

Aplikasi Penagihan (Invoice)
Studi Kasus : Unit Bisnis Penunjang Infrastruktur
PT. PLN Batam

TUGAS AKHIR

Oleh :

Brian Crida Hyan Dejong **3311001059**

Dicky Pratam **3311001042**

Disusun untuk memenuhi syarat kelulusan Program Diploma III



PROGRAM STUDI TEKNIK INFORMATIKA

POLITEKNIK NEGERI BATAM

BATAM

2013

LEMBAR PENGESAHAN

Batam, 26 Juli 2013

Pembimbing,

Andy Triwinarko, MT

NIK. 105038

LEMBAR PERYATAAN

Dengan ini, saya:

NIM : 3311001059

Nama : Brian Crida Hyan DeJong

adalah mahasiswa Teknik Informatika Politeknik Batam yang menyatakan bahwa tugas akhir dengan judul:

Aplikasi Penagihan (Invoice)

(Studi Kasus : Unit Bisnis Penunjang Infrastruktur PT. PLN Batam)

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya orang lain
2. tidak melakukan pemalsuan data
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa ijin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Akhir ini.

Batam, 26 Juli 2013

Brian Crida Hyan Dejong
3311001059

LEMBAR PERYATAAN

Dengan ini, saya:

NIM : 3311001042

Nama : Dicky Pratama

adalah mahasiswa Teknik Informatika Politeknik Batam yang menyatakan bahwa tugas akhir dengan judul:

Aplikasi Penagihan (Invoice)

(Studi Kasus : Unit Bisnis Penunjang Infrastruktur PT. PLN Batam)

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya orang lain
2. tidak melakukan pemalsuan data
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa ijin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Akhir ini.

Batam, 26 Juli 2013

Dicky Pratama
3311001042

KATA PENGANTAR

Dengan mengucapkan puji syukur kehadirat Tuhan Yang Maha Esa yang telah memberikan rahmat, hidayah dan karuniaNya sehingga kami dapat menyelesaikan laporan Tugas Akhir ini.

laporan ini kami susun untuk menyelesaikan masa pendidikan pada jurusan Teknik Informatika di Politeknik Negeri Batam. Dalam penyusunan laporan ini kami menyadari banyak sekali kekurangan, namun berhubung banyaknya pihak luar yang mendukung dan turut membantu, sehingga laporan ini dapat diselesaikan. Oleh karena itu dalam kesempatan ini kami sangat bersyukur dan berterima kasih kepada semua pihak yang telah membantu kelancaran laporan ini.

Laporan ini dapat terselesaikan dengan adanya bantuan dari pihak pembimbing materi maupun teknis, oleh karena itu kami mengucapkan terimakasih kepada :

1. Tuhan Yang Maha Esa sehingga kami dapat menyelesaikan Laporan Tugas Akhir ini.
2. Orang Tua dan Keluarga kami yang telah memberikan dorongan baik moral maupun material.
3. Bapak Andy Triwinarko, MT selaku dosen pembimbing.
4. Bapak Ari Wibowo, MT selaku dosen pengampu.
5. Teman-teman satu kelas yang selalu mendukung.
6. Pihak-pihak lain yang banyak membantu dan memberikan dukungan.

Batam, Juli 2013

Penulis

DAFTAR ISI

Bab I	Pendahuluan.....	1
I.1	Latar Belakang	1
I.2	Rumusan Masalah	2
I.3	Batasan Masalah.....	3
I.4	Tujuan Penelitian	3
Bab II	Tinjauan Pustaka.....	4
II.1	Unit Bisnis Penunjang Infrastruktur PT. PLN Batam.....	4
II.2	Sistem Informasi	4
II.3	HTML.....	7
II.4	PHP.....	9
II.5	MySQL	10
II.6	Apache.....	13
II.7	CSS.....	14
II.8	Ajax	16
II.9	Jquery	17
II.10	FPDF	19
Bab III	Analisis dan Perancangan	21
III.1	Deskripsi Umum Aplikasi / Sistem	21
III.2	Komponen-komponen kebutuhan sistem.....	22
III.3	Diagram Use Case	23
III.4	Skenario Use Case	23
III.4.1	Kelola Produk.....	23
III.4.1.1	Add Product.....	23

III.4.1.2 Update Product	24
III.4.1.3 Delete Product	24
III.4.2 Kelola Invoice	24
III.4.2.1 Add Invoice	24
III.4.2.2 View Invoice	25
III.4.2.3 Delete Invoice.....	25
III.4.2.4 Cetak Invoice.....	25
III.4.3 Kelola Payment	26
III.4.3.1 View Payment.....	26
III.4.3.2 Pay (Status).....	26
III.4.3.3 Cetak Kuitansi	26
III.4.3.4 Delete Payment.....	26
III.4.4 Kelola Taxes.....	27
III.4.4.1 Add Taxes.....	27
III.4.4.2 Delete Taxes	27
III.4.5 Setting	27
III.5 Robustness Diagram	28
III.5.1 Kelola Produk.....	28
III.5.2 Kelola Invoice	28
III.5.3 Kelola Payment	29
III.5.4 Kelola Taxes.....	29
III.5.5 Setting	30
III.6 Sequence Diagram.....	30
III.6.1 Kelola Produk.....	30
III.6.2 Kelola Invoice	32

III.6.2.1 Add Invoice	32
III.6.2.2 Delete Invoice.....	32
III.6.2.3 Cetak Invoice.....	33
III.6.3 Kelola Payment	33
III.6.3.1 Pay.....	33
III.6.3.2 Delete Payment.....	34
III.6.3.3 Cetak Kuitansi	34
III.6.4 Kelola Kelola Taxes.....	35
III.6.4.1 Add Tax.....	35
III.6.4.2 Update Tax	35
III.6.4.3 Delete Tax	36
III.6.5 Setting	36
III.7 Diagram Kelas	37
III.8 Diagram Kelas Rinci.....	38
III.8.1 Kelas Gui Produk.....	38
III.8.2 Kelas Gui Invoice	38
III.8.3 Kelas Gui Payment	38
III.8.4 Kelas Gui Tax.....	38
III.8.5 Kelas Gui Pengaturan	38
III.8.6 Kelas Kelola Produk.....	39
III.8.7 Kelas Kelola Invoice.....	39
III.8.8 Kelas Kelola Payment.....	40
III.8.9 Kelas Kelola Tax	40
III.8.10 Kelas Setting	40
III.9 Algoritma	40

III.9.1	Kelola Produk.....	40
III.9.2	Kelola Invoice	42
III.9.3	Kelola Payment	44
III.9.4	Kelola Taxes.....	46
III.9.5	Setting	48
III.10	Rancangan Antarmuka	49
Bab IV	Implementasi dan Pengujian	56
IV.1	Implementasi Kelas PHP	56
IV.2	Implementasi Antarmuka PHP	57
IV.2.1	Implementasi Login.....	57
IV.2.2	Implementasi Home.....	58
IV.2.3	Implementasi Create New Invoice	58
IV.2.4	Implementasi Produk.....	60
IV.2.5	Implementasi View Invoice	60
IV.2.6	Implementasi Payment	61
IV.2.7	Implementasi Taxes.....	61
IV.2.8	Implementasi Setting	62
IV.3	Pengujian.....	63
IV.3.1	Skenario Pengujian.....	63
IV.3.2	Hasil Rincian Pengujian	65
Bab V	Kesimpulan dan Saran	67
V.1.	Kesimpulan.....	67
V.2.	Saran	67
Bab VI	Daftar Pustaka	68

Daftar Gambar

Gambar 1 Deskripsi Umum Aplikasi/Sistem.....	21
Gambar 2 Diagram Use Case	23
Gambar 3 Robustness Diagram Kelola Produk.....	28
Gambar 4 Robustness Diagram Kelola Invoice	28
Gambar 5 Robustness Diagram Kelola Payment	29
Gambar 6 Robustness Diagram Kelola Tax.....	29
Gambar 7 Robustness Diagram Pengaturan.....	30
Gambar 8 Sequende Diagram Add Produk.....	30
Gambar 9 Sequence Diagram Update Produk.....	31
Gambar 10 Sequence Diagram Delete Produk.....	31
Gambar 11 Sequence Diagram Add Invoice.....	32
Gambar 12 Sequence Diagram Delete Invoice	32
Gambar 13 Sequence Diagram Cetak Invoice.....	33
Gambar 14 Sequence Diagram Pay	33
Gambar 15 Sequence Diagram Delete Payment.....	34
Gambar 16 Sequence Diagram Cetak Kuitansi.....	34
Gambar 17 Sequence Diagram Add Tax	35
Gambar 18 Sequence Diagram Update Tax.....	35
Gambar 19 Sequence Diagram Delete Tax.....	36
Gambar 20 Sequence Diagram Pengaturan.....	36
Gambar 21 Diagram Kelas.....	37
Gambar 22 Form Login.....	49
Gambar 23 Halaman Utama.....	50
Gambar 24 Create New Invoice (Setup Client).....	51

Gambar 25 Create New Invoice (Add Product)	52
Gambar 26 Produk	52
Gambar 27 All Invoice.....	53
Gambar 28 Payment.....	53
Gambar 29 Taxes	54
Gambar 30 Setting	55
Gambar 31 Antarmuka Login.....	57
Gambar 32 Antarmuka Home	58
Gambar 33 Antarmuka Setup Client.....	58
Gambar 34 Antarmuka Add Produk	59
Gambar 35 Antarmuka Save/Export Invoice	59
Gambar 36 Antarmuka Produk.....	60
Gambar 37 Antarmuka View Invoice	60
Gambar 38 Antarmuka Payment	61
Gambar 39 Antarmuka Setting	63
Gambar 40 Skenario Pengujian	64

DAFTAR TABEL

Tabel 1 Implentasi Kelas PHP.....	56
Tabel 2 Implementasi Antarmuka PHP	57
Tabel 3 Hasil Rincian Pengujian	66

Bab I Pendahuluan

I.1 Latar Belakang

Dewasa ini perkembangan teknologi informasi sudah sedemikian pesat. Perkembangan yang pesat tidak hanya teknologi perangkat keras dan perangkat lunak saja, tetapi metode komputasi juga ikut berkembang, baik dari aspek administratif maupun teknologi mengharuskan agar proses pelayanan dapat berlangsung secara efektif dan efisien. Untuk mewujudkan hal tersebut dibutuhkan beberapa fasilitas pendukung, dimana salah satu fasilitas pendukung tersebut adalah aplikasi teknologi informasi dalam bidang sistem informasi perusahaan.

Penggunaan sistem informasi menjanjikan suatu proses yang lebih efisien daripada proses konvensional. Perusahaan-perusahaan saat ini masih banyak yang menggunakan cara manual untuk mengolah data dengan menggunakan buku tulis yang tentunya sangat tidak efektif dan efisien. Digitalisasi dapat meningkatkan kinerja perusahaan dalam melakukan pengolahan data.

Unit Bisnis Penunjang Infrastruktur adalah salah satu divisi dari PT. PLN Batam yang bergerak dibidang pengembangan infrastruktur milik PT. PLN Batam untuk bisnis telematika. Unit Bisnis Penunjang Infrastruktur sampai saat ini dalam melakukan pengolahan data seperti mengelola data- data mengenai perusahaan rekanan dan mengolah data tagihan masih dilakukan secara manual.

Sistem yang ada sekarang ini belum memberikan akses yang optimal terhadap seluruh civitas Unit Bisnis Penunjang Infrastruktur PT. PLN Batam. Sehingga sudah seharusnya Unit Bisnis Penunjang Infrastruktur PT PLN Batam dapat mengembangkan diri dengan membuat suatu sistem informasi yang dapat melakukan pengolahan data yang cepat, efisien dan akurat kepada pemakainya.

Oleh karena itu, Unit Bisnis Penunjang Infrastruktur PT. PLN Batam memerlukan sebuah sistem pengolah data tagihan (*invoice*) yang dapat mengolah berbagai data tagihan yang ada.

Sistem Informasi pengelola data tagihan (*invoice*) adalah sebuah perangkat lunak berbasis website yang bermanfaat untuk membantu pengelola infrastruktur dari Unit Bisnis Penunjang Infrastruktur dalam melaksanakan tugasnya pembuatan surat penagihan terhadap rekanan bisnis yang bekerja sama dengan Unit Bisnis Penunjang Infrastruktur yang melakukan penyewaan inventaris milik PT. PLN Batam, seperti penyewaan tiang buat perusahaan TV kabel dan Fiber Optic. Sistem Informasi pengelola data tagihan (*invoice*) ini merupakan salah satu alternatif pemberian layanan dalam pengolahan data tagihan yang bisa memberikan informasi yang akurat dalam waktu yang singkat dan dapat diakses kapan saja.

Melalui pengembangan sistem informasi pengelola data tagihan (*invoice*) ini diharapkan dapat memberi layanan yang optimal kepada seluruh civitas Unit Bisnis Penunjang Infrastruktur PT. PLN Batam dan diharapkan memberikan kemudahan bagi Unit Bisnis Penunjang Infrastruktur dalam melakukan pekerjaan sehingga proses pengelolaan data lebih efektif dan efisien.

I.2 Rumusan Masalah

Beberapa rumusan masalah yang akan dibahas dalam Tugas Akhir ini adalah sebagai berikut:

- 1) Bagaimana membangun suatu sistem yang dapat mengoptimalkan Unit Bisnis Penunjang Infrastruktur melakukan pengolahan data tagihan kepada perusahaan rekanan secara mudah, efektif, dan efisien ?
- 2) Apakah sistem dapat diimplementasikan dan memenuhi kebutuhan dari Unit Bisnis Penunjang Infrastruktur dalam pengolahan data tagihan kepada perusahaan rekanan ?

I.3 Batasan Masalah

Batasan masalah yang akan dibahas dalam Tugas Akhir ini adalah :

1. Ruang lingkup pengimplementasian aplikasi invoice pada lingkup Unit Bisnis Penunjang Infrastruktur PT. PLN Batam .
2. Tidak membahas mengenai aplikasi data rekanan yang merupakan aplikasi yang digunakan untuk mengelola data-data perusahaan rekanan (*client*).

I.4 Tujuan Penelitian

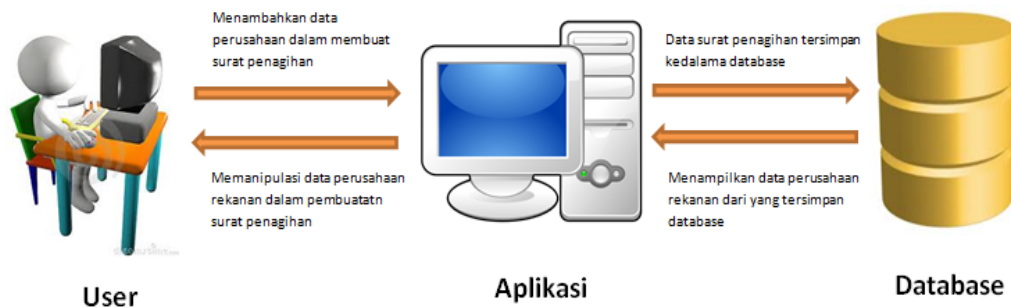
- 1) Untuk membangun sistem yang dapat mengoptimalkan pengolahan berbagai data tagihan kepada perusahaan rekanan Unit Bisnis Penunjang Infrastruktur secara mudah, efektif, dan efisien.
- 2) Sistem dapat diimplementasikan dan memenuhi kebutuhan dari Unit Bisnis Penunjang Infrastruktur dalam pengolahan data tagihan kepada perusahaan rekanan.

Bab III Analisis dan Perancangan

Pada bab ini akan menjelaskan tentang analisis dan perancangan aplikasi penagihan atau invoice dengan studi kasus Unit Bisnis Penunjang Infrastruktur PT. PLN Batam.

III.1 Deskripsi Umum Aplikasi / Sistem

Dalam penelitian ini analisis yang dilakukan adalah suatu pemahaman dimana aplikasi ini dapat digunakan untuk memanipulasi data perusahaan rekanan dan surat penagihan yang ditujukan kepada perusahaan rekanan tersebut. Analisis yang dilakukan berupa deskripsi umum sistem dan analisis kebutuhan sistem.



