- Fungsi : Sebagai pengumuman pemenang tender resmi
- Sumber : Sistem
- Tujuan : Publik
- Media : File PDF
- Frekuensi : Setiap Administrator menetapkan pemenang tender
- Format : Lampiran B8
9. Nama Dokumen : Laporan
- Fungsi : Sebagai laporan data *e-procurement*
- Sumber : Administrator
- Tujuan : Pimpinan
- Media : File PDF
- Frekuensi : Sesuai kebutuhan
- Format:       Lampiran B9