Gambar 1 Deskripsi Umum Aplikasi/Sistem

Sistem dalam aplikasi ini digunakan untuk membuat surat penagihan (*invoice*) yang ditujukan untuk perusahaan rekanan. Pada aplikasi ini hanya memiliki satu aktor saja, yaitu USER sebagai pengelola dari surat penagihan.

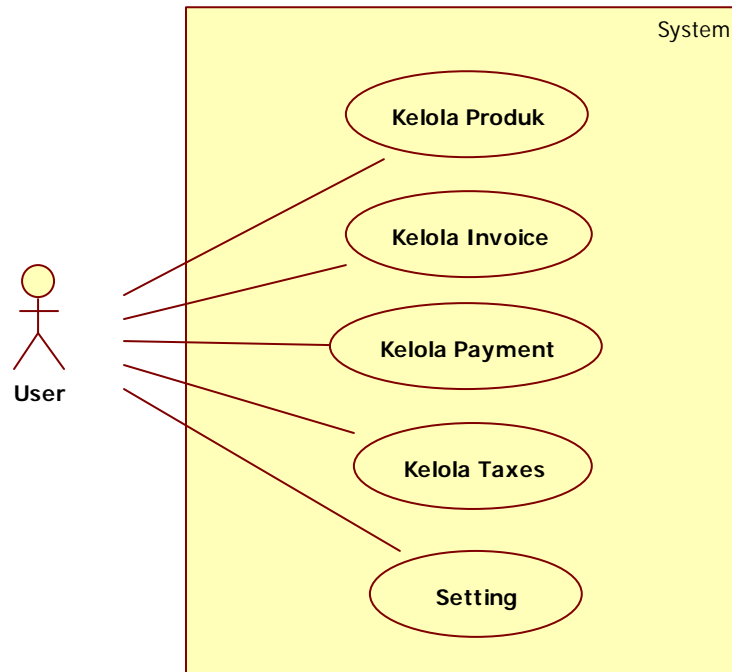
III.2 Komponen-komponen kebutuhan sistem

Komponen-komponen yang sangat berperan dalam menunjang peranan sistem yang dirancang terhadap pengolahan data. Beberapa komponen yang dibutuhkan sistem untuk dapat beroperasi dengan baik, antara lain:

1. Hardware, merupakan komponen yang sangat penting dibutuhkan dalam mewujudkan sistem yang diusulkan. Dalam hal ini penulis merinci spesifikasi komponen hardware yaitu:
 - a. Laptop
 - b. Hardisk 320 GB
 - c. Memori 2 GB
 - d. Video Card 512 MB

2. Software yang digunakan pembuatan aplikasi ini sebagai berikut :
 - a. Sistem Operasi Windows 7
 - b. Adobe DreamWeaver CS5
 - c. Adobe Photoshop CS5
 - d. Notepad++
 - e. Mozilla Firefox
 - f. MySQL

III.3 Diagram Use Case



Gambar 2 Diagram Use Case

III.4 Skenario Use Case

III.4.1 Kelola Produk

III.4.1.1 Add Product

Kondisi Awal : Sistem menampilkan menu add product, data produk belum ada.

Skenario : User mengisi data produk seperti nama, deskripsi, jumlah, dan harga dan user menekan tombol submit.

Kondisi Akhir : Data produk telah ditambahkan.

III.4.1.2 Update Product

Kondisi Awal : sistem menampilkan menu product, data produk ditampilkan. User menekan tombol edit.

Skenario : User mengubah data product dan user menekan tombol submit.

Kondisi Akhir : Data produk berhasil diubah.

III.4.1.3 Delete Product

Kondisi Awal : Sistem menampilkan daftar produk.

Skenario : User memilih produk dan menekan tombol hapus.

Kondisi Akhir : Data produk telah dihapus.

III.4.2 Kelola Invoice

III.4.2.1 Add Invoice

Kondisi Awal : Data invoice belum ada

Skenario : User memilih client, menambah item, mengisi jumlah item dan menekan tombol save/export.
Sistem menyimpan data invoice.

Kondisi Akhir : Data invoice telah ditambahkan.

Alternate Course

Client belum dipilih : Tampil pesan error, data tidak bisa disimpan.

Item belum dipilih : Data tidak bisa disimpan

III.4.2.2 View Invoice

Kondisi Awal : Menampilkan halaman utama.

Skenario : User memilih menu View Invoice.

Kondisi Akhir : Sistem Menampilkan daftar invoice beserta pilhan
untuk mengubah, mengunduh dan menghapus
invoice.

III.4.2.3 Delete Invoice

Kondisi Awal : Sistem menampilkan halaman view invoice.

Skenario : User memilih invoice dan menekan tombol hapus
invoice.

Kondisi Akhir : Data Invoice yang dipilih telah dihapus.

III.4.2.4 Cetak Invoice

Kondisi Awal : Menampilkan halaman View Invoice.

Skenario : User memilih invoice dan menekan tombol
download invoice.

Kondisi Akhir : Sistem mengambil data invoice dan menyimpan ke
direktori dalam format PDF.

III.4.3 Kelola Payment

III.4.3.1 View Payment

- Kondisi Awal : Menampilkan halaman utama.
- Skenario : User memilih menu payment.
- Kondisi Akhir : Sistem menampilkan status dari daftar invoice.

III.4.3.2 Pay (Status)

- Kondisi Awal : Menampilkan menu payment, Status pembayaran belum diubah
- Skenario : User menekan tombol pembayaran.
- Kondisi Akhir : Status pembayaran telah diubah.

III.4.3.3 Cetak Kuitansi

- Kondisi Awal : Menampilkan halaman Payment dan Nama client yang telah melakukan pembayaran.
- Skenario : User menekan tombol cetak kuitansi dan memasukkan no. invoice sebagai inputan data client kemudian user menekan tombol print.
- Kondisi Akhir : Sistem mencetak kuitansi dan menyimpannya dalam bentuk PDF

III.4.3.4 Delete Payment

- Kondisi Awal : Menampilkan menu pembayaran dan daftar status

pembayaran.

Skenario : Sistem menampilkan daftar payment, user memilih salah satu payment dan menekan tombol hapus.

Kondisi Akhir : Data payment telah dihapus.

III.4.4 Kelola Taxes

III.4.4.1 Add Taxes

Kondisi Awal : Sistem menampilkan menu Taxes.

Skenario : User mengisi nama tax dan value tax, user menekan tombol simpan.

Kondisi Akhir : Data tax telah tersimpan.

III.4.4.2 Delete Taxes

Kondisi Awal : Sistem menampilkan data tax.

Skenario : User memilih data tax, dan user menekan tombol hapus.

Kondisi Akhir : Data tax telah dihapus.

III.4.5 Setting

Kondisi Awal : Menampilkan menu Setting.

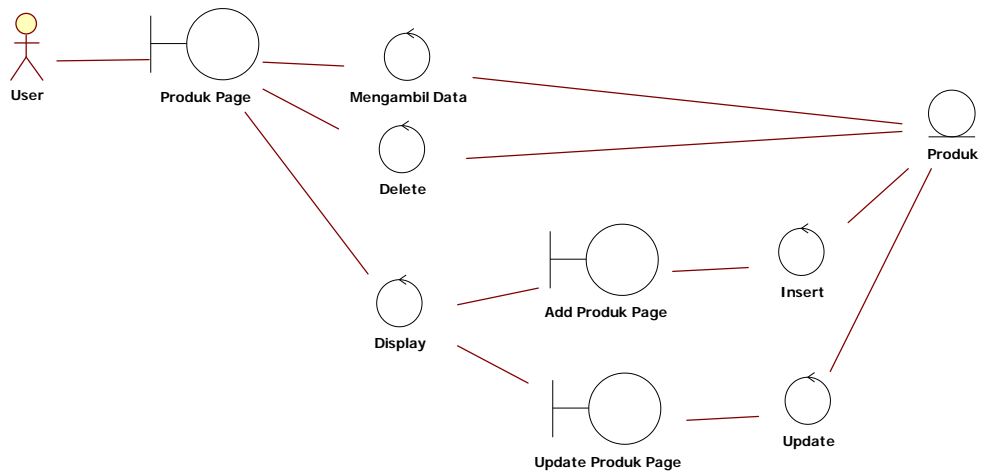
Skenario : User memasukkan data perusahaan, mengupload logo perusahaan, dan dapat mengubah password. Sistem menyimpan data perusahaan

dan password yang telah diubah

Kondisi Akhir : Data perusahaan telah ditambahkan dan password telah diubah.

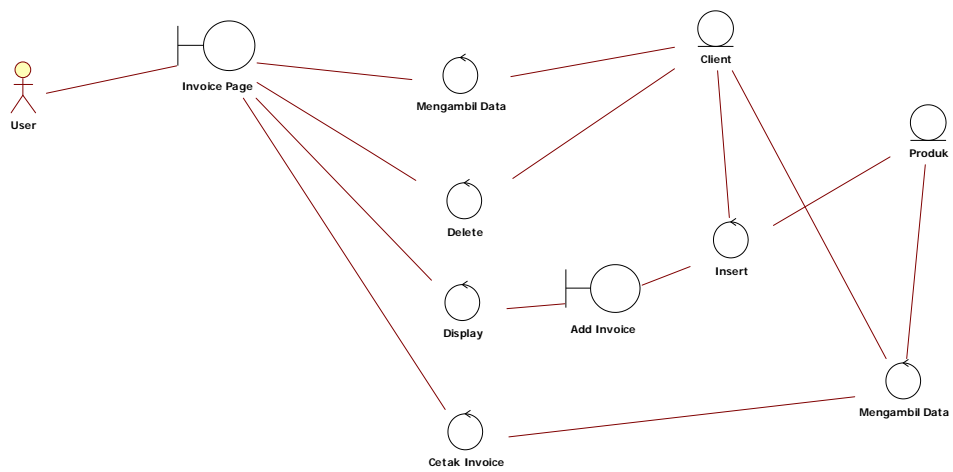
III.5 Robustness Diagram

III.5.1 Kelola Produk



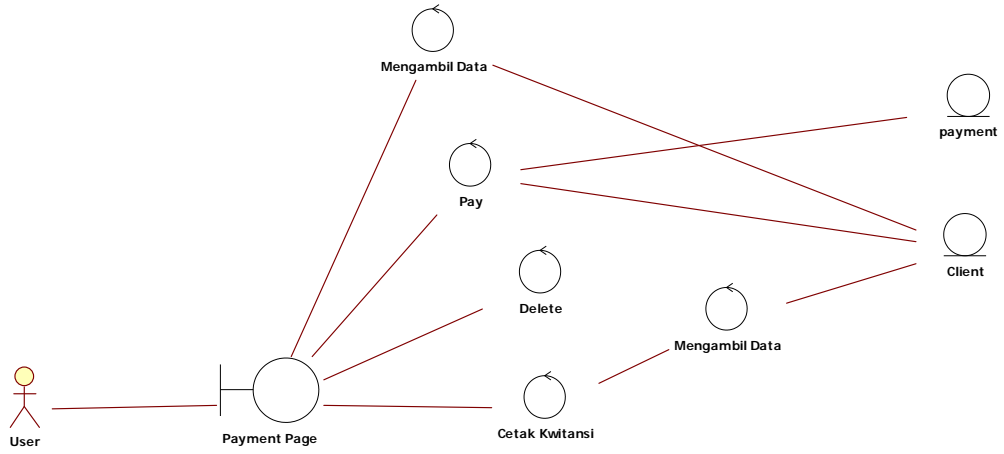
Gambar 3 Robustness Diagram Kelola Produk

III.5.2 Kelola Invoice



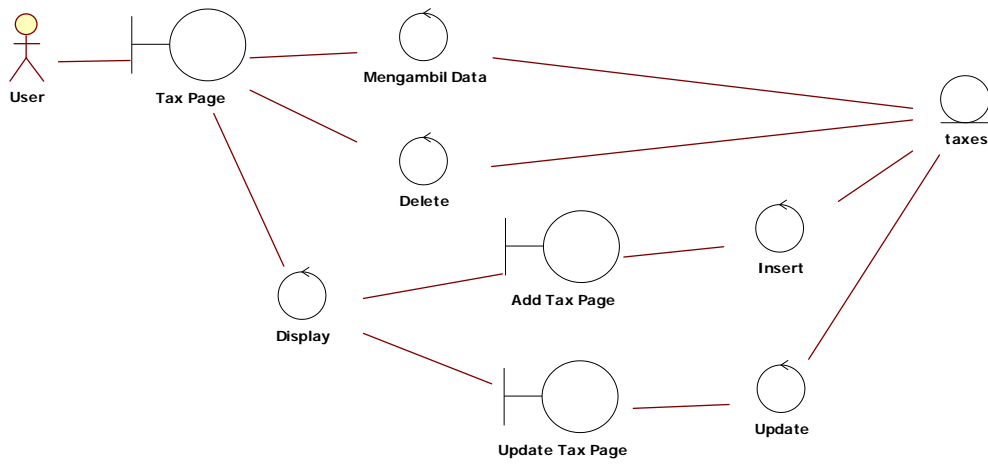
Gambar 4 Robustness Diagram Kelola Invoice

III.5.3 Kelola Payment



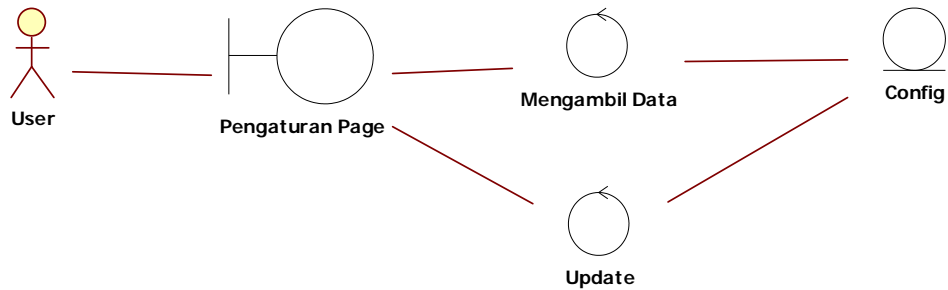
Gambar 5 Robustness Diagram Kelola Payment

III.5.4 Kelola Taxes



Gambar 6 Robustness Diagram Kelola Tax

III.5.5 Setting

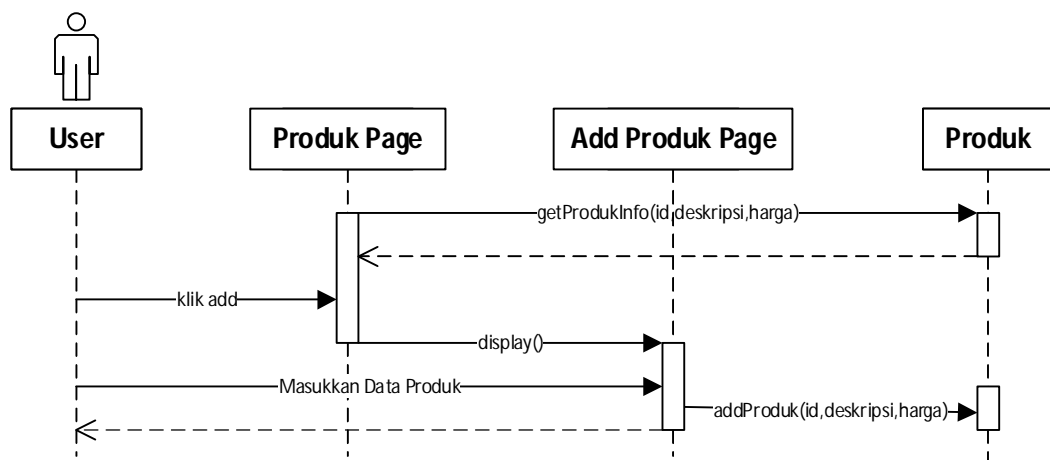


Gambar 7 Robustness Diagram Pengaturan

III.6 Sequence Diagram

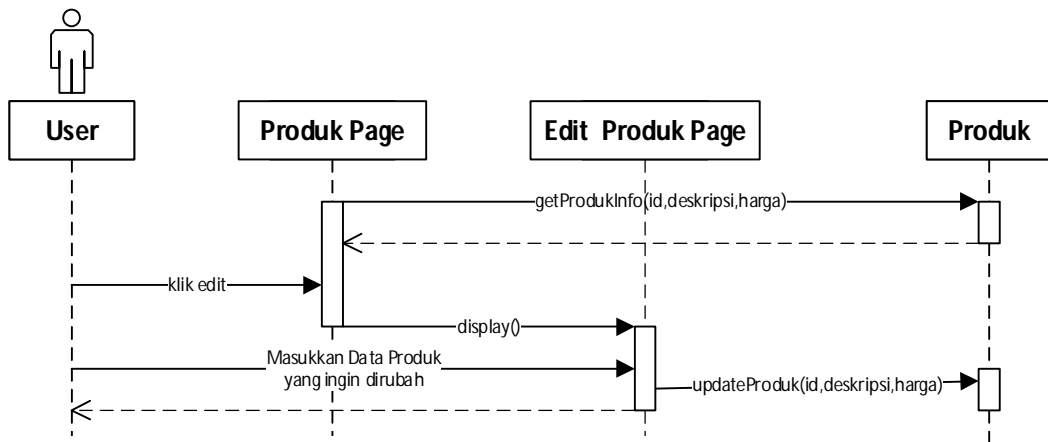
III.6.1 Kelola Produk

III.6.1.1 Add Produk



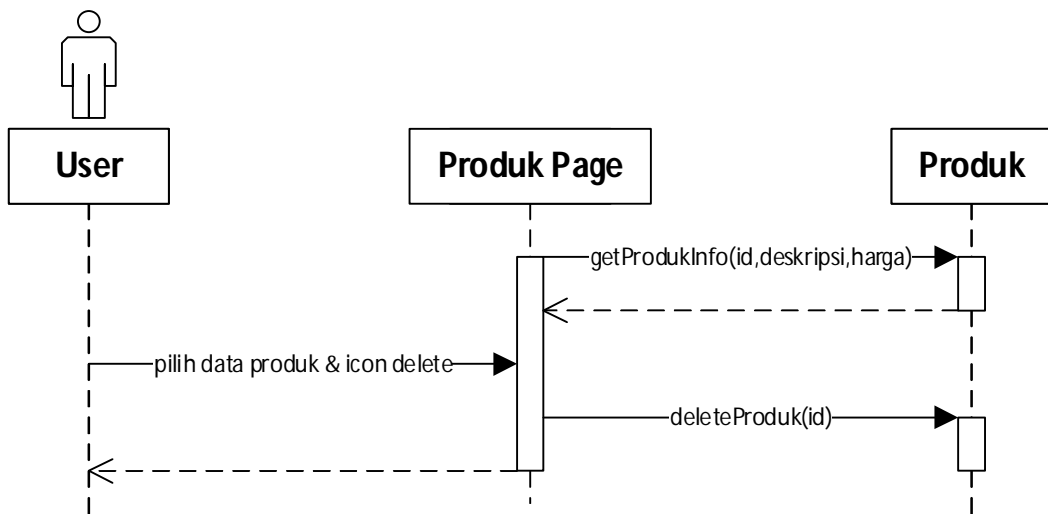
Gambar 8 Sequende Diagram Add Produk

III.6.1.2 Update Produk



Gambar 9 Sequence Diagram Update Produk

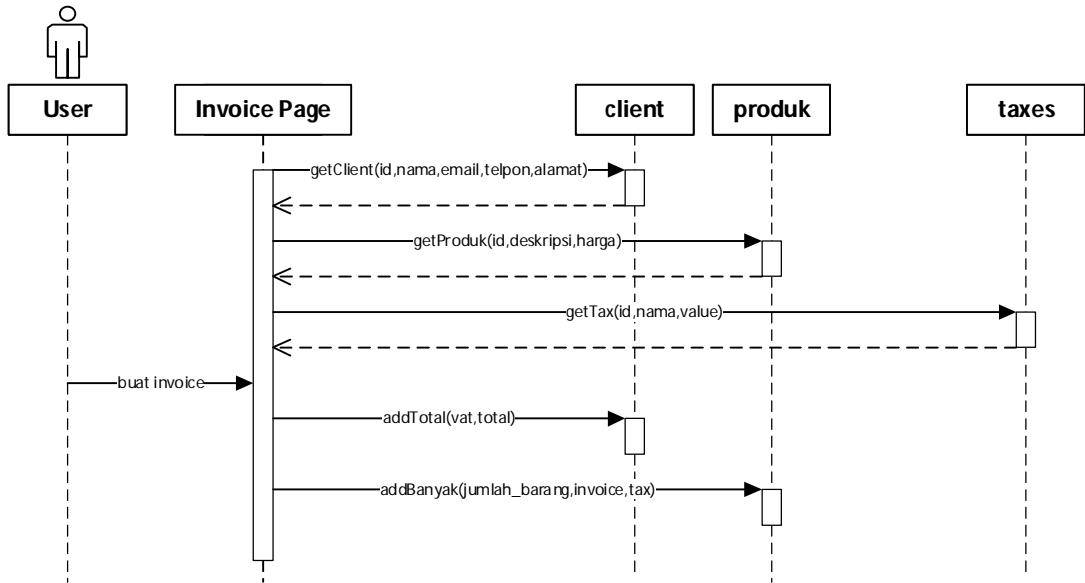
III.6.1.3 Delete Produk



Gambar 10 Sequence Diagram Delete Produk

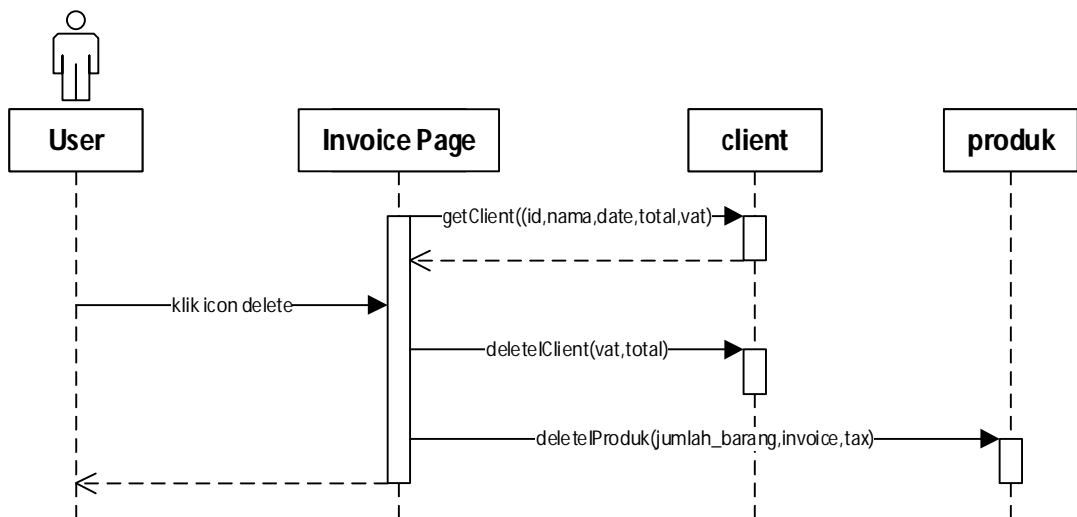
III.6.2 Kelola Invoice

III.6.2.1 Add Invoice



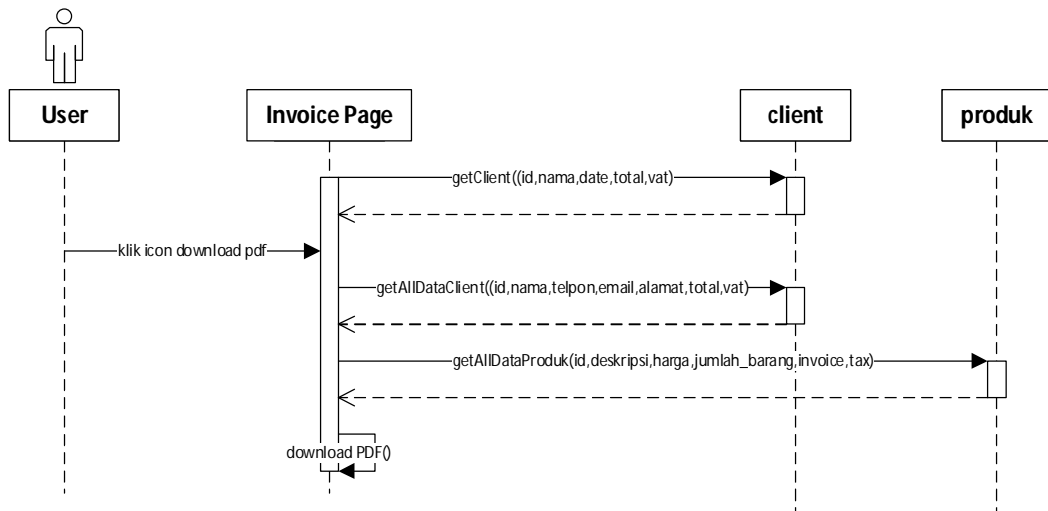
Gambar 11 Sequence Diagram Add Invoice

III.6.2.2 Delete Invoice



Gambar 12 Sequence Diagram Delete Invoice

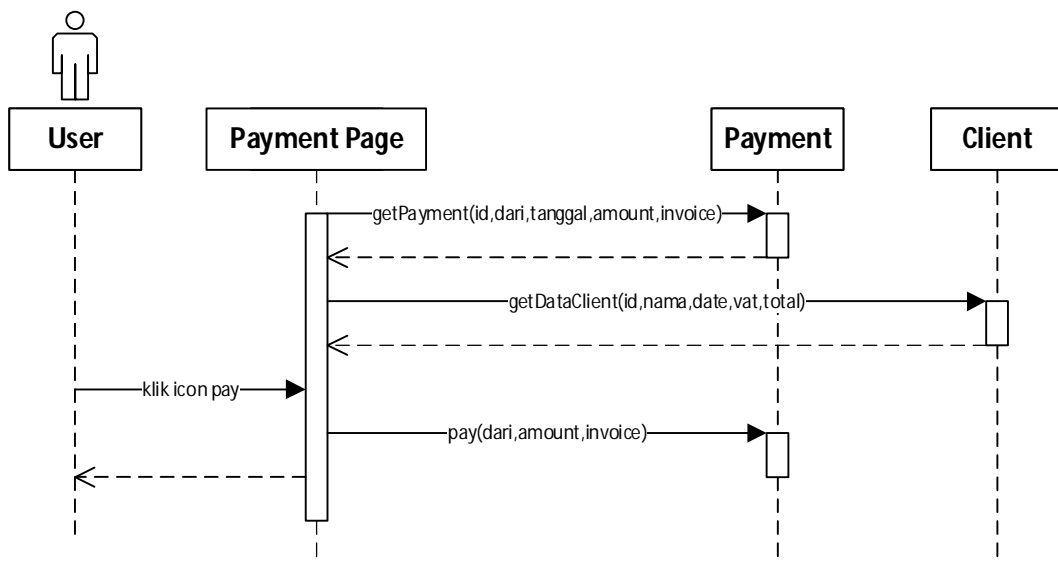
III.6.2.3 Cetak Invoice



Gambar 13 Sequence Diagram Cetak Invoice

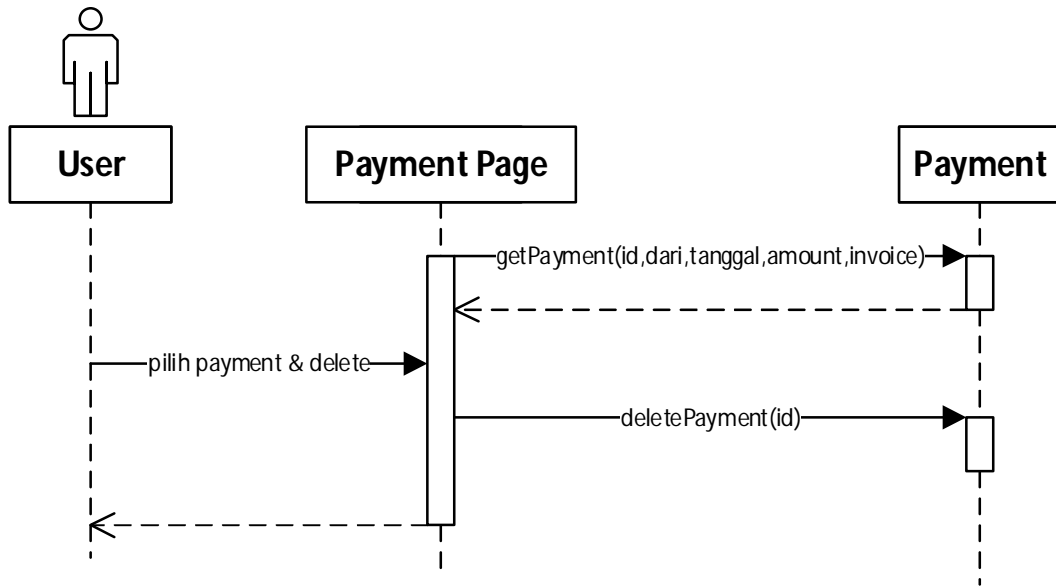
III.6.3 Kelola Payment

III.6.3.1 Pay



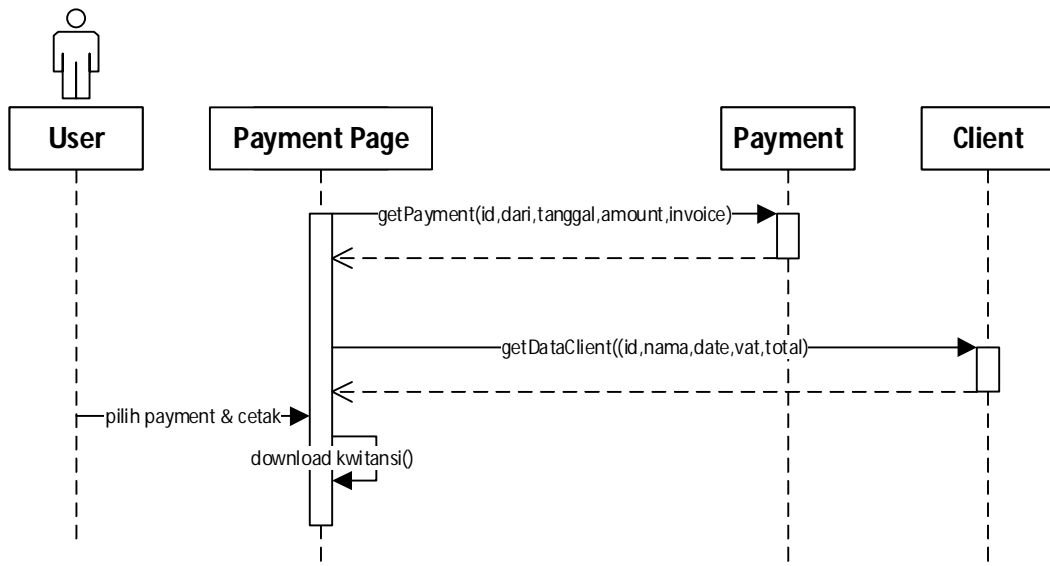
Gambar 14 Sequence Diagram Pay

III.6.3.2 Delete Payment



Gambar 15 Sequence Diagram Delete Payment

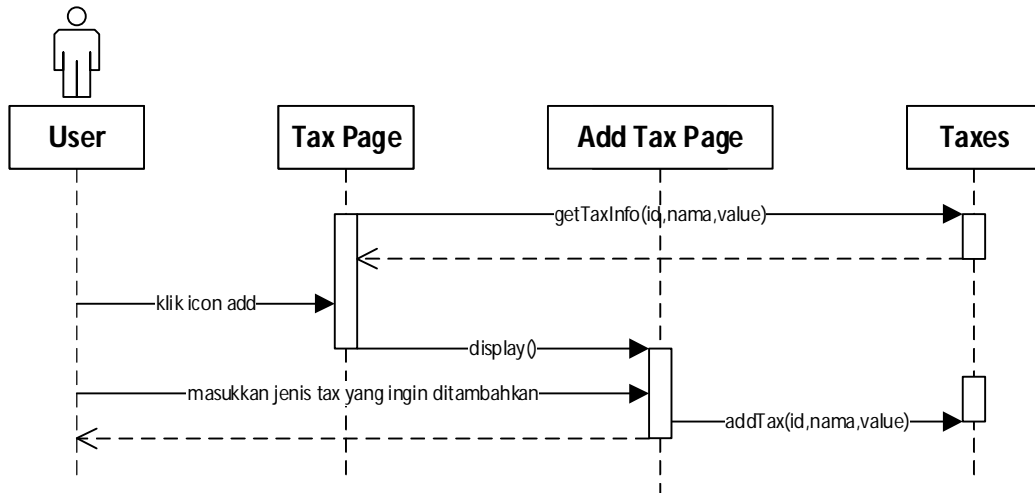
III.6.3.3 Cetak Kuitansi



Gambar 16 Sequence Diagram Cetak Kuitansi

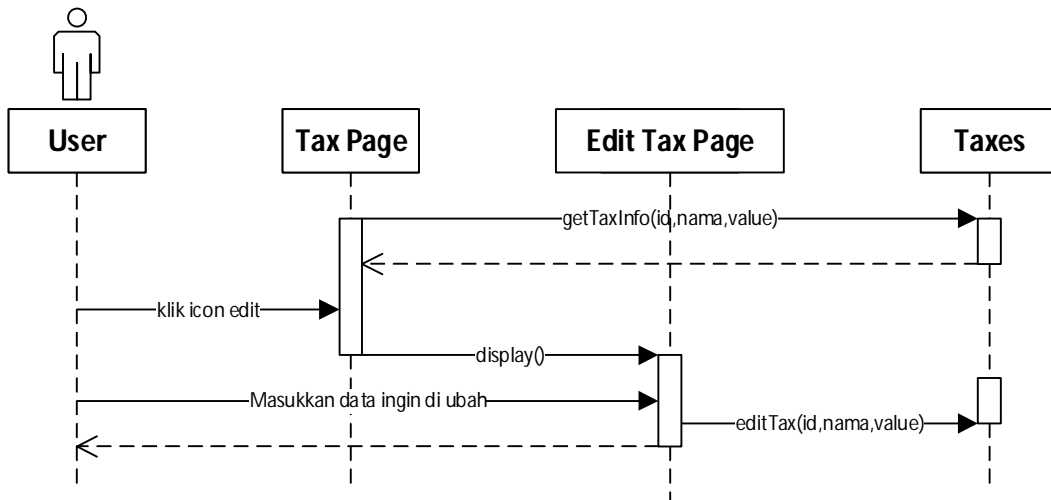
III.6.4 Kelola Kelola Taxes

III.6.4.1 Add Tax



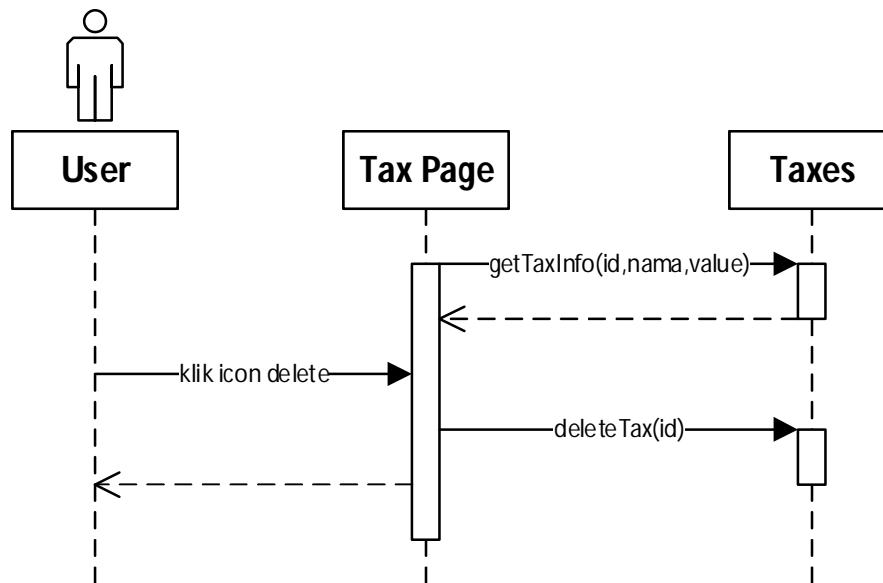
Gambar 17 Sequence Diagram Add Tax

III.6.4.2 Update Tax



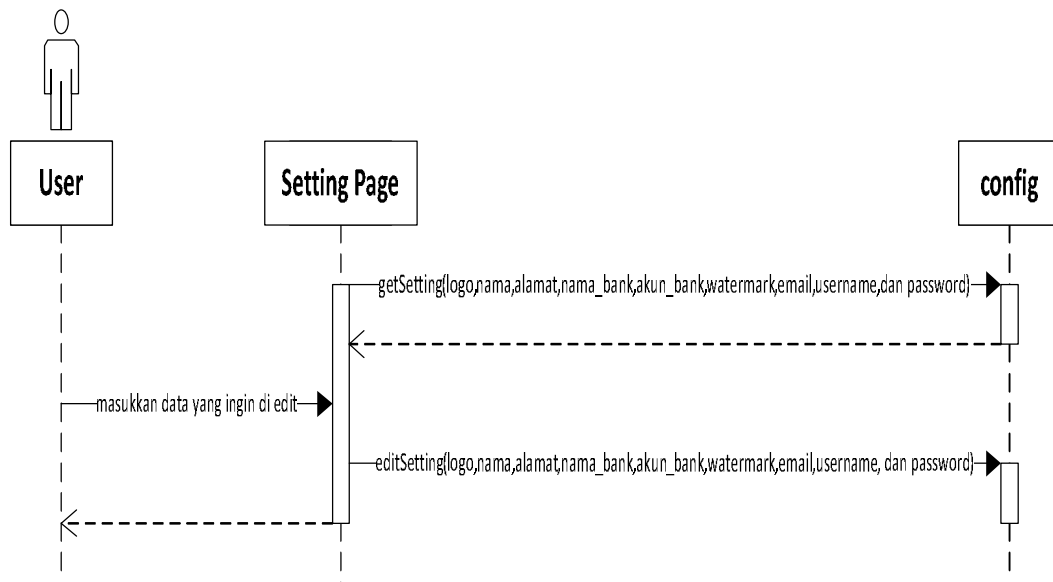
Gambar 18 Sequence Diagram Update Tax

III.6.4.3 Delete Tax



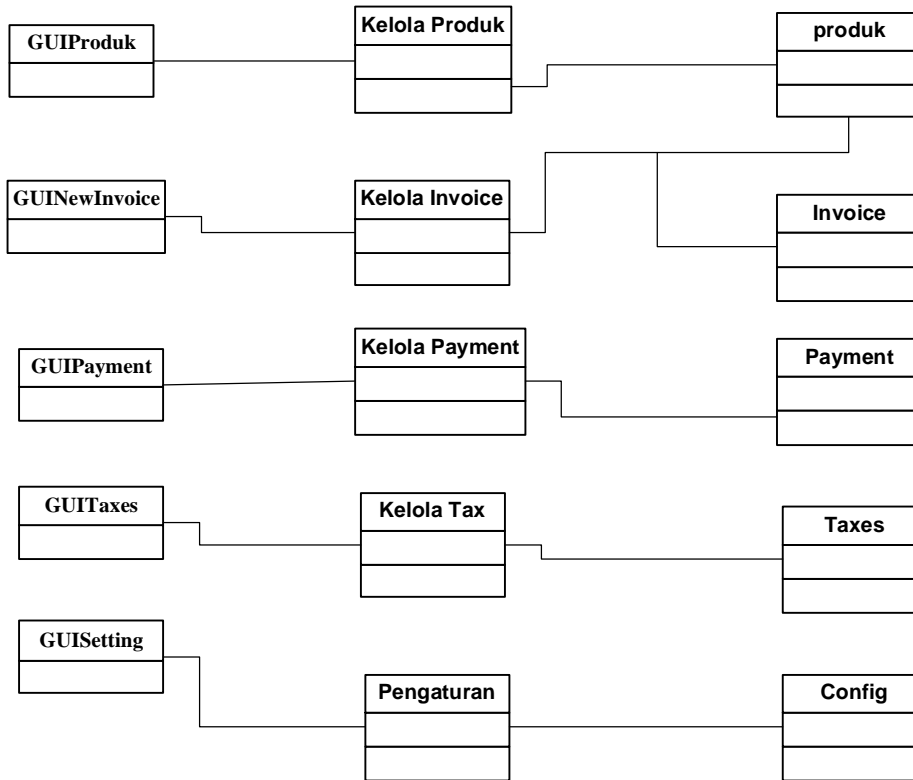
Gambar 19 Sequence Diagram Delete Tax

III.6.5 Setting



Gambar 20 Sequence Diagram Pengaturan

III.7 Diagram Kelas



Gambar 21 Diagram Kelas

III.8 Diagram Kelas Rinci

III.8.1 Kelas Gui Produk

GuiProduk
+ display()

III.8.2 Kelas Gui Invoice

GuiInvoice
+ display()

III.8.3 Kelas Gui Payment

GuiPayment
+ display()

III.8.4 Kelas Gui Tax

GuiTax
+ display()

III.8.5 Kelas Gui Pengaturan

GuiPengaturan
+ display()

III.8.6 Kelas Kelola Produk

KelolaProduk
+ getProdukInfo() + addProduk() + editProduk() + deleteProduk()

III.8.7 Kelas Kelola Invoice

KelolaInvoice
+ getClient() + getProduk() + getTax() + addTotal() + addBanyak() + deleteIClient() + deleteIProduk() + getAllDataClient() + getAllDataProduk() + downloadPDF()

III.8.8 Kelas Kelola Payment

KelolaPayment
+ getPayment() + getDataClient() + pay() + deletePayment() + DownloadKuitansi() + upload()

III.8.9 Kelas Kelola Tax

KelolaTax
+ getTaxInfot() + addTax() + editTax() + deletetax()

III.8.10 Kelas Setting

Setting
+ getSetting() + editSetting()

III.9 Algoritma

III.9.1 Kelola Produk

Nama Operasi : getProductInfo()

Deskripsi : Menampilkan data produk ke halaman produk

Initial state : data ada di database

Final state : data ditampilkan kehalaman produk

Algoritma

eksekusi query “SELECT * FROM produk order by id desc”;

write(produk)

Data produk ditampilkan dihalaman produk

Nama Operasi : AddProduk()

Deskripsi : Menambahkan data produk kedalam database

Initial state : data belum ada di database

Final state : data ditambahkan kedalam database

Algoritma

Read(produk)

Read(harga)

Eksekusi query “INSERT INTO produk (namaProduk,harga) VALUES (\$produk,\$harga) ”; Data produk ditambah sesuai dengan inputan user

Nama Operasi : editProduk()

Deskripsi : Merubah data produk yang ada didalam database

Initial state : data belum ada di database

Final state : data ditambahkan kedalam database

Algoritma

Inisialisai (id)

Read(deskirpsi)

Read(harga)

Eksekusi query “UPDATE produk SET namaProduk = '\$namaProduk', harga = '\$harga' WHERE id = '\$id'”; Data produk diubah sesuai dengan inputan user

Nama Operasi : deleteProduk()

Deskripsi : Menghapus data produk dari database

Initial state : data ada di database

Final state : data dihapus dari database

Algoritma

Inisialisai (id)

Read(deskirpsi)

Read(harga)

Eksekusi query “DELETE FROM produk WHERE id=\$_GET['delete']”: Data produk dihapus

III.9.2 Kelola Invoice

Nama Operasi : getClient(), getProduk(), getTax, getAllDataIClient() & getAllDataProduk()

Deskripsi : Menampilkan data invoice

Initial state : data ada di database

Final state : data ditampilkan kehalaman invoice

Algoritma

eksekusi query “SELECT * FROM client ORDER BY id”;

write(client)

THEN

eksekusi query “SELECT * FROM products WHERE invoice=".\$row['id']."";”;

write(produk)

THEN

eksekusi query “SELECT SUM(amount) AS total FROM payments WHERE invoice=".\$row['id']."";”;

write(amount)

Nama Operasi : addTotal() & addBanyak()

Deskripsi : Membuat Invoice

Initial state : data belum ada di database

Final state : data ditambahkan kedalam database

Algoritma

Input (nama client)

Input (Produk)

Input (Tax)

Input (total)

Input (Banyak)

Ambil no invoice dari database

Nama Operasi : deleteIClient() & deleteIProduk()

Deskripsi : Manghapus Invoice

Initial state : data ada di database

Final state : data dihapus dari database

Algoritma

delete (nama client)

delete (IProduk)

delete (IClient)

Nama Operasi : downloadPDF()

Deskripsi : Mencetak Invoice

Initial state : data ada di database

Final state : data dicetak dalam bentuk PDF

Algoritma

Ambil data client dari database.

Ambil data produk dari database.

Cetak invoice ke PDF

III.9.3 Kelola Payment

Nama Operasi : getPayment() dan getDataClient()

Deskripsi : Menampilkan data Payment

Initial state : data ada di database

Final state : data ditampilkan kehalaman Payment

Algoritma

```
do query "SELECT * FROM payments ORDER BY id desc";
write(payment)
THEN
do query "SELECT * FROM client WHERE id='".$row['invoice']."'";
write(client)
```

Nama Operasi : pay()

Deskripsi : Mengubah status pembayaran

Initial state : status pembayaran UNPAID

Final state : status pembayaran berubah menjadi PAID

Algoritma

Inisialisasi invoice

READ(amount)

```
do query "SELECT SUM(amount) as total FROM payments WHERE
invoice='".$row['invoice']."'";
```

Nama Operasi : deletePayment()

Deskripsi : menghapus data payment

Initial state : data ada didatabase

Final state : data dihapus dari database

Algoritma

Inisialisasi id

```
eksekusi query "delete from payments where id='".$_$_GET['delete'].'"
```

Nama Operasi : cetakKuitansi()

Deskripsi : mencetak Kuitansi dalam bentuk PDF

Initial state : data ada didatabase

Final state : data dicetak kedalam bentuk PDF

Algoritma

Ambil data dari payment;

Ambil data dari client;

Cetak Kuitansi

III.9.4 Kelola Taxes

Nama Operasi : getTAXInfo()

Deskripsi : Menampilkan data tax

Initial state : data ada di database

Final state : data ditampilkan kehalaman tax

Algoritma

```
do query "SELECT * FROM taxes WHERE hidden='0' ORDER BY id DESC  
LIMIT 0,10";
```

```
write(taxes)
```

Nama Operasi : addTax()

Deskripsi : menambahkan data tax

Initial state : data belum ada didatabase

Final state : data ada didatabase

Algoritma

Read (nama)

Read (value)

Read (default)

```
Eksekusi query INSERT INTO taxes(`name`,`value`,`default`)  
VALUES(".$_POST['name'].",".$_POST['value'].",".$_POST['default'].");
```

Nama Operasi : editTax()

Deskripsi : merubah data tax

Initial state : data ada didatabase

Final state : data yang didatabase berubah

Algoritma

Read (nama)

Read (value)

```
Eksekusi query UPDATE taxes SET  
`name`=".$_POST['name'].","`value`=".$_POST['value'].","`default`=".$_POST['  
default'].'" WHERE id=".$_POST['id'].";
```

Nama Operasi : deleteTax()

Deskripsi : menghapus data tax

Initial state : data ada didatabase

Final state : data dihapus dari database

Algoritma

Inisialisasi id

```
eksekusi query "UPDATE taxes SET hidden='1' WHERE id='".$_POST['id'].'";
```

III.9.5 Setting

Nama Operasi : getSetting()

Deskripsi : menampilkan data pengaturan

Initial state : data ada didatabase

Final state : data ditampilkan dihalamn pengaturan

Algoritma

```
Eksekusi query SELECT * FROM config;
```

```
Write(config);
```

Nama Operasi : editSetting()

Deskripsi : merubah data pengaturan

Initial state : data ada didatabase

Final state : data pengaturan di ubah

Algoritma

Read(logo)

Read(nama)

Read(alamat)

Read(nma_bank)

Read(akun_bank)

Read(watermark)

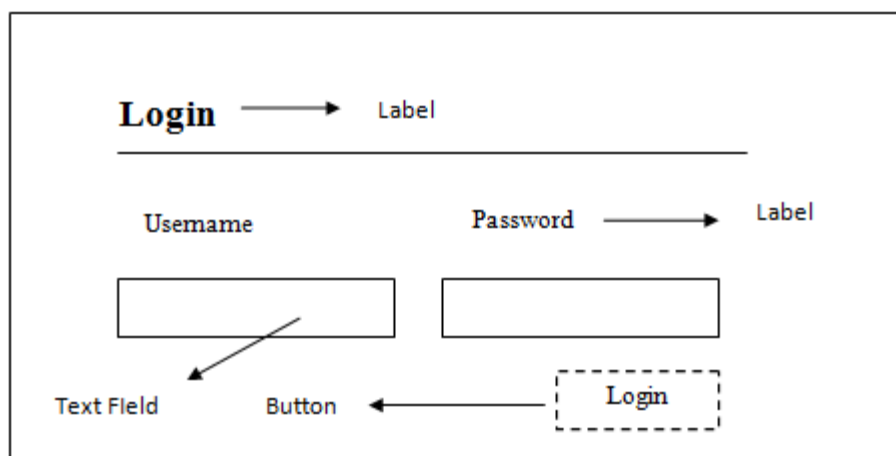
Read(email)

Read(Username)

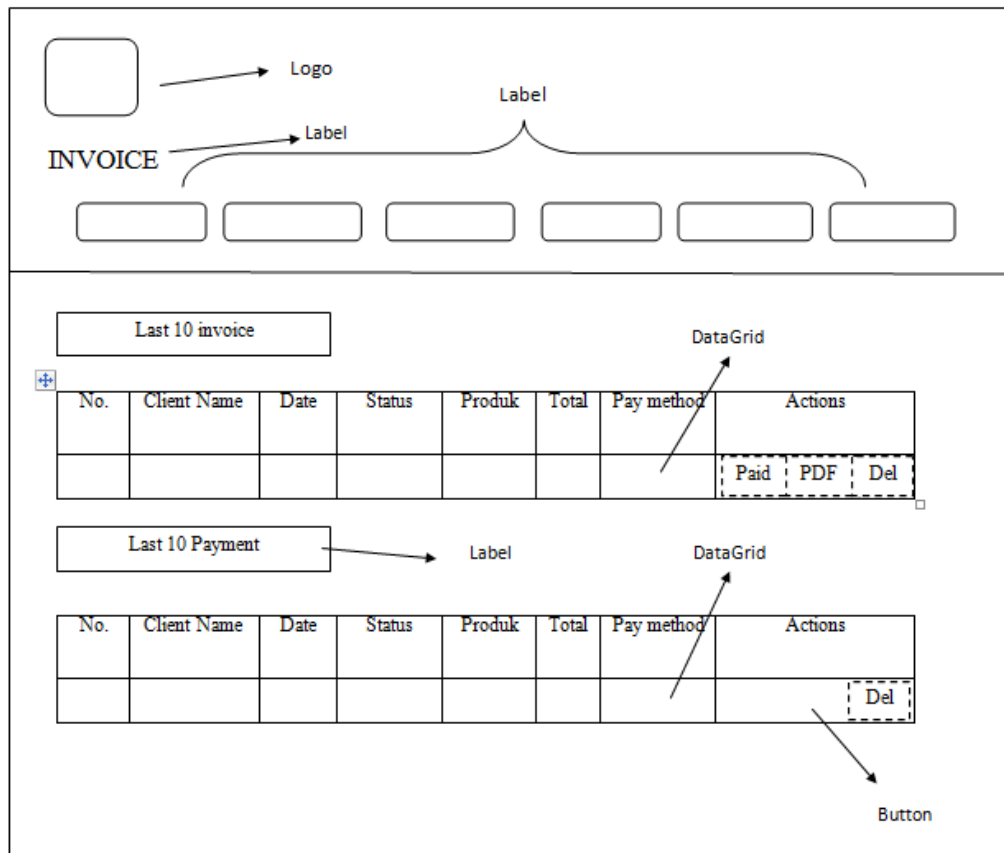
Read>Password)

Eksekusi query UPDATE config SET logo = '\$logo', alamat='\$alamat', nama_bank='\$nama_bank', akun_bank='\$akun_bank', watermark='\$watermark', email='\$email',username='\$username,password='\$password'WHERE nama='\$nama';

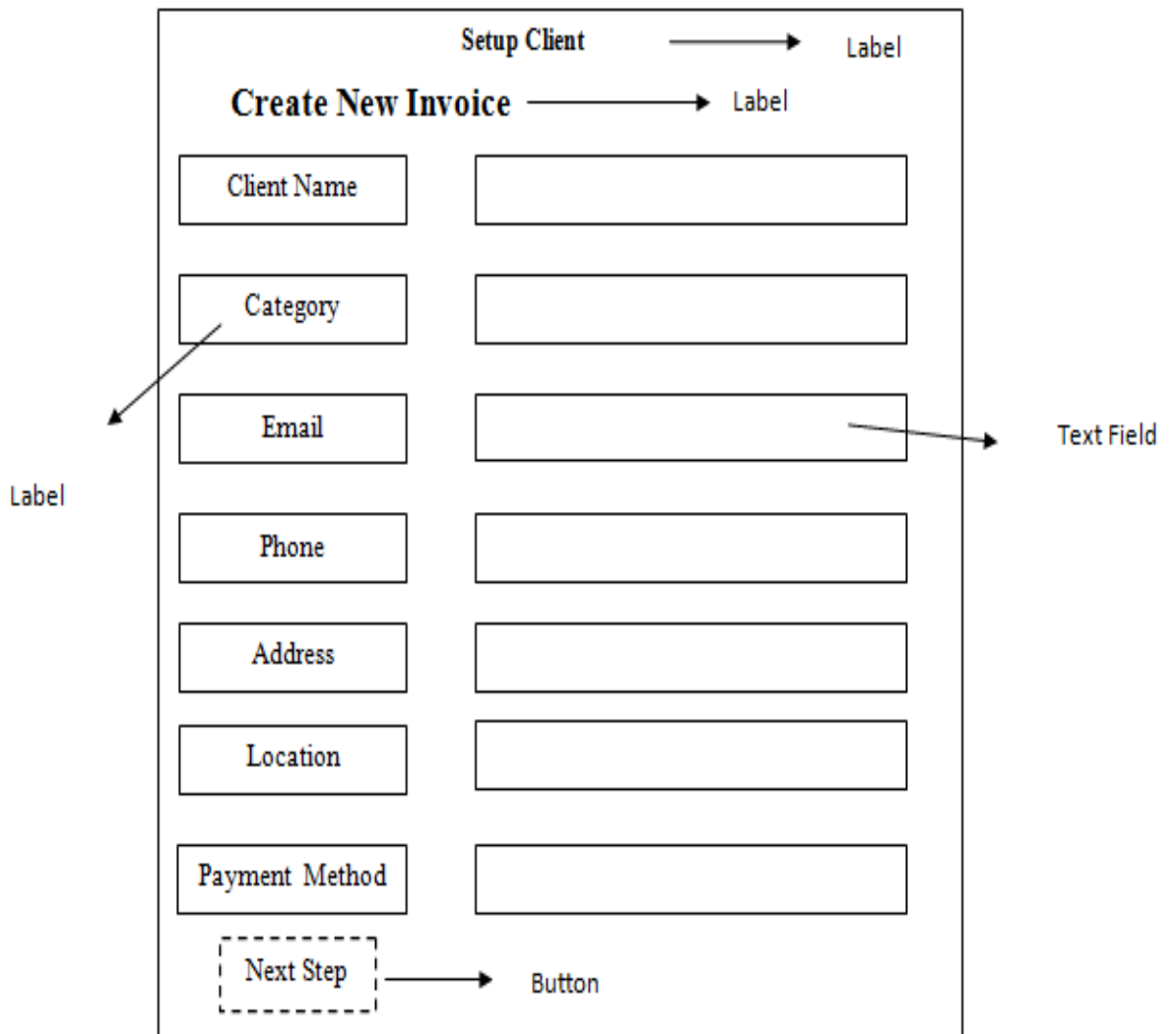
III.10 Rancangan Antarmuka



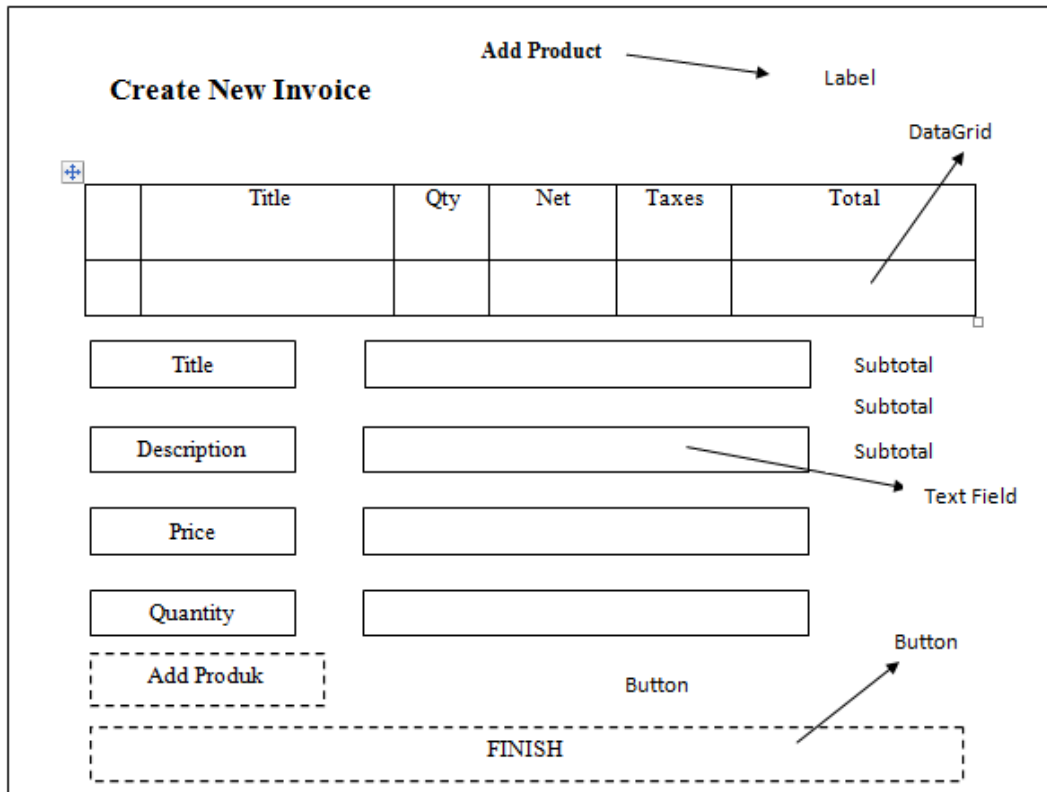
Gambar 22 Form Login



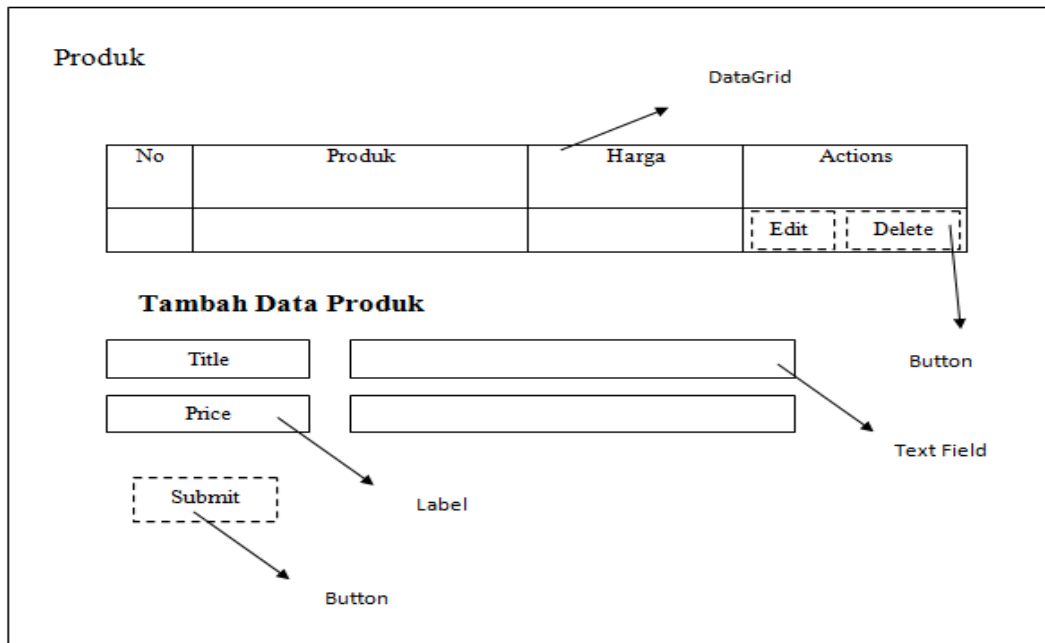
Gambar 23 Halaman Utama



Gambar 24 Create New Invoice (Setup Client)



Gambar 25 Create New Invoice (Add Product)



Gambar 26 Produk

ALL INVOICE

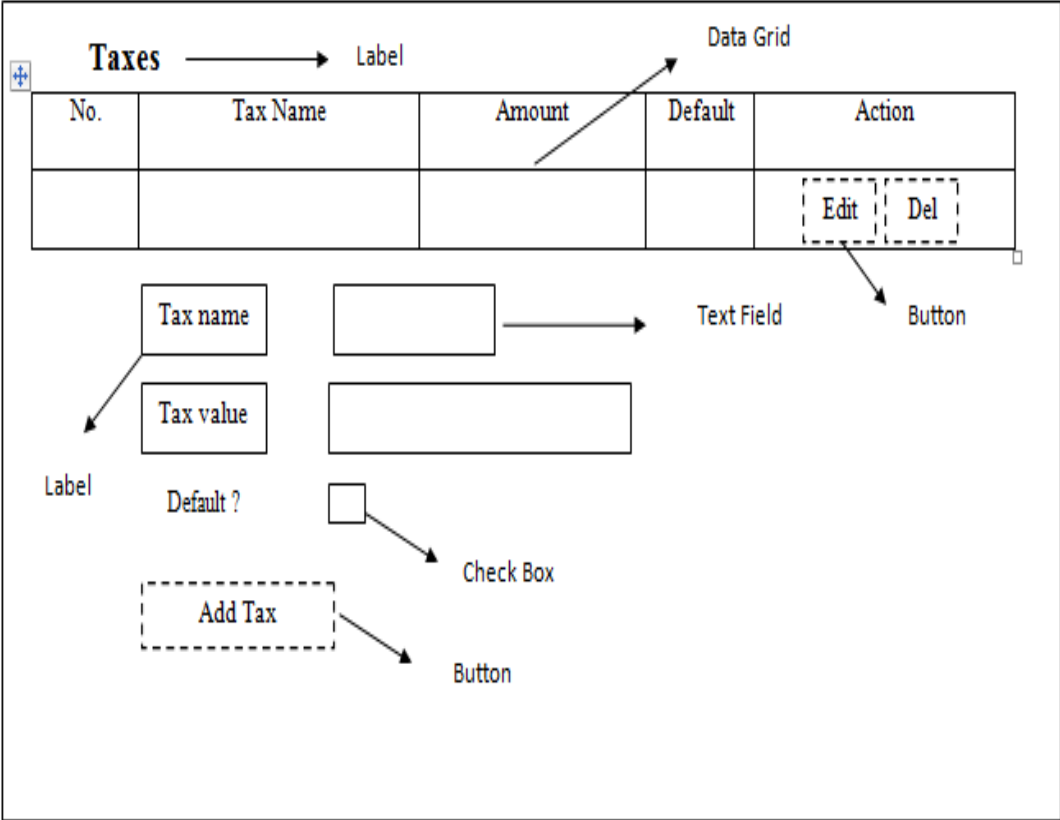
No.	Client Name	Date	Status	Produk	Total	Payment method	Actions
							Paid PDF Del

Gambar 27 All Invoice

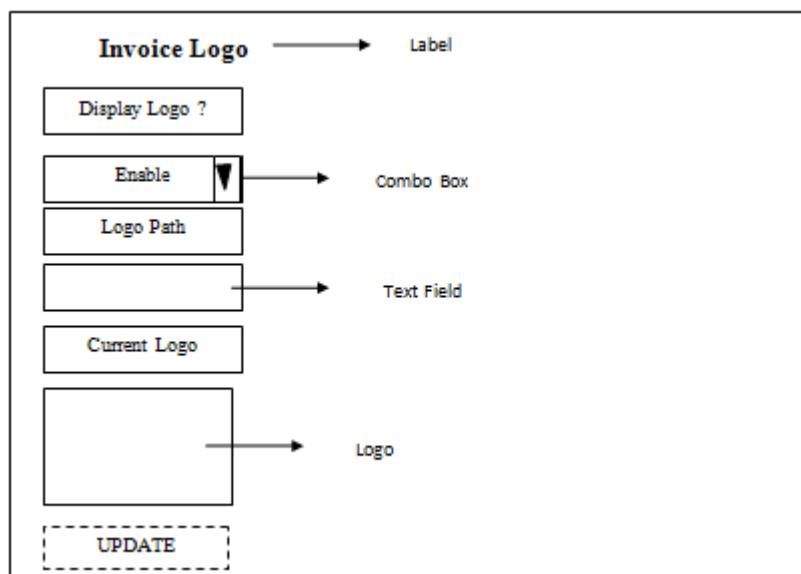
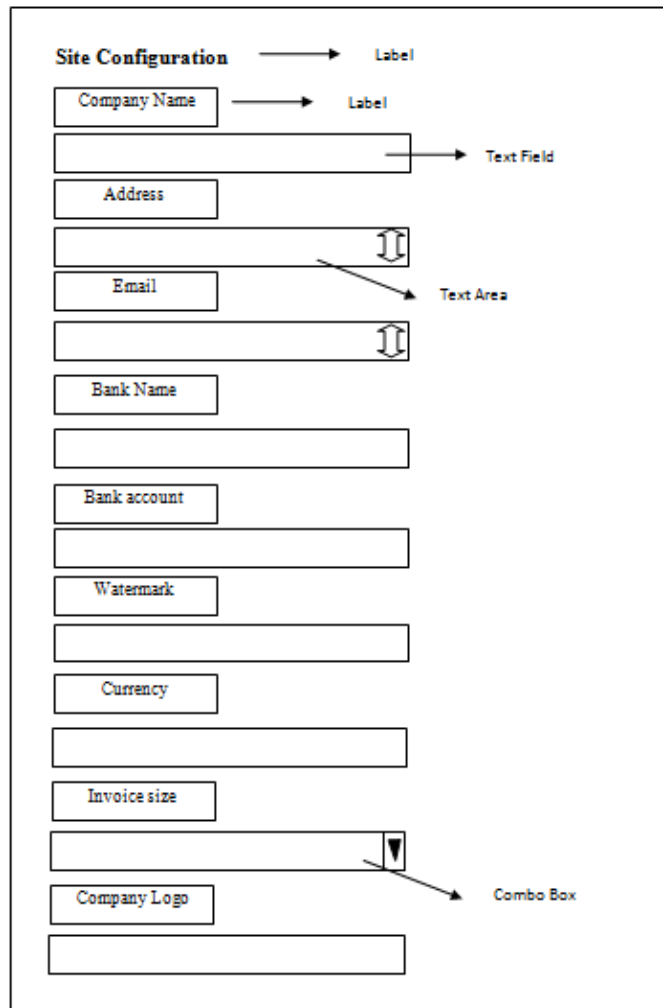
Payment Label

No.	From	Amount	Invoice No.	Date	Actions
					Kuit Del

Gambar 28 Payment



Gambar 29 Taxes



Gambar 30 Setting

Bab II Tinjauan Pustaka

II.1 Unit Bisnis Penunjang Infrastruktur PT. PLN Batam

Unit Bisnis Penunjang Infrastruktur PT PLN Batam adalah divisi yang bergerak dalam pengembangan infrastruktur PT. PLN untuk bisnis telematika. Hal-hal utama yang dikerjakan oleh Unit Bisnis Penunjang Infrastruktur adalah:

- Menangani Fiber Optic milik PT PLN Batam
- Menangani dan mengatur kerjasama dengan pihak ketiga untuk pemanfaatan infrastruktur milik PT PLN Batam.
- Menggunakan Fiber Optic pada Automatic Meter Reading (AMR) sebagai ganti dari GSM.

II.2 Sistem Informasi

Sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan informasi yang diperlukan untuk pengambilan keputusan.

Sistem informasi dalam suatu organisasi dapat dikatakan sebagai suatu sistem yang menyediakan informasi bagi semua tingkatan dalam organisasi tersebut kapan saja diperlukan. Sistem ini menyimpan, mengambil, mengubah, mengolah dan mengkomunikasikan informasi yang diterima dengan menggunakan sistem informasi atau peralatan sistem lainnya.

Komponen Sistem informasi

Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (building blok), yang terdiri dari komponen input, komponen model,

komponen output, komponen teknologi, komponen hardware, komponen software, komponen basis data, dan komponen kontrol. Semua komponen tersebut saling berinteraksi satu dengan yang lain membentuk suatu kesatuan untuk mencapai sasaran.

1. Komponen input

Input mewakili data yang masuk kedalam sistem informasi. Input disini termasuk metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Komponen model

Komponen ini terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.

3. Komponen output

Hasil dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua pemakai sistem.

4. Komponen teknologi

Teknologi merupakan “tool box” dalam sistem informasi, Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, neghasilkan dan mengirimkan keluaran, dan membantu pengendalian dari sistem secara keseluruhan.

5. Komponen hardware

Hardware berperan penting sebagai suatu media penyimpanan vital bagi sistem informasi. Yang berfungsi sebagai tempat untuk menampung

database atau lebih mudah dikatakan sebagai sumber data dan informasi untuk memperlancar dan mempermudah kerja dari sistem informasi.

6. Komponen software

Software berfungsi sebagai tempat untuk mengolah, menghitung dan memanipulasi data yang diambil dari hardware untuk menciptakan suatu informasi.

7. Komponen basis data

Basis data (database) merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi menggunakan perangkat lunak paket yang disebut DBMS (Database Management System).

8. Komponen kontrol

Banyak hal yang dapat merusak sistem informasi, seperti bencana alam, api, temperatur, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, ketidak efisienan, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.[4]

II.3 HTML

Hypertext Markup Language (html) adalah bahasa yang digunakan untuk menulis halaman web. html merupakan pengembangan dari standar pemformatan

dokumen teks yaitu Standard Generalized Markup Language (SGML). html sebenarnya adalah dokumen ASCII atau teks biasa, yang dirancang untuk tidak tergantung pada suatu sistem operasi tertentu. html dibuat oleh Tim Berners-Lee ketika masih bekerja untuk CERN dan dipopulerkan pertama kali oleh browser Mosaic. Selama awal tahun 1990 html mengalami perkembangan yang sangat pesat. Setiap pengembangan html pasti akan menambahkan kemampuan dan fasilitas yang lebih baik dari versi sebelumnya. Namun perkembangan resmi dikeluarkan pada bulan November 1995 oleh IETF (Internet Engineering Task Force). html 2.0 ini merupakan penyempurnaan dari html+ (1993). html 3.0 (1995) memberikan kemampuan lebih daripada versi sebelumnya. Sebuah usaha dari World Wide Web Consortium's (W3C) html Working Group pada tahun 1996 menghasilkan html 3.2. html versi ini secara resmi diterbitkan pada bulan Januari 1997. html versi terbaru adalah html 4.01 yang dikeluarkan secara resmi oleh W3C pada tanggal 24 April 1998. html merupakan perbaikan dari html 4.0 yang lebih dulu diterbitkan. (18 Desember 1997).

DASAR-DASAR HTML

Mendesain html berarti melakukan suatu tindakan pemrograman. Namun html bukanlah sebuah bahasa pemrograman. Namun html hanyalah berisi perintah-perintah yang telah terstruktur berupa tag-tag penyusun. Menuliskan tag-tag html tidaklah sebatas hanya memasukkan perintah-perintah tertentu agar html kita dapat di akses oleh browser. Mendesain html adalah sebuah seni tersendiri. Homepage yang merupakan implementasi dari html adalah refleksi dari orang yang membuatnya. Untuk itu kita perlu mendesainnya dengan baik agar para pengunjung homepage yang kita buat merasa senang dan bermanfaat. Mendesain html dapat dilakukan dengan dua cara:

1. Menggunakan html Editor, seperti Microsoft FrontPage, Adobe Dreamweaver, dan lain-lain.

2. Dengan cara menuliskan sendiri secara manual satu persatu tag-tag html ke dalam dokumen html.

Ada kelebihan dan kekurangan dari dua cara di atas. Cara pertama kelebihanannya adalah html Editor merupakan sebuah program yang khusus didesain untuk membuat, melakukan editing bahkan mem-publish ke internet. Dengan kemampuannya menggabungkan kemudahan dan kecanggihan teknologi internet ke dalam dokumen html maka cara ini sangat disukai oleh para pemula dan desainer yang tidak ingin belajar lebih mendalam mengenai html. Sedangkan cara kedua adalah menuliskan secara manual satu persatu tag-tag html. Hal ini sangat disarankan sulit dikarenakan akan memakan tenaga dan waktu ekstra untuk melakukannya, ditambah lagi Anda harus melakukan cara-cara konvensional untuk melihat hasilnya pada web browser. Namun pada cara kedua adalah dasar dari segala bentuk html yang akan Anda pelajari, karena dengan cara itulah Anda akan lebih paham mengenai cara kerja dan berbagai perintah yang biasa dipakai pada bahasa html.

Struktur Dokumen HTML

Di bawah ini adalah struktur dari dokumen HTML , buka Notepad kemudian ketikkan teks di bawah ini:

```
<HTML>
<HEAD>
<TITLE>contoh1.htm</TITLE>
</HEAD>
<BODY>
Kepala atau kop dokumen
<p>
Isi dokumen
</BODY>
</HTML>
```

Setelah selesai klik menu "File" menu dan pilih "Save As" . Akan muncul kotak Save yang akan menyimpan dokumen Anda. Lalu ketikkan contoh1.htm . Pada

kotak yang paling bawah Anda dapat melihat "Save file as Type" or "Save as Type". Defaultnya adalah .txt, jadi Anda harus mengubahnya ke All Files(*.*). Lalu Anda dapat mengklik Save.

Buka Microsoft Internet Explorer dengan mengklik Start, Program lalu pilih Internet Explorer. Internet Explorer akan terbuka lalu Anda dapat memilih menu File, Open lalu cari lokasi file di disket Anda.

Struktur di atas diapit oleh tag <HTML> dan </HTML> yang menandai awal dan akhir dari sebuah dokumen yang digunakan untuk menyisipkan informasi mengenai dokumen, misalkan versi, revisi dan sebagainya. Sedangkan bagian yang diapit oleh tag <TITLE> dan </TITLE> menunjukkan judul dari halaman web Anda.

Bagian kedua yang diapit <BODY> dan </BODY> berisi dokumen atau informasi yang hendak disajikan.[6]

II.4 PHP

PHP adalah singkatan dari "PHP: *Hyper text reprocessor*", yang merupakan sebuah bahasa scripting yang terpasang pada HTML. Sebagian besar sintaks mirip dengan bahasa C, Java dan Perl, ditambah beberapa fungsi PHP yang spesifik. Tujuan utama penggunaan bahasa ini adalah untuk memungkinkan perancang web menulis halaman web dinamik dengan cepat. PHP juga dapat dilihat sebagai pilihan lain dari ASP.NET/C#/VB.NET Microsoft, ColdFusion Macromedia, JSP/Java Sun Microsystems, dan CGI/Perl.

Pada saat ini bahasa PERL dan CGI sudah jauh ketinggalan jaman sehingga sebagian besar designer web banyak beralih ke bahasa *server-side scripting* yang lebih dinamis seperti PHP. Seluruh aplikasi berbasis web dapat dibuat dengan PHP. Namun kekuatan yang paling utama PHP adalah pada konektivitasnya dengan *system database* di dalam web.

System database yang dapat didukung oleh PHP adalah :

1. Oracle
2. MySQLMs.
3. Access Sybase
4. PostgreSQL

Kelebihan PHP dari bahasa pemrograman lain antara lain adalah sebagai berikut :

1. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. Web Server yang mendukung PHP dapat ditemukan dimana - mana dari mulai apache, IIS, Lighttpd, hingga Xitami dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis - milis dan developer yang siap membantu dalam pengembangan.
4. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.
5. PHP adalah bahasa open source yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah system.[2]

II.5 MySQL

MySQL merupakan sebuah perangkat lunak *database management system (DBMS)* yang *multithread, multiuser*. MySQL dikembangkan untuk menangani basis data yang berukuran besar. Dalam perkembangannya yang sangat konstan, MySQL menawarkan fungsi-fungsi yang sangat berguna. MySQL adalah Relational Database Management System (RDBMS) yang

didistribusikan secara gratis dibawah lisensi GPL (General Public License). Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat komersial.

MySQL merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (Structured Query Language). SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu sistem database (DBMS) dapat diketahui dari cara kerja optimizer-nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh user maupun program-program aplikasinya. Sebagai database server, MySQL dapat dikatakan lebih unggul dibandingkan database server lainnya dalam query data. Hal ini terbukti untuk query yang dilakukan oleh single user, kecepatan query MySQL bisa sepuluh kali lebih cepat dari PostgreSQL dan lima kali lebih cepat dibandingkan Interbase.

MySQL memiliki beberapa keistimewaan, antara lain :

1. Portabilitas. MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.
2. Open Source. MySQL didistribusikan secara open source, dibawah lisensi GPL sehingga dapat digunakan secara cuma-cuma.
3. 'Multiuser'. MySQL dapat digunakan oleh beberapa user dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
4. 'Performance tuning'. MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.

5. Jenis Kolom. MySQL memiliki tipe kolom yang sangat kompleks, seperti signed / unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.
6. Perintah dan Fungsi. MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam perintah (query).
7. Keamanan. MySQL memiliki beberapa lapisan sekuritas seperti level subnetmask, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta sandi terenkripsi.
8. Skalabilitas dan Pembatasan. MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman (records) lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.
9. Konektivitas. MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, Unix soket (UNIX), atau Named Pipes (NT).
10. Lokalisasi. MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meski pun demikian, bahasa Indonesia belum termasuk di dalamnya.
11. Antar Muka. MySQL memiliki interface (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (Application Programming Interface).
12. Klien dan Peralatan. MySQL dilengkapi dengan berbagai peralatan (tool) yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk online.

13. Struktur tabel. MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE, dibandingkan basis data lainnya semacam PostgreSQL ataupun Oracle.[3]

II.6 Apache

Apache merupakan Web server open source dan tersedia diberbagai platform. Web server ini sangat terkenal dan banyak digunakan sebagian besar website yang ada di internet. Apache biasanya disertakan dalam setiap instalasi sistem operasi Linux, sedangkan untuk Windows harus di unduh terlebih dahulu. Fungsi dari Apache adalah sebagai perantara *script* program dengan Web server itu sendiri. Jika Apache tidak dalam keadaan *running*, maka *script* yang dibuat tidak bisa dilihat hasilnya dalam *browser*.

Apache memiliki fitur-fitur canggih seperti pesan kesalahan yang dapat dikonfigur, autentikasi berbasis basis data dan lain-lain. Apache juga didukung oleh sejumlah antarmuka pengguna berbasis grafik (GUI) yang memungkinkan penanganan server menjadi mudah. Apache merupakan perangkat lunak *open source* yang dikembangkan oleh komunitas terbuka yang terdiri dari pengembang-pengembang dibawah naungan Apache Software Foundation.

Beberapa dukungan apache :

1. Kontrol akses
2. CGI
3. PHP
4. SSI

Kelebihan dan Kekurangan :

1. Open Source
2. Proses instalasinya mudah

3. Mudah untuk dikustomisasi (Apache hanya punya 4 file konfigurasi) ataupun menambah
4. peripheral dalam web servernya
5. Bisa digunakan di berbagai platform mesin dari mainframe sampai embedded system
6. Ada komunitas yang besar sehingga mudah mencari solusinya jika ditemukan masalah Mudah dicari di internet.
7. Server Apache otomatis berkomunikasi dengan clientnya untuk mendapatkan tampilan web terbaik.
8. Keamanannya bagus dan bisa menggunakan SSL (Secure Socket Layer)
Tidak bisa mengatur load seperti IIS sehingga akan terus memfork proses baru sampai dalam batas yang diijinkan OS. Akan memudahkan penyerang karena RAM akan cepat habis
9. Mudah diserang oleh DoS (pada Apache versi 1.3 dan versi 2 sampai versi 2.0.36)
10. Apache tidak memproses karakter kutip dalam string Referrer dan User-Agent yang
11. dikirimkan oleh Client. Ini berarti Client dapat memformulasi inputnya secara hati-hati untuk
12. merusak format baris log akses.[1][3]

II.7 CSS

Cascading Style Sheet atau yang disingkat CSS memungkinkan untuk mendesain (style) tampilan dokumen (Php dan HTML) dengan memisahkan isi dari dokumen HTML dengan kode untuk menampilkannya (CSS). Di dalam

dunia desain web, layout atau mengatur letak web, format huruf, dan lain nya yang ada dalam sebuah web menjadi bagian yang amat penting bahkan bisa jadi yang paling lama dibanding dengan bagain desain web lainnya. Karena jika kita membua layout yang salah, maka web kita juga akan salah diterjemahkan oleh browser. Hal itu juga bisa disebabkan karena setiap browser memiliki dukungan bahasa(CSS) yang berbeda dan tentunya default dari browsernya sendiri yang berbeda-beda.

Jika kita memiliki banyak file HTML, kita hanya perlu satu file CSS, sehingga ketika kita mengganti jenis huruf pada file CSS maka semua file HTML yang berhubungan pada file CSS tersebut akan berubah. CSS distandarisasi oleh W3C (World Wide Web Consortium). CSS dapat dipasang pada dikumen HTML yang telah jadi. Dalam melayout sebuah web ada dua metode yang sering dipakai untuk membuat sebuah layout halaman web ada dua metode yang sering dipakai untuk membuat sebuah layout halaman web yaitu:

1. Metode Table
2. Metode Div

Manfaat dari CSS:

1. Kode HTML menjadi lebih sederhana dan lebih mudah diatur,
2. Ukuran file menjadi lebih kecil, sehingga load file lebih cepat,
3. Mudah untuk merubah tampilan, hanya dengan merubah file CSS saja,
4. Dapat berkolaborasi dengan JavaScript.
5. Digunakan dalam hampir semua web browser.

Ada 3 cara untuk memasang CSS pada dokumen HTML yaitu:

1. External Style Sheet (file CSS berbeda dari file HTML)

2. Internal Style Sheet (Kode CSS dipasang di dalam tag head HTML)
3. Inline Style Sheet (Kode CSS langsung dipasang di tag HTML ,Tidak direkomendasikan). [5].

II.8 Ajax

AJAX adalah singkatan dari “Asynchronous JavaScript and XML”, yang dibuat dari serangkaian teknologi dengan berbagai kemampuan : JavaScript, XML dan sebuah method komunikasi asinkron antara client dan server.

3 teknologi yang saling berinteraksi : JavaScript menangkap isyarat, gerak serta aksi. Sebagaimana pada situasi yang mungkin terjadi, JavaScript menggunakan jalur komunikasi pada server (object JavaScript dengan nama XMLHttpRequest) untuk memanggil method yang tersimpan pada server dan menggunakan XML sebagai mekanisme pengiriman data. Jika JavaScript pada client telah menerima respon dari server, maka JavaScript akan menggunakan kemampuan bawaannya untuk memanipulasi struktur DOM halaman untuk menambahkan content yang didapat dari server. Perubahan yang terjadi pada struktur DOM kemudian diterjemahkan oleh browser pada client, sehingga meningkatkan efek interaktivitas pada user.

AJAX sebagai sebuah Arsitektur

Pada awalnya AJAX dibuat sebagai sebuah rangkaian dari teknologi, namun kemudian AJAX mengalami perkembangan. Sebagai contoh, adanya aplikasi web yang tidak menggunakan XML dalam mentransfer data dari client ke server. Cara tersebut dilakukan dengan menggunakan object XMLHttpRequest.

Berdasarkan realita di atas, beberapa pihak mendefinisikan sebagai paradigma baru dalam pemrograman, disamping teknologi yang menyediakan fungsionalitas. Mari kita bahas lebih mendalam tentang arsitektur AJAX.

Pengembang telah mengembangkan web programming sebelum hadirnya AJAX : action dari user yang membutuhkan data dari server ditampilkan dalam halaman yang digunakan, dimana data request dari user dikirimkan menuju server. Setelah mengolah halaman tersebut, server menampilkan halaman baru bagi user yang mengandung hasil dari proses sebelumnya.

Permasalahan dari macam arsitektur tersebut adalah lambat dan cukup memakan waktu, terutama bila dibandingkan dengan aplikasi desktop. Aplikasi desktop mampu merespon cepat atas request dari user, aplikasi ini tidak memproses ulang masing – masing komponen interface yang akan ditampilkan sebagai respon.

AJAX menggunakan arsitektur pemrograman tersebut pada aplikasi Web. Daripada memberikan sebuah halaman penuh pada server dan mendapatkan pula sebuah halaman penuh sebagai hasil operasi, AJAX mengizinkan kita untuk mengirimkan request dalam ukuran yang lebih kecil pada server. Halaman yang terpakai hanya termodifikasi untuk menampilkan hasil, bukan tergantikan dengan sebuah halaman baru. Faktor penting yang lain dari arsitektur AJAX adalah request dan response dijalankan secara asinkron : AJAX tidak melarang user untuk melakukan proses lain pada halaman yang dipakai. User dapat mengisi dan menggunakan area lain pada halaman, sedangkan AJAX bekerja pada background. Yang terakhir, AJAX mengizinkan user untuk berinteraksi dengan server sebagai respon terhadap seluruh hal yang dilakukan oleh user. Arsitektur yang ada sebelumnya hanya mengizinkan kita untuk berkomunikasi dengan server pada saat user menekan tombol atau link yang akan mengirim data pada halaman. AJAX memperbolehkan untuk me-request data baru dari server dalam bentuk mouseovers, keypress dan even lain yang dikenali oleh JavaScript.

II.9 JQuery

jQuery adalah javascript library, jQuery mempunyai semboyan “write less, do more”. jQuery dirancang untuk memperingkas kode-kode javascript. JQuery adalah javascript library yang cepat dan ringan untuk menangani dokumen HTML, menangani event, membuat animasi dan interaksi ajax. JQuery

dirancang untuk mengubah cara anda menulis javascript.

Mengapa memilih jQuery?

Ada beberapa alasan mengapa lebih baik menggunakan jQuery daripada library lainnya, antara lain:

1. Kompatibel dengan hampir seluruh browser
2. jQuery telah digunakan oleh website-website raksasa
3. Kompatibel dengan seluruh versi CSS (dari CSS 1 sampai CSS 3)
4. Didukung oleh banyak komunitas
5. Disupport oleh plugin yang lengkap
6. Filenya hanya satu dan ukurannya relatif kecil, sekitar 20kb
7. Open source atau Free
8. jQuery lebih diminati oleh para developer web saat ini

Apa kemampuan yang dimiliki jQuery?

Berikut beberapa kemampuan yang dimiliki yang dapat membuat kita berdecak kagum:

- Mempermudah akses dan manipulasi elemen tertentu pada dokumen.
- Mempermudah modifikasi/perubahan tampilan halaman web.
- Mempersingkat Ajax (*Asynchronous Javascript and XML*).
- Memiliki API (*Application Programming Interface*).
- Mampu merespon interaksi antara user dengan halaman web dengan lebih cepat.

- Menyediakan fasilitas untuk membuat animasi sekilas Flash dengan mudah. [2]

II.10 FPDF

FPDF adalah kelas dari PHP yang memungkinkan penggunanya membuat berbentuk PDF. FPDF tidak melibatkan sama sekali yang berasal dari PHP itu sendiri. F pada FPDF memiliki pengertian atau arti, yang maksudnya adalah tidak diperlukan izin atau bayaran untuk memakainya (Bebas). Pengguna dapat dengan bebas memakai dan merombak kelas yang ada pada FPDF sesuai dengan kebutuhan. Hasil dari FPDF ini dapat disimpan ke Dalam Bentuk .pdf dan dapat digunakan sesuai kebutuhan.

PHP merupakan salah satu bahasa pemrograman berbasis web yang banyak digunakan saat ini. Tidak hanya digunakan untuk membuat sebuah website, tetapi juga untuk berbagai aplikasi dan sistem berbasis web seperti sistem penjualan, sistem keuangan, hingga sistem administrasi sekolah. Salah satu bagian yang tidak terpisahkan dari sebuah sistem, adalah laporan. Dalam sebuah aplikasi berbasis web (web application), laporan dapat dibuat dan disajikan dalam berbagai bentuk seperti laporan PDF, laporan berbasis CSS (CSS reporting) dan sebagainya. Dari berbagai pilihan tersebut, laporan dalam bentuk PDF memiliki keunggulan karena memiliki format standar, keamanan terhadap perubahan data serta dukungan berbagai browser dan sistem operasi dalam membuka file PDF.

Dalam tutorial ini akan dibahas mengenai bagaimana membuat laporan PDF melalui PHP menggunakan library FPDF. FPDF (<http://fpdf.org>) merupakan salah satu library PHP yang memungkinkan untuk membuat / generate file PDF secara langsung dari PHP (on-the-fly) tanpa harus menginstall atau mengaktifkan modul PHP tertentu (seperti PDFLib). Tutorial ini akan diawali dengan penjelasan singkat mengenai FPDF beserta keunggulannya, lalu dilanjutkan dengan contoh sederhana penggunaan FPDF untuk meng-generate file PDF. Setelah itu akan dibahas mengenai contoh membuat laporan sederhana dimana datanya diambil

dari database MySQL. Di bagian akhir tutorial, akan disertakan sumber dan referensi terkait dengan tutorial ini.

Seperti sudah disebutkan di atas, FPDF merupakan suatu kumpulan class (library) PHP yang memungkinkan penggunanya dapat membuat halaman PDF dengan mudah dan tanpa perlu menginstall modul tambahan. F dari nama FPDF merupakan singkatan dari Free, jadi FPDF dapat didownload, digunakan dan dikembangkan secara bebas. FPDF dapat didownload secara gratis di <http://fpdf.org/en/download.php> beserta dokumentasi lengkapnya di <http://fpdf.org/en/doc/index.php>. Tersedia juga dokumentasi dalam bahasa Indonesia.

Berikut ini beberapa kelebihan dari FPDF yang terdapat di situs resminya, <http://fpdf.org>, adalah:

- Terdapat pilihan utk ukuran unit, format halaman dan margin
- Memiliki kemampuan dalam manajemen header, footer dan tabel-
Memiliki fasilitas utk pemisah halaman otomatis
- Memiliki fasilitas utk pemisah baris otomatis dan perataan teks-
Mendukung JPEG dan PNG
- Mendukung warna dan link
- Mendukung kompresi halaman
- Mendukung TrueType Font dan Encoding

Bab IV Implementasi dan Pengujian

IV.1 Implementasi Kelas PHP

PHP		
No	Nama Kelas	Nama File Fisik
1.	Koneksi	db_config.php
2.	Index	Index.php
3.	Create New Invoice	newInvoice.php
	Setup_Client	Step1.php
	Add_Product	Step2.php
	Save/Export	Step3.php
4.	Product	produk.php
	tambah_produk	tambah.php
	update_produk	prosesupdate.php
7.	Payment	Payment.php
	Cetak_kuitansi	cetakKuitansi.php
8.	Taxes	Taxes.php
9.	Setting	Config.php

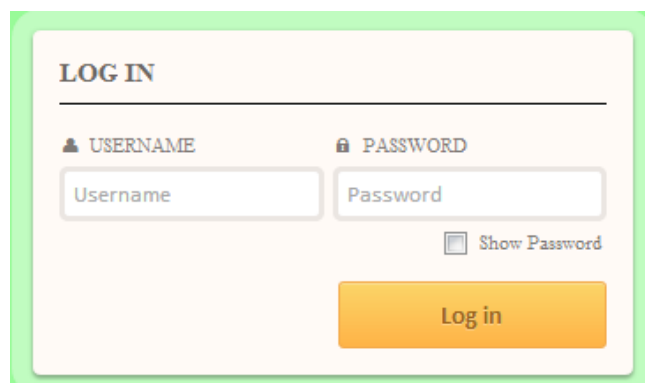
Table 1 Implentasi Kelas PHP

IV.2 Implementasi Antarmuka PHP

PHP		
No	Nama Kelas	Nama File Fisik
1.	Index	Index.php
2.	Create New Invoice	newInvoice.php
3.	Produk	produk.php
4.	View_Invoice	allInvoice.php
5.	Payment	payment.php
6.	Taxes	taxes.php
7.	Setting	config.php

Table 2 Implementasi Antarmuka PHP

IV.2.1 Implementasi Login



The image shows a login interface with the following elements:

- LOG IN**: Title of the form.
- USERNAME**: Label for the first input field.
- PASSWORD**: Label for the second input field.
- Username**: Placeholder text in the first input field.
- Password**: Placeholder text in the second input field.
- Show Password**: A checkbox with the label "Show Password" located below the password field.
- Log in**: A yellow button at the bottom right of the form.

Gambar 31 Antarmuka Login



IV.2.2 Implementasi Home

bright.
PLN Batam


Invoice

Home New Invoice Product All Invoice Payment Taxes Setting

Last 10 Invoice

ID	Client Name	Date	Paid?	Prod. Total	Payment Method	Actions
1	PT.PLN	Yesterday	Paid	1 Rp. 66.372.150	Transfer	 

Last 10 Payment

ID	From	Amount	Invoice No.	Payment Method.	Date	Actions
1	PT.PLN	Rp. 66.372.150	#100356	Transfer	Today	

Gambar 32 Antarmuka Home

IV.2.3 Implementasi Create New Invoice

Setup Client Add Products Save/Export

Create New Invoice

CLIENT NAME

CATEGORY

EMAIL

PHONE

ADDRESS

LOCATION

PAYMENT METHOD

NEXT STEP

Gambar 33 Antarmuka Setup Client

Invoice #100345

Title	Qty	Net	Taxes	Total

Subtotal: Rp.0
Taxes: Rp.0
TOTAL: Rp.0.00

PRODUCT TITLE

Taxes

DESCRIPTION

PRICE

QUANTITY

ADD PRODUK

FINISH

Gambar 34 Antarmuka Add Produk

[✓ Setup Client](#)
[➤ Add Products](#)
[➤ Save/Export](#)

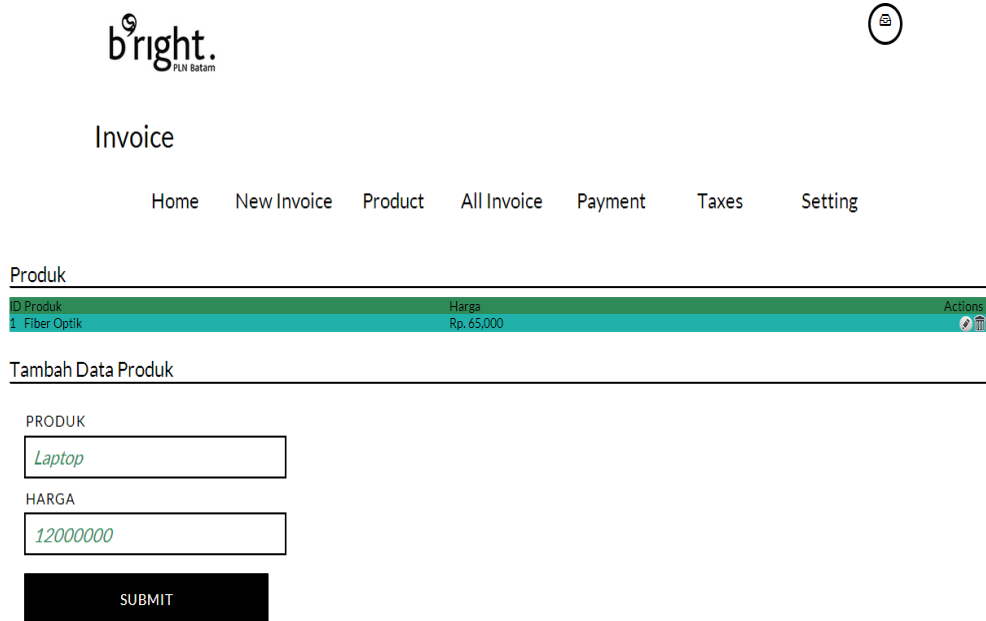
Create New Invoice



Download PDF

Gambar 35 Antarmuka Save/Export Invoice

IV.2.4 Implementasi Produk





b'right.
PLN Batam

Invoice

Home New Invoice Product All Invoice Payment Taxes Setting

Produk

ID Produk	Harga	Actions
1 Fiber Optik	Rp. 65,000	 

Tambah Data Produk

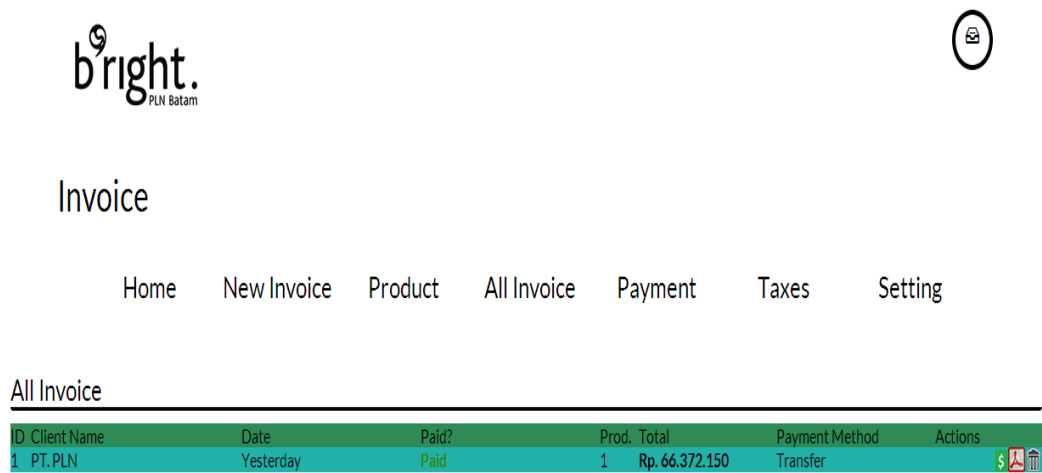
PRODUK

HARGA

SUBMIT

Gambar 36 Antarmuka Produk

IV.2.5 Implementasi View Invoice





b'right.
PLN Batam

Invoice

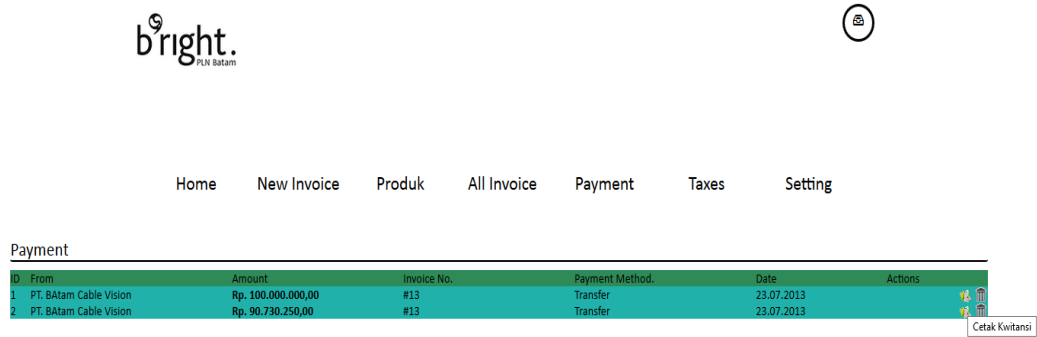
Home New Invoice Product All Invoice Payment Taxes Setting

All Invoice

ID	Client Name	Date	Paid?	Prod. Total	Payment Method	Actions
1	PT. PLN	Yesterday	Paid	1 Rp. 66.372.150	Transfer	 

Gambar 37 Antarmuka View Invoice

IV.2.6 Implementasi Payment



b'right.
PLN Batam

Home New Invoice Produk All Invoice Payment Taxes Setting

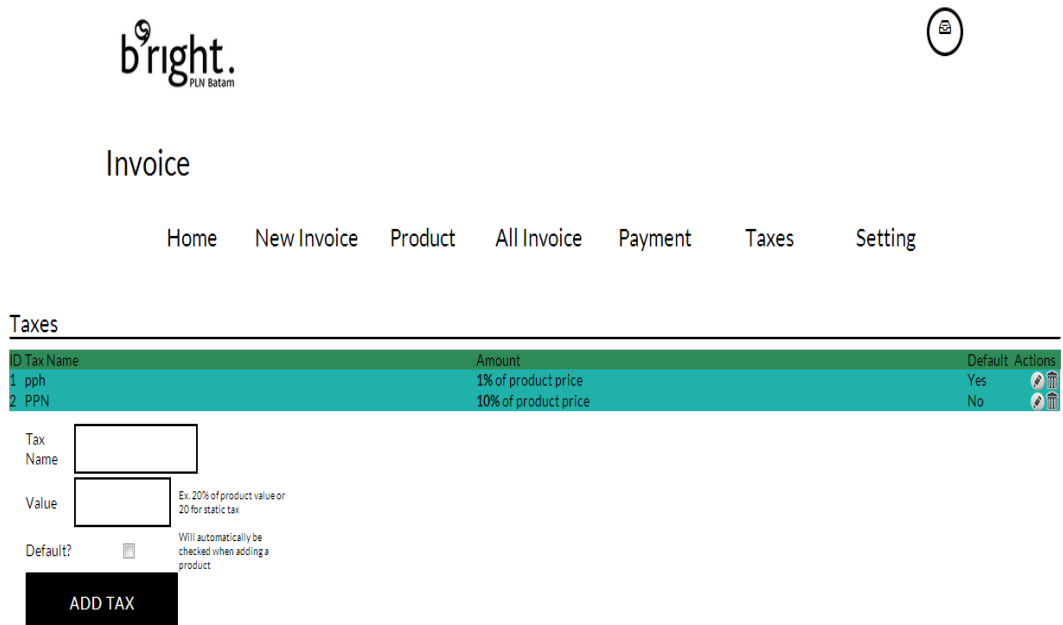
Payment

ID	From	Amount	Invoice No.	Payment Method	Date	Actions
1	PT. Batam Cable Vision	Rp. 100.000.000,00	#13	Transfer	23.07.2013	
2	PT. Batam Cable Vision	Rp. 90.730.250,00	#13	Transfer	23.07.2013	

Cetak Kwitansi

Gambar 38 Antarmuka Payment

IV.2.7 Implementasi Taxes



b'right.
PLN Batam

Invoice

Home New Invoice Product All Invoice Payment Taxes Setting

Taxes

ID	Tax Name	Amount	Default	Actions
1	pph	1% of product price	Yes	
2	PPN	10% of product price	No	

Tax Name

Value Ex. 20% of product value or 20 for static tax

Default? Will automatically be checked when adding a product

ADD TAX

IV.2.8 Implementasi Setting

Site Configuration

COMPANY NAME

PT PLN BATAM

ADDRESS

JALAN ENGKU PUTRI 3
BATAM CENTER

FOOTER NOTES

Thank you for your business.
Unit Bisnis Multimedia PT PLN Batam

EMAIL

Email: riki@plnbatam.com

BANK NAME

Bank Mandiri

WEBSITE

BANK ACCOUNT

WATERMARK

CURRENCY

Rp.

WEBSITE

BANK ACCOUNT

WATERMARK

CURRENCY

INVOICE SIZE

Invoice Logo

DISPLAY LOGO?

LOGO PATH

CURRENT LOGO


UPDATE

Update Password

MASUKKAN PASSWORD LAMA

MASUKKAN PASSWORD BARU

MASUKKAN PASSWORD BARU

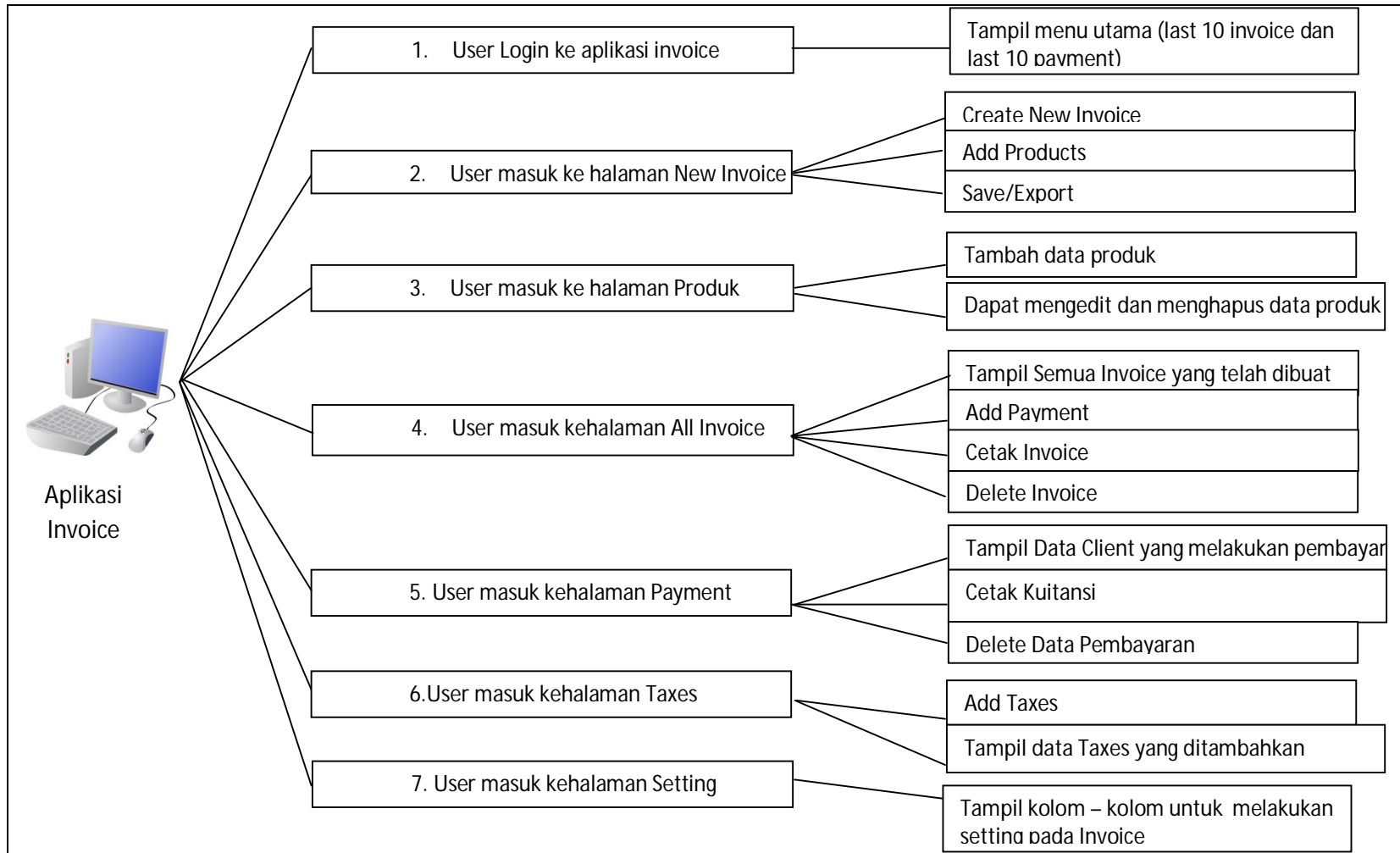
UPDATE PASSWORD

Gambar 39 Antamuka Setting

IV.3 Pengujian

IV.3.1 Skenario Pengujian

Skenario pengujian dilakukan terhadap fungsi untuk mengetahui bagaimana Aplikasi Invoice Berbasis Web bekerja. Skenario Aplikasi Invoice Berbasis Web dapat dilihat pada gambar berikut dibawah ini :



Gambar 40 Skenario Pengujian

IV.3.2 Hasil Rincian Pengujian

No	Kelas	Fungsi	Usecase	Skenario	Data Uji	Hasil Pengujian	Keterangan
1.	Index	Select	Melakukan Login	User melakukan login dan masuk ke Halaman utama.	Login berhasil: Masuk ke menu utama Login tidak berhasil: Login gagal	Login Berhasil	
						Login gagal	
2.	GUIHome	Select		Sistem menampilkan Last 10 Invoice dan Last 10 Payment	Koneksi berhasil: Menampilkan data Koneksi tidak berhasil: Tidak menampilkan data	Tampilkan data	
						Tampilkan data gagal	
3.	GUINewInvoice	Select Insert Delete	Kelola Invoice	Add Invoice, View Invoice, Delete Invoice, dan Cetak Invoice	Membuat Invoice Baru, Menampilkan Invoice yang dibuat, Menghapus Invoice, dan Mencetak Invoice	Invoice Berhasil dibuat dan dicetak dalam bentuk file PDF	
4.	GUIProduk	Select Insert	Kelola Produk	Add Product, Update Produk, dan Delete Product	Membuat produk baru, mengubah data produk, menghapus data produk	Buat Produk Baru dan Tampilkan Data Produk.	
		Update				Mengubah data produk	

		Delete				Menghapus data produk	
5.	GUIPayment	Select	Kelola Payment	View Payment, Add Payment, dan Cetak Kuitansi	Menampilkan data client yang melakukan pembayaran, Mencetak Kuitansi dan Menghapus data pembayaran.	Tampilkan data sukses	
		Insert				Mencetak Kuitansi	
		Delete				Menghapus data pembayaran	
6.	GUITaxes	Cetak	Kelola Taxes	View Taxes, Add Taxes, Edit Taxes, dan Delete Taxes	Menampilkan data Taxes, Menambah data Taxes, dan Menghapus data Taxes	Tampilkan data Taxes	
		Select				Tambah data Taxes	
		Insert				Hapus data Taxes	
7.	GUISetting	Update	Setiing	View Setting, Edit Setting, dan Update Setting	Menampilkan Halaman/Kolom Setting, Mengubah Setting, dan Mengupdate Setting	Tampilkan Halaman Setting	
		Select				Ubah Setting	
		Insert				Update Setting	

Table 3 Hasil Rincian Pengujian

Bab V Kesimpulan dan Saran

V.1. Kesimpulan

Kesimpulan dari Aplikasi Penagihan (Invoice) berbasis Web yang telah kami buat adalah :

1. Aplikasi ini dapat diimplementasikan dan memenuhi kebutuhan dari Unit Bisnis Penunjang Infrastruktur yang digunakan untuk mengoptimalkan pelayanan dan kinerja seluruh civitas dari Unit Bisnis Penunjang Infrastruktur PT. PLN Batam dalam melakukan pengolahan berbagai data surat tagihan (*invoice*) secara efektif dan efisien kepada perusahaan rekanan yang bekerja sama dalam pemanfaatan inventaris milik PT. PLN Batam.

V.2. Saran

Setelah dilakukan pengujian serta evaluasi, penulis menyarankan akan lebih baik jika aplikasi invoice diintegrasikan dengan aplikasi data rekanan agar lebih kompleks guna mempermudah pengguna dalam melakukan pengolahan data tagihan berdasarkan data client yang tersimpan pada aplikasi data rekanan dalam proses pembuatan surat tagihan.

Daftar Pustaka

- [1]. Bowen, Rich. Apache Administrator's handbook, Sams Publishing, United States of America. Indianapolis : Sams, 2002.
- [2]. Desrizal. Panduan Lengkap PHP, Ajax, JQuery. Codingwear : Programmer in Style, 2010.
- [3]. Galbraith, Patrick. Developing Web Applications with Perl, memcached, MySQL and Apache. Indiana : Wiley, 2009.
- [4]. Magaline, Ferdinand , N. Mahamudu Billy. Sistem Informasi, 2008.
- [5]. Pouncey, Ian & York, Richard. Beginning CSS : "cascading style sheets for web design". Indiana : Wiley, 2011.
- [6]. Supriadi, Aris. WEB DESIGN HTML (Hypertext Markup Language). Jakarta : CIKINI, 2010.
- [7]. Westriningsih. Membuat Web E-Commerce dengan Adobe Dreamweaver CS5.5. Yogyakarta : ANDI, 2012